Computational Structural Dynamics of Proteins

Timothy R. Lezon
Department of Computational & Systems Biology
University of Pittsburgh

Contents

Preface

The course that I teach desperately needed a text book. This is it, or the beginnings of it. This text does not cover all the material that is covered in the lectures, but will continue to evolve. Notably, many figures are missing.

Little of the material in here is not presented in greater depth in other texts, but it is convenient to have everything in one place.

The level is of introductory graduate students in computational biology. I have made every effort to include necessary background information on mathematics and physics to make the "real" material accessible to students who lack strong backgrounds in math and physics.

There are no mathematical proofs in this text. Some equations are derived, but I don't use theorems and lemmas. Feel free to consult a mathematician if you don't believe something presented.

Chapter 1

A Crash Course in Linear Algebra

You will need to understand some math to learn protein structural dynamics. There is no way around it. One chapter is not enough to teach all the math that is needed, but it will touch on some of the important points and familiarize you with the style and notation that will be used (hopefully) consistently throughout this text. This is a chapter on linear algebra, which deals with solving multiple equations simultaneously. Conceptually, it is the mathematics of knowing which way things are pointing and how objects transform other objects.

1.1 The problem with hyperspace

Linear algebra deals with spaces of high dimension, which can be a tough concept to grasp when we only live in a 3D world. Even though it can be difficult to imagine higher dimensions, the mathematics involved in 10 or 20 or infinite dimensions is essentially the same as that involved in three dimensions. There are mathematical formulas for manipulating hyperspheres and hypercubes and other fantastical objects that we can't visualize, but the math is the same, and it all boils down to recursion. The general strategy is first to become comfortable with ideas in a one dimensional space. Then we look at the same concepts in two dimensions, which is still easy to visualize. If we understand how ideas change between one and two dimensions, then we can straightforwardly add a third dimension. And if we know how to add a third dimension, then we can add a fourth dimension, and so on.

A *dimension* is an axis on a graph. Anything that can be quantified constitutes its own one dimensional space, or its own one dimensional subspace of some higher dimensional space. We are familiar with seeing two dimensional plots. Any two-column data table can be plotted in a 2D graph. For example, a biologist studying a population of critters may record the weight and length of individuals sampled from the population. *Length* and *weight* are two dimensions in which the population is measured, and we can talk about the *length-weight space* and its properties.

The dimensions of any space are the properties that are being quantified. In computational biology we might hear about the 20-dimensional amino acid space, or the 20^N dimensional *sequence space* of a protein with N residues. In the context of protein struc-

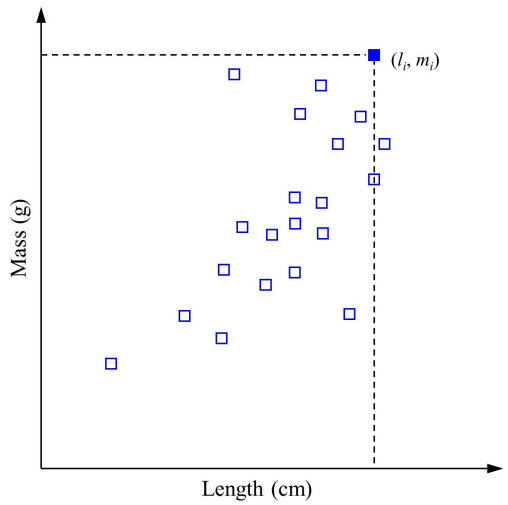


Figure 1.1: An example of a two-dimensional space, possibly corresponding to biometrics of critters. The position of critter i in this space is given by the vector $\mathbf{r}_i = (l_i w_i)^T$, corresponding to the ordered pair $l_i, w_i)$.

tural dynamics, we will be considering systems of n dimensions, where n is usually on the order of the number of residues in a protein.

1.2 A note on notation

Throughout this text, *scalars* will appear in italics, *vectors* will be lowercase and bold faced, and *matrices* will be uppercase and bold faced. All of these terms will be defined shortly. There may be exceptions to these rules, for example when vectors are combined to form a matrix, but hopefully the mathematics will be clear from the context.

Object	Appearance	Description
Scalar	a or A	A number
Vector	$\mathbf{x}$	A column of numbers
Vector component	x_i	The number in the i^{th} row of $\mathbf{x}$
Vector transpose	$\tilde{\mathbf{x}}$ or $\mathbf{x}^T$	A row of numbers
Matrix	$\mathbf{M}$	An array of numbers
Matrix component	M_{ij}	The number in the i^{th} row and j^{th} column of $\mathbf{M}$
Matrix column	$\mathbf{m}^{(k)}$ or $\mathbf{M}^{(k)}$	The k^{th} column of the matrix $\mathbf{M}$
Submatrix	$\mathbf{M_{IJ}}$	A small matrix within the larger matrix $\mathbf{M}$

1.3 Vectors: The concept

We've probably (hopefully) learned in basic physics that vectors are objects with magnitude and direction, and scalars are objects with magnitude but no direction. This is fine. The magnitude of a vector is a scalar. Displacement is a vector; its magnitude is a scalar, called *distance*. Velocity is a vector; its magnitude, *speed*, is a scalar. And so on. These particular examples are limited to 3D space (which, somewhat confusingly, is just called *space*), but *vectors can have any number of dimensions*. To go back to the data table example, if a biologist measures n properties of an organism, we can associate with the organism an n-dimensional vector, the components of which are the values of the n measurements. The organism's position in the n-dimensional space is specified by this vector. Regardless of the number of dimensions that it has, a vector is just an arrow. The length of the arrow is given by the Euclidian norm:

$$|\mathbf{a}| = \sqrt{\sum_{i=1}^{n} a_i^2} \ ,$$

where a_i refers to component i of the vector $\mathbf{a}$. We can define the *dot product*, or *inner product*, between two vectors as

$$\mathbf{a} \cdot \mathbf{b} = \sum_{i=1}^{n} a_i b_i \ ,$$

which allows us to write the magnitude of a vector as $|\mathbf{a}| = \sqrt{\mathbf{a} \cdot \mathbf{a}}$. We can show using basic trigonometry that $\mathbf{a} \cdot \mathbf{b} = |\mathbf{a}||\mathbf{b}| \cos\theta_{ab}$, where θ_{ab} is the angle between the vectors $\mathbf{a}$ and $\mathbf{b}$.

Problem 1.1. *Show that* $\mathbf{a} \cdot \mathbf{b} = |\mathbf{a}||\mathbf{b}| \cos\theta_{ab}$.

Thus, we can use the dot product to find the angle between two vectors, regardless of their dimension. Why is this? If we put the tails of the two vectors at the same point – call it A – and we call the tips of the vectors points B and C, then the triangle ABC uniquely defines a plane. An angle in a plane is just a regular angle that we can measure with a protractor. So two vectors define a space of two dimensions (unless one happens to be a multiple of the other, but we'll get to that later). What if we add a third vector? We can put its tail at A and call its tip D. If D is in the plane ABC, then our three vectors still exist in a 2D space. But if D is outside the plane ABC, then we need three dimensions to describe our three vectors. That is, we need the 2D space ABC, plus some 1D space that is perpendicular to ABC. If we continue adding vectors this way, we will find that a set of m vectors that have n components will span at most a space of m dimensions if $n > m$. If $n < m$, then the vectors will span at most n dimensions for what I hope are obvious reasons.

The dot product can also be used to find the projection of one vector onto another. That is, it tells us what fraction of one vector is made up of a second vector. If the dot product happens to be zero, it tells us that the angle between the two vectors is 90°, or that the vectors do not project onto each other at all. When two vectors have zero inner product, they are said to be *orthogonal*.

The arrow idea is useful for conceptualizing vectors, but it is not the best way of manipulating them mathematically. For that we will turn to matrices, which will bring us back to vectors and hopefully provide some insights into linear algebra as a whole.

1.4 Matrices

A matrix is a rectangular array of numbers. An $m \times n$ matrix has m rows (going across) and n columns (going down). Matrix elements are indexed by row and column.

$$\mathbf{A} = \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & & A_{2n} \\ \vdots & & \ddots & \\ A_{m1} & A_{m2} & \cdots & A_{mn} \end{pmatrix} .$$

A matrix might be something as simple as a data table, where each element is the value of some feature of some sample. The biologist in the earlier example might construct a matrix of two columns and m rows. The columns are the *length* and *mass* measurements for the individuals in a population, and each row represents an individual.

1.4.1 Transpose

The *transpose* of a matrix is the same matrix with its rows and columns swapped. It is denoted with a tilde or superscript T.

$$\tilde{\mathbf{A}} = \begin{pmatrix} A_{11} & A_{21} & \cdots & A_{m1} \\ A_{12} & A_{22} & & A_{m2} \\ \vdots & & \ddots & \\ A_{1n} & A_{2n} & \cdots & A_{mn} \end{pmatrix} .$$

Note that $\tilde{A}_{ij} = A_{ji}$, and that $\tilde{\tilde{\mathbf{A}}} = \mathbf{A}$ (the transpose of the transpose is the original matrix).

1.4.2 Matrix arithmetic

If two matrices are the same size, we can add them just by adding their components:

$$\mathbf{A} + \mathbf{B} = \begin{pmatrix} A_{11} + B_{11} & \cdots & A_{1n} + B_{1n} \\ \vdots & \ddots & \\ A_{m1} + B_{m1} & \cdots & A_{mn} + B_{mn} \end{pmatrix} .$$

Multiplying matrices is a little trickier. Usually we multiply matrices only if the number of columns of the first is equal to the number of rows of the second. The product is a matrix with the same number of rows as the first matrix and the same number of columns as the

second matrix. That is, if $\mathbf{A}$ is $m \times n$ and $\mathbf{B}$ is $n \times p$, then $\mathbf{C} = \mathbf{AB}$ is $m \times p$. Its components are given by

$$\begin{aligned} C_{ij} &= (\mathbf{AB})_{ij} \\ &= \sum_{k=1}^{n} A_{ik} B_{kj} \ . \end{aligned} \tag{1.1}$$

Graphically,

$$\left(\begin{array}{c} \\ \mathbf{C}_{m\times p} \\ \\ \end{array} \right) = \left(\begin{array}{c} \\ \mathbf{A}_{m\times n} \\ \\ \end{array} \right) \left(\begin{array}{c} \\ \mathbf{B}_{n\times p} \\ \\ \end{array} \right)$$

A kind of explanation for why the matrix product has this form will be given shortly, but for now just commit it to memory. You will definitely need to know how to multiply matrices, so it's worth taking a couple of seconds to make sure that you know how to calculate a matrix product using the equation above. You don't have to understand it or know what it *means* right now, but just how to use it.

Matrix multiplication is not commutative. That is $\mathbf{AB} \neq \mathbf{BA}$. However,

$$\widetilde{\mathbf{AB}} = \tilde{\mathbf{B}}\tilde{\mathbf{A}} \ . \tag{1.2}$$

This is an important identity that will pop up from time to time.

Eq. 1.1 allows us to define the *identity matrix*, $\mathbf{1}$, sometimes called $\mathbf{I}$, with elements $\mathbf{1}_{ij} = \delta_{ij}$. Graphically,

$$\mathbf{1} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} \ .$$

The identity is a square matrix with 1's on the diagonal and 0's everywhere else. Multiplying a matrix by $\mathbf{1}$ is like multiplying a scalar by one. In general, the dimension of $\mathbf{1}$ will be implied by its use in equations. For the $m \times n$ matrix $\mathbf{A}$, we can write $\mathbf{1A} = \mathbf{A}$, where $\mathbf{1}$ is the $m \times m$ identity. Alternatively, we could write $\mathbf{A1} = \mathbf{A}$, in which case $\mathbf{1}$ represents the $n \times n$ identity. The dimension of $\mathbf{1}$ should match up with the dimension of the matrix or matrices that it's multiplying.

Matrix division does not exist. Instead, we use the matrix *inverse*, $\mathbf{A}^{-1}$, defined as

$$\mathbf{A}\mathbf{A}^{-1} = \mathbf{A}^{-1}\mathbf{A} = \mathbf{1} \ .$$

Only square matrices have true inverses according to this definition, but we will later see how to construct *pseudo*inverses of matrices of arbitrary size. The matrix inverse is important insofar as it enables much of matrix algebra. It also reminds us of an important rule of matrix algebra: order matters. Consider solving the scalar equation $y = ax$ for a. Dividing each side

by x, we find $a = y/x$. If we want to avoid division, we can multiply each side by the inverse of x, yielding the equivalent expression $a = yx^{-1}$. And because of the commutitive property of scalar multiplication, we can switch the order of the right-hand side, giving $a = x^{-1}y$.

Now let us consider an analogous matrix problem, $\mathbf{Y} = \mathbf{AX}$, and let us also assume that $\mathbf{X}$ is invertible. As matrix multiplication is not commutitive, order matters:

$$\begin{aligned} \mathbf{Y} &= \mathbf{AX} \\ \mathbf{YX}^{-1} &= \mathbf{AXX}^{-1} \\ \mathbf{YX}^{-1} &= \mathbf{A1} \\ \mathbf{A} &= \mathbf{YX}^{-1} \ . \end{aligned}$$

To get rid of $\mathbf{X}$, we multiplied both sides *from the right* by $\mathbf{X}^{-1}$. This removed the $\mathbf{X}$ from the right-hand side of the equation and added a $\mathbf{X}^{-1}$ to the right of $\mathbf{Y}$ on the left-hand side of the equation. The result, $\mathbf{A} = \mathbf{YX}^{-1}$, is not the same as $\mathbf{A} = \mathbf{X}^{-1}\mathbf{Y}$, because order matters! Keeping track of whether multiplication takes place from the left or from the right is the primary difference between matrix algebra and scalar algebra.

1.4.3 Vectors are just simple matrices

Now back to vectors. A vector is just a matrix with a single column, like this:

$$\mathbf{a} = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} \ .$$

This definition allows us to construct matrices by grouping together vectors. An $m \times n$ matrix might be thought of as an ordered set of n vectors that have m dimensions.

The transpose of a vector is a row of the same numbers:

$$\tilde{\mathbf{a}} = (a_1 \ a_2 \ \cdots \ a_n) \ .$$

Note that $\tilde{a}_i = a_i$ because we only need one index per element. It is understood that a_i is in the first (and only) column of $\mathbf{a}$, and in the first (and only) row of $\tilde{\mathbf{a}}$. Sometimes vectors are explicitly specified to be *column* vectors or *row* vectors, but this is only for the purposes of mathematics and does not really change the concept (at least where we are concerned). As a rule, a vector that multiplies something from the left is a row, and a vector that multiplies something from the right is a column. In either case, the vector is just a list of numbers, or an arrow. By convention, a vector without a qualifier is a column.

Using the definition of transpose we can write the dot product as

$$\begin{aligned} \mathbf{a} \cdot \mathbf{b} &\equiv \tilde{\mathbf{a}}\mathbf{b} \\ &= \sum_{i=1}^{n} a_i b_i \ , \end{aligned}$$

where the sum is taken over the n components of the vectors. So a dot product is just a row matrix times a column matrix. When constructing the dot product, we are implicitly transposing the first vector into a row, and then taking a standard matrix product. The product is a matrix with one row and one column, or a scalar. For this reason, the "dot" notation will henceforth be avoided to prevent confusion. Think of vectors as one-column matrices, and explicitly consider transposes as needed for algebraic purposes. This will avoid some of the confusion that can come from mixing vectors and matrices.

Going back to the case of multiplying two rectangular matrices, $\mathbf{A}$ (which is $m \times n$) and $\mathbf{B}$ (which is $n \times p$): Now that we know how inner products are handled in matrix notation, we can see that the elements of $\mathbf{C} = \mathbf{AB}$ are just the inner products of the vectors given by the columns of $\tilde{\mathbf{A}}$ with those given by $\mathbf{B}$. The product of two matrices is just a matrix of inner products.

We can construct the *outer product*, $\mathbf{a}\tilde{\mathbf{b}}$, by multiplying a column vector by a row vector. The outer product is itself a matrix with the same number of rows as $\mathbf{a}$ and the same number of columns as $\tilde{\mathbf{b}}$:

$$\begin{pmatrix} a_1b_1 & a_1b_2 & \cdots & a_1b_n \\ \vdots & \ddots & & \\ a_mb_1 & a_mb_2 & \cdots & a_mb_n \end{pmatrix} = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{pmatrix} (b_1\ b_2\ \cdots\ b_n)\ .$$

For the outer product, the two vectors do not have to have the same number of components. If they do, however, you'll get a square matrix, and the sum over the diagonal elements (the *trace* of the matrix) is the inner product of the two vectors.

We can also multiply matrices and vectors. For example, if we multiply an $m \times n$ matrix with an n-component column vector, the result is an m-component column vector:

$$\begin{aligned} \mathbf{Ab} &= \begin{pmatrix} A_{11} & \cdots & A_{1n} \\ \vdots & \ddots & \\ A_{m1} & \cdots & A_{mn} \end{pmatrix} \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix} \\ &= \begin{pmatrix} A_{11}b_1 + A_{12}b_2 + \cdots + A_{1n}b_n \\ \vdots \\ A_{m1}b_1 + A_{m2}b_2 + \cdots + A_{mn}b_n \end{pmatrix} \end{aligned}$$

We can take the transpose of this using the identity given in Eq. 1.2, $\widetilde{\mathbf{Ab}} = \tilde{\mathbf{b}}\tilde{\mathbf{A}}$, which is a row vector times an $n \times m$ matrix. We know that $\mathbf{Ab}$ is a column, so its transpose must be a row. Therefore, a row vector times a matrix is a row vector. Once again, we can see that under normal circumstances, when a vector multiplies a matrix from the left it is considered a row, but when it multiplies a matrix from the right, it is considered a column.

Multiplying vectors with matrices is a staple of linear algebra, and now is a good time to start thinking about what it means. An $m \times n$ matrix might be thought of as an object that eats an n-component vector and spits out an m-component vector[1]. Depending on the

[1]This wording was stolen from John Baez's website, which I should reference here.

matrix and what it represents, the vector that it spits out may exist in the same space as the vector that is taken in, or in an entirely different vector space.

1.5 Eigensystems

Consider the equation

$$\mathbf{A}\mathbf{v} = \lambda\mathbf{v} \ ,$$

where $\mathbf{A}$ is a square matrix. When $\mathbf{A}$ multiplies the vector $\mathbf{v}$, the result is the same vector scaled by a factor λ. If this equation (which I slickly failed to motivate with any real-world examples) holds, then that means that $\mathbf{v}$ has a special relationship with $\mathbf{A}$: All $\mathbf{A}$ does to $\mathbf{v}$ is stretch or shrink it. The *direction* that $\mathbf{v}$ points in is unaffectd by $\mathbf{A}$. We call $\mathbf{v}$ and *eigenvector* of $\mathbf{A}$, and λ its associated *eigenvalue*. If $\mathbf{A}$ is $n \times n$, we expect it to have n eigenvalues, each associated with its own eigenvector[2]. Let's label the eigenvectors with an index $k = 1 \ldots n$. Now we have n equations,

$$\mathbf{A}\mathbf{v}^{(k)} = \lambda_k \mathbf{v}^{(k)}$$

or, by making the $\mathbf{v}^{(k)}$'s the columns of a matrix $\mathbf{V}$, we get

$$\mathbf{A}\mathbf{V} = \mathbf{V}\mathbf{\Lambda} \ ,$$

where $\mathbf{\Lambda}$ is a diagonal matrix containing the eigenvalues λ_k. It might seem odd that $\mathbf{\Lambda}$ wound up on the right side of $\mathbf{V}$, but it's a good exercise to convince yourself that the above equation is mathematically correct. As a final step, we can multiply each side from the right by $\mathbf{V}^{-1}$, leaving us with

$$\mathbf{A} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1} \ .$$

This tells us that when $\mathbf{A}$ multiplies a vector from the left, it has the effect of first transforming the vector into some new coordinate system (using $\mathbf{V}^{-1}$), then stretching each component of the transformed vector by some factor (using $\mathbf{\Lambda}$), and then transforming the modified vector back to the original coordinate system (using $\mathbf{V}$). The eigenvector matrix $\mathbf{V}$ and its inverse then have the interesting jobs of transforming vectors out of and into some space where $\mathbf{A}$ is diagonal. It is easier to work with diagonal matrices because they treat each of a vector's components independently. We will encounter several cases where using a diagonal matrix will come in handy.

1.6 The special case of the real symmetric matrix

There are a number of "special" matrices that are defined and discussed throughout the mathematical and physical sciences literature. Here we are mostly concerned with one kind

[2]Maybe the reason for this isn't clear, but for the curious: We can rewrite $\mathbf{A}\mathbf{v} = \lambda\mathbf{v}$ as $(\mathbf{A} - \lambda\mathbf{1})\mathbf{v} = 0$. The only non-trivial solutions are those for which the determinant $|\mathbf{A} - \lambda\mathbf{1}| = 0$. Expanding the determinant yields an n^{th} order polynomial in λ, indicating n roots or eigenvalues. It is left to the student to learn what a determinant is.

of matrix: The real symmetric matrix. Matrices like this are *real* because their elements have no imaginary parts. They are *symmetric* about the diagonal, such that $A_{ij} = A_{ji}$. Symmetric matrices are necessarily square. Other matrices may pop up from time to time, but there is little need to explain their properties in detail here. The student is referred to the text by Arfken [1] for a more complete discussion.

A real symmetric matrix $\mathbf{A}$ can be decomposed or factored

$$\mathbf{A} = \mathbf{V}\mathbf{\Lambda}\tilde{\mathbf{V}} \ , \tag{1.3}$$

where $\mathbf{V}$ is an orthogonal matrix and $\mathbf{\Lambda}$ is diagonal (that is, all off-diagonal elements of $\mathbf{\Lambda}$ are identically zero). We write this compactly using the Kronecker delta function: $\Lambda_{ij} = \delta_{ij}\lambda_i$.

We have identified $\mathbf{V}$ as an orthogonal matrix, which means that its transpose is its inverse: $\mathbf{V}\tilde{\mathbf{V}} = \tilde{\mathbf{V}}\mathbf{V} = \mathbf{1}$. In words, the columns of $\mathbf{V}$ are orthogonal to each other – they have zero inner product – and each has unit length. The columns (and also rows) of $\mathbf{V}$ form an *orthonormal basis* over the space spanned by $\mathbf{A}$. Orthogonal matrices may be thought of as matrices that perform rotations. They preserve the lengths of vectors, as well as the relative handedness of coordinate systems (i.e., they do not cause inversions or reflections). The columns of $\mathbf{V}$ are the *eigenvectors* of $\mathbf{A}$, and the diagonal elements of $\mathbf{\Lambda}$ are the corresponding *eigenvalues*. When $\mathbf{A}$ has units, we think of $\mathbf{V}$ as dimensionless directions, and we give $\mathbf{\Lambda}$ the units of $\mathbf{A}$.

Note that any column of $\mathbf{V}$ can be multiplied by -1 without affecting $\mathbf{A}$. The matrix $\mathbf{V}$ in the eigendecomposition is therefore not unique, but unique only up to an arbitrary choice of direction for each of its columns. It is also worth noting that two or more eigenvectors that share the same eigenvalue are *degenerate* and not unique. When two eigenvectors have the same eigenvalue, any linear combination of these two eigenvectors will also have the same eigenvalue. Thus, degenerate eigenvectors are not uniquely defined.

Eq. 1.3 contains a great deal of information. First, it tells us that multiplying a vector by $\mathbf{A}$ is the same as rotating the vector to a new coordinate system, stretching each component of this rotated vector by some amount λ_i, and rotating it back to the original frame. It also tells us that the matrix $\mathbf{A}$ is the sum of n symmetric matrices, each given by the outer product of one eigenvector with itself and scaled by an eigenvalue. Finally, Eq. 1.3 suggests a way of calculating the inverse of $\mathbf{A}$ as $\mathbf{A}^{-1} = \mathbf{V}\mathbf{\Lambda}^{-1}\tilde{\mathbf{V}}$, where $(\Lambda^{-1})_{ij} = \delta_{ij}/\lambda_i$.

There is one problem with this definition of $\mathbf{A}^{-1}$: What happens if $\lambda_i = 0$? Certainly we don't want to divide by zero, so we take the drastic measure of ignoring it. Writing Eq. 1.3 element-wise:

$$A_{ij} = \sum_{k=1}^{n} V_{ik}V_{jk}\lambda_k \ .$$

We see that we will get the same result if we ignore all terms in which $\lambda_k = 0$. Suppose that there are T such eigenvalues, then we can give them the last p indices and re-write the elements of $\mathbf{A}$ as

$$A_{ij} = \sum_{k=1}^{n-p} V_{ik}V_{jk}\lambda_k \ .$$

To invert, we can simply write

$$A^{\dagger}_{ij} = \sum_{k=1}^{n-p} V_{ik} V_{jk} / \lambda_k \ . \tag{1.4}$$

Eq. 1.4 defines the *pseudo-inverse* of $\mathbf{A}$ and provides a well-defined approximation to the actual inverse of $\mathbf{A}$, which doesn't exist. Note that we display the pseudo-inverse with a dagger, $\mathbf{A}^{\dagger}$, to identify it as something that is not quite the true inverse. When working with linear algebra and an inverse is needed, we generally will solve equations assuming that the inverse exists, and substitute the pseudo-inverse at the end, although this can be mathematically dangerous.

Problem 1.2. *Consider two orthogonal modes, $\mathbf{v}^{(1)}$ and $\mathbf{v}^{(2)}$, that share the eigenvalue λ. Show that an arbitrary linear combination $\mathbf{v}' = a_1\mathbf{v}^{(1)} + a_2\mathbf{v}^{(2)}$ is also a valid mode with eigenvalue λ.*

Problem 1.3. *Find the eigenvalues and eigenvectors of*

$$\begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix} \ .$$

Bibliography

[1] George B. Arfken and Hans J. Weber. *Mathematical Methods for Physicists*. Academic Press, San Diego, CA, 1995.

Chapter 2

Structural Dynamics of Chain Molecules

2.1 Introduction

Before we can perform theoretical and computational studies on biomolecules, it would behoove us to have an idea of how other, similar molecules that are not associated with living organisms behave. The molecules of life – DNA, RNA and proteins – are chain molecules; however, their behavior is remarkably different than other chain molecules, such as might be found in household plastics. Here we will investigate the statistical physics of chain molecules using simple but well-studied models. We will start by defining some statistical measures of chain conformations, and then look at successively more complicated models. In the end we will see that simple models reveal a surprising amount of information about the statistical properties of chains, but that simple models are not sufficient for describing the general properties of proteins. Additional information on the topics covered here can be found in the texts of Flory [1], Dill [2], Phillips [3] and Hiemenz [4].

2.2 Statistical measures of chain conformation

First we will establish some conventions for discussion. Consider a chain as an ordered set of N *beads*, or *particles*, $1 \ldots N$, connected by $N-1$ *links*. For convenience, it will be useful to define $n \equiv N-1$ as the number of links. Depending on our system of study, beads may be atoms, residues, monomers, etc., and links may be bonds (between atoms) or pseudobonds (e.g., between C_α atoms in a protein). The location of bead i is given by $\mathbf{r}_i = (x_i, y_i, z_i)^T$, and link i points from bead i to bead $i+1$: $\mathbf{l}_i = \mathbf{r}_{i+1} - \mathbf{r}_i$ and has length $l_i = \sqrt{(\mathbf{r}_{i+1} - \mathbf{r}_i)^2}$.

The link length l_i does not have to be the same for all i, but we will often assume that this is the case for simplicity. When considering proteins, for example, there are three distinct backbone bond lengths, corresponding to N-C_α, C_α-C and C-N bonds. If all backbone atoms are treated as beads, then there will be three different bond lengths. If instead only C_α atoms act as beads, then all links will have approximately equal length (3.8Å). The primary use of

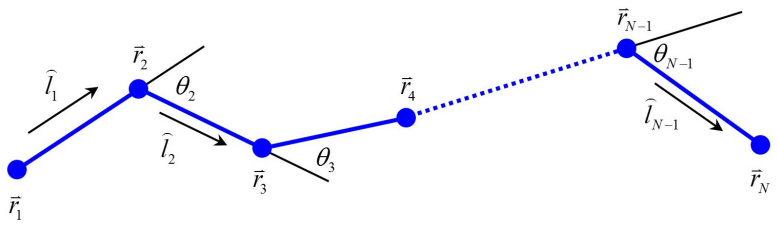


Figure 2.1: A simple model chain molecule.

the link length (and other chain details) in these notes is to calculate statistical averages, so the average link length, $l = \frac{1}{n}\sum_{i=1}^{n}\sqrt{\mathbf{l}_i \cdot \mathbf{l}_i}$ is in general sufficient.

The position of the first bead on the chain, $\mathbf{r}_1$, is arbitrary. Given $\mathbf{r}_1$, we can use the definition of $\mathbf{l}_1$ to find the position of the second bead as $\mathbf{r}_2 = \mathbf{r}_1 + \mathbf{l}_1$. Continuing in this way, the position of bead i is

$$\mathbf{r}_i = \mathbf{r}_1 + \sum_{j=1}^{i-1} \mathbf{l}_j \ .$$

The *end-to-end vector* is defined as

$$\begin{aligned} \mathbf{h} &\equiv \mathbf{r}_N - \mathbf{r}_1 \\ &= \sum_{i}^{N-1} \mathbf{l}_i \ . \end{aligned} \tag{2.1}$$

For practical reasons, $\mathbf{h}$ is not a particularly informative measure of chain conformation. Real chain molecules move – not just internally, but externally. A rigid chain will tend to rotate and translate in space, even in the absence of internal motions of its beads relative to each other. This rotation is spatially isotropic, meaning that all orientations are equally likely. Thus, if we average $\mathbf{h}$ over all chain orientations, we will find that it becomes zero: $\langle \mathbf{h} \rangle = 0$.

2.2.1 End-to-end distance

A slightly better measure, and the one that will get us started on calculating statistical features of chains, is the end-to-end *distance*:

$$h \equiv \langle |\mathbf{h}|^2 \rangle^{1/2} \ . \tag{2.2}$$

The squared end-to-end distance will prove useful in many calculations and can be found:

$$\begin{aligned} h^2 &= \langle \left(\sum_{i=1}^{N-1} \mathbf{l}_i \right) \cdot \left(\sum_{j=1}^{N-1} \mathbf{l}_j \right) \rangle \\ &= \langle \sum_{i=1}^{N-1} \sum_{j=1}^{N-1} \mathbf{l}_i \cdot \mathbf{l}_j \rangle \\ &= \langle \sum_{i=1}^{N-1} \mathbf{l}_i \cdot \mathbf{l}_i + \sum_{i=1}^{N-1} \sum_{j \neq i} \mathbf{l}_i \cdot \mathbf{l}_j \rangle \\ &= \sum_{i=1}^{N-1} \langle \mathbf{l}_i \cdot \mathbf{l}_i \rangle + \sum_{i=1}^{N-1} \sum_{j \neq i} \langle \mathbf{l}_i \cdot \mathbf{l}_j \rangle \\ &= nl^2 + \sum_{i=1}^{N-1} \sum_{j \neq i} \langle \mathbf{l}_i \cdot \mathbf{l}_j \rangle \ , \end{aligned} \tag{2.3}$$

where we have made use of the average squared link length, l^2, in the last step. The first term in Eq. 2.3 captures a universal dependence of h on the number of links and their average length. The second term depends on the constraints of the chain and is model-dependent.

Fun as it is to calculate expressions like Eq. 2.3, they are often of little use in reality. The mean end-to-end distance is difficult to accurately measure in experiments, and it only provides information on two of the beads on the chain. The remainder of the chain, which is often of great interest, is ignored by h, leaving something to be desired. Nonetheless, the simple form of h^2 is useful because it relates to more descriptive measures of chain conformation.

2.2.2 Radius of gyration

A more commonly used statistical descriptor of chain conformation is the *radius of gyration*, R_g, defined as the average distance of any bead from the chain's center of mass. Recall that the center of mass is given by

$$\mathbf{r}_{CM} = \frac{\sum_{i=1}^{N} m_i \mathbf{r}_i}{\sum_{i=1}^{N} m_i} \ ,$$

where m_i is the mass of bead i. Defining the position of bead i relative to the center of mass as $\mathbf{s}_i \equiv \mathbf{r}_i - \mathbf{r}_{CM}$, the radius of gyration is

$$R_g \equiv \langle s^2 \rangle^{1/2} \ . \tag{2.4}$$

Eq. 2.4 is valid not only for chains, but for any distribution of points in space. R_g is a measure of the spread of a distribution: Small R_g implies a compact distribution, whereas large R_g indicates an open or extended distribution. It can be shown that R_g is related to the average distance between particles. An expression that is equivalent to Eq. 2.4 is

$$R_g^2 = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j>i} \langle |r_{ij}|^2 \rangle \ . \tag{2.5}$$

Here r_{ij} simply means the distance between particles i and j. As with h, it is often easier to work with R_g^2. Not only can R_g apply to any distribution of discrete points, it can also be generalized to continua:

$$R_g^2 = \int \mathrm{d}V \rho(\mathbf{r})(\mathbf{r} - \mathbf{r}_{CM})^2 \ .$$

Applying the above expression to a uniform sphere of radius R, we can see that $R_g = \sqrt{\frac{3}{5}}R$.

It might be fun to think about the relationship between h and R_g. Consider a chain in a random conformation with some R_g. Without knowing anything else about the chain, we might assume that its beads are distributed uniformly in space, amounting to something like a sphere of radius $R = \sqrt{\frac{5}{3}}R_g$. The first bead of the chain has an equal probability of being anywhere in the sphere, as does the last bead. From Eq. 2.5, we know that the average distance between the ends of the chain is R_g. So we might expect $R_g^2 \sim h^2$, naively. This can be shown rigorously for various cases, but as a simple approximation, it shows that knowing h^2 tells us something about R_g^2. Thus, if you can't calculate R_g, it may suffice to calculate h^2. Our initial excursion into the end-to-end distance wasn't all for naught, after all.

Problem 2.1. *Show that the radius of gyration for a system of N particles satisfies*

$$R_g^2 = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j>i} \langle |r_{ij}|^2 \rangle \ ,$$

where r_{ij} is the distance between particles i and j.

Problem 2.2. *Find R_g for a sphere of radius R_A and density ρ_A encased by a spherical shell of outer radius $R_B = 2R_A$ and density $\rho_B = 2\rho_A$.*

Problem 2.3. *Find the mean square radius of gyration of an infinitely thin rod of length L, with mass density (per unit length) λ in the center $L/2$ section and mass density 2λ for the $L/4$ sections at each end.*

2.2.3 Persistence length

A third measure of chain conformation that may come in handy is the *persistence length*,

$$\begin{aligned} \xi_p &= \frac{1}{l} \sum_{j=i}^{n} \langle \mathbf{l}_i \cdot \mathbf{l}_j \rangle \\ &= \frac{1}{l} \left[\langle \mathbf{l}_i \cdot \mathbf{l}_i \rangle + \langle \mathbf{l}_i \cdot \mathbf{l}_{i+1} \rangle + \ldots + \langle \mathbf{l}_i \cdot \mathbf{l}_n \rangle \right] \ . \end{aligned} \tag{2.6}$$

Flory [1] defines this as the "average sum of the projections of all bonds $j : j \geq i$ on an arbitrary bond i in an indefinitely long chain." Persistence length is a measure of the chain's tendency to remain straight, or the average distance that a chain travels before turning 90°.

Happily, ξ_p is also related to h^2. The arbitrary link that we have referenced is somewhere in the middle of the chain, far removed from either of the ends, so we could just as easily calculated ξ_p by summing from 1 to i instead of from i to n:

$$\xi_p = \langle \frac{\mathbf{l}_i}{l} \cdot \sum_{j=i}^{n} \mathbf{l}_j \rangle = \langle \frac{\mathbf{l}_i}{l} \cdot \sum_{j=1}^{i} \mathbf{l}_j \rangle \ .$$

Combining these two equivalent definitions for ξ_p,

$$2\xi_p = \frac{1}{l} \sum_{j=1}^{n} \langle \mathbf{l}_i \cdot \mathbf{l}_j \rangle + \frac{1}{l} \langle \mathbf{l}_i \cdot \mathbf{l}_i \rangle \ .$$

Understanding that for a homogeneous chain, ξ_p should be independent of the reference link, one can average it over all links:

$$\begin{aligned} 2\xi_p &= \frac{1}{n} \sum_{i=1}^{n} \frac{1}{l} \sum_{j=1}^{n} \langle \mathbf{l}_i \cdot \mathbf{l}_j \rangle + l \\ \xi_p &= \frac{h^2}{2nl} + \frac{l}{2} \ . \end{aligned} \tag{2.7}$$

Once again, we can relate an informative quantity, ξ_p, to an easily calculable one, h^2.

2.3 Freely jointed chain

The simplest model of a chain is the Freely Jointed Chain, or Random Flight. This model assumes no restrictions on bond angles and amounts to a random walk in three dimensions. Even though the assumptions (no bond angle restrictions, no penalty for self-intersection, no solvent) may make this model appear to be comically simple, it makes a surprisingly accurate first attempt at exploring chain molecules.

2.3.1 FJC: End-to-end distance

From Eq. 2.3,

$$h^2 = nl^2 + \sum_{i=1}^{n} \sum_{j \neq i} \langle \mathbf{l}_i \cdot \mathbf{l}_j \rangle \ . \tag{2.8}$$

The second term on the right-hand side of Eq. 2.8 is zero. The chain is free to rotate about all bonds, so $\langle \mathbf{l}_i \cdot \mathbf{l}_j \rangle = l^2 \delta_{ij} \ \forall \ i, j$. This can be shown easily by integrating over the conformations of the chain. Note that this result implies that $\xi_p = l$ for the freely jointed chain, indicating that the chain has no persistence past one link. The end-to-end distance,

$$h = \sqrt{n} l \ , \tag{2.9}$$

recovers the scaling that we find for a random walk.

Problem 2.4. *Show that*

$$\sum_{i=1}^{n}\sum_{j\neq i}\langle \mathbf{l}_i \cdot \mathbf{l}_j\rangle = 0$$

for a freely jointed chain.

Problem 2.5. *Consider a freely jointed chain with n_A steps of length l_A and n_B steps of length l_B. Find the mean square end-to-end distance, $\langle h^2\rangle$, for alternating, random and diblock cases. Are they the same or different? Why?*

2.3.2 FJC: Radius of gyration

That was easy enough, right? Now, what about the radius of gyration of the FJC? Let's start from Eq. 2.5:

$$\begin{aligned} R_g^2 &= \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j>i}\langle |r_{ij}|^2\rangle \\ &= \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j>i}|j-i|l^2 \\ &= \frac{l^2}{N^2}\sum_{k=1}^{N-1}k(N-k) \\ &= \frac{l^2}{N^2}\left[N\sum_{k=1}^{N-1}k - \sum_{k=1}^{N-1}k^2\right] . \end{aligned} \tag{2.10}$$

The two geometric series in Eq. 2.10 can be simplified algebraically. The first one is just the number of elements in the upper triangle of a square matrix, but the closed form of the second series is more involved (A simple derivation appears in B). Substituting:

$$\begin{aligned} R_g^2 &= \frac{l^2}{N^2}\left[\frac{N^2(N-1)}{2} - \frac{N(N-1)(2N-1)}{6}\right] \\ &= \frac{l^2}{6}\left[N - \frac{1}{N}\right] . \end{aligned} \tag{2.11}$$

For $N \gg 1$, the second term is negligible, leaving

$$\begin{aligned} R_g^2 &\approx \frac{Nl^2}{6} \\ &= \frac{h^2+l^2}{6} . \end{aligned}$$

So R_g^2 indeed goes like h^2 for FJC, agreeing with our earlier approximation. In the limit of small l, $R_g^2 \sim h^2/6$.

2.3.3 Distribution of end-to-end vector for FJC

As its alternate name ("Random Flight") implies, the FJC is just diffusion in three dimensions. We know that, in one dimension, the probability for traveling a distance Δx after taking n steps of length l is

$$P(n, \Delta x) = \frac{1}{\sqrt{2\pi n l^2}} \exp\left\{-\frac{x^2}{2nl^2}\right\} .$$

We can conceptually extend this to a random flight in three dimensions by altering the step size or the number of steps. Consider a random walk on a 3D lattice. There is no difference between $x-$, $y-$ and $z-$ directions, so we expect $P(n, \Delta x) = P(n, \Delta y) = P(n, \Delta z)$. If our walk has n total steps, then we expect $n/3$ steps to be taken in each direction. Alternatively, we might argue that a single step of length l in 3D can be decomposed into $x-$, $y-$ and $z-$ components: $l^2 = l_x^2 + l_y^2 + l_z^2$. Again invoking spatial isotropy, we find $l_x = l_y = l_z = l/\sqrt{3}$. Thus, if we perform a 3D random walk starting at the origin and using steps of length l, the probability of the walk having some $x-$component after n steps is

$$P'(n, x) = \sqrt{\frac{3}{2\pi n l^2}} \exp\left\{-\frac{3x^2}{2nl^2}\right\} ,$$

and similarly for the $y-$ and $z-$ components. As usual, this result can be shown more rigorously by those so inclined. The probability of finding a final displacement vector $\mathbf{h}$ after n steps of a random flight in 3D is then

$$\begin{aligned} P(n, \mathbf{h}) &= P'(n, h_x) P'(n, h_y) P'(n, h_z) \\ &= \left[\frac{3}{2\pi n l^2}\right]^{3/2} \exp\left\{-\frac{3h^2}{2nl^2}\right\} . \end{aligned} \tag{2.12}$$

Does this result make sense? It indicates that the probability of winding up at $\mathbf{h}$ depends on the magnitude, but not the direction, of $\mathbf{h}$, consistent with spatial isotropy. The width of the distribution is $\sigma = l\sqrt{n/3}$, just as we would expect for a random walk. It is also peaked at $\mathbf{h} = 0$, indicating that the chain returns to the origin. That might seem peculiar, and will be discussed more below; however, we can confirm this with another simple calculation. From the definition of the end-to-end vector (Eq. 2.1) we find

$$\begin{aligned} \langle \mathbf{h} \rangle &= \langle \mathbf{r}_N - \mathbf{r}_1 \rangle \\ &= \langle \sum_{i=1}^{N-1} \mathbf{l}_i \rangle \\ &= \sum_{i=1}^{n} \langle \mathbf{l}_i \rangle \\ &= 0 . \end{aligned}$$

Finally, there is a non-zero probability of finding a chain with length greater than nl. This is obviously unphysical and results from approximating a multinomial distribution with a Gaussian.

Problem 2.6. *A random walk with step size l in 3D can be thought of as 3 random walks of step size $\frac{l}{\sqrt{3}}$ in 1D. Show that this scaling is expected by calculating the average projection of a random 3-dimensional unit vector onto a single axis. (Hint: use spherical coordinates and the z axis)*

2.3.4 End-to-end distance revisited

So what about that FJC returning to the origin? Does this mean that we expect the distance between ends to be zero? Not at all. We might reason that the probability for finding a particular $\mathbf{h}$ vector will decrease with its magnitude. If we consider a sphere of radius h centered at $\mathbf{r}_1$ (which we will also take as the origin), then there is only one vector $\mathbf{h}$ that corresponds to zero distance between the chain ends. As h increases, the allowed area onto which the second chain end may fall increases, so the probability of a specific $\mathbf{h}$ vector for a given h decreases. More formally, we can say that the probability $P(n,h)$ of finding an end-to-end *distance* h is just the sum of all probabilities of finding end-to-end *vectors* $\mathbf{h}$ with magnitude $|\mathbf{h}| = h$. This sum is the integral over the surface of the sphere with radius h:

$$\begin{aligned} P(n,h) &= \int_0^{2\pi} \mathrm{d}\phi \int_0^{\pi} \mathrm{d}\theta \; r^2 \sin\theta P(n, \mathbf{h} : |\mathbf{h}| = r) \\ &= 4\pi h^2 \left[\frac{3}{2\pi n l^2}\right]^{3/2} \exp\left\{-\frac{3h^2}{2nl^2}\right\} \; . \end{aligned} \tag{2.13}$$

Note that Eq. 2.13 is already normalized:

$$\int_0^{\infty} \mathrm{d}h \; P(n,h) = 1 \; .$$

Eq. 2.13 is the familiar Maxwell-Boltzmann distribution for particle speeds in a gas. Differentiation yields a maximum probability at $h = \sqrt{2nl^2/3}$.

So what is $\langle h^2 \rangle$? We find it by integrating, as usual:

$$\begin{aligned} \langle h^2 \rangle &= \int \mathrm{d}r \; r^2 P(n,r) \\ &= 4\pi h^2 \left[\frac{3}{2\pi n l^2}\right]^{3/2} \int_0^{\infty} \mathrm{d}r \; r^4 \exp\left\{-\frac{3r^2}{2nl^2}\right\} \\ &= nl^2 \; , \end{aligned} \tag{2.14}$$

where we have made use of the well-known result $\int_0^{\infty} \mathrm{d}x \; x^4 \mathrm{e}^{-\alpha x^2} = \frac{3}{8\alpha^2}\sqrt{\frac{\pi}{\alpha}}$. Eq. 2.14 is exactly the result that we expected: The distance of the walk increases with the square root of the number of steps, as we found in Eq. 2.9.

2.3.5 Elasticity in FJC

We can use our knowledge of distributions of the FJC to calculate its mechanical properties. Let us start by considering the free energy of the FJC. Recall

$$F = \langle E \rangle - TS$$

The $\langle E \rangle$ term is zero because there is neither a potential or motion in the FJC. Were we to assume that the chain can move, the lack of self-interaction would make the kinetic energy equal for all configurations, and $\langle E \rangle = 0$ still. We are left with

$$F = -TS \ .$$

The entropy for a chain of N beads with end-to-end vector $\mathbf{h}$ is

$$S(N, \mathbf{h}) = k_B \ln W(N, \mathbf{h})$$

where $W(N, \mathbf{h})$ is the number of conformations of chains of length N that have end-to-end vector $\mathbf{h}$. This is related to the probability of $\mathbf{h}$ (Eq. 2.12) by $W(N, \mathbf{h}) = P(N, \mathbf{h})W(N)$, where $W(N)$ is the total number of configurations of the freely jointed chain of N beads (i.e., considering all end-to-end vectors). We can then write the free energy as

$$\begin{aligned} F(N, \mathbf{h}) &= -k_B T \ln P(N, \mathbf{h}) - T \ln W(N) \\ &= \frac{3k_B T h^2}{2Nl^2} - \frac{3T}{2} \ln \left[\frac{3W(N)^{2/3}}{2\pi N l^2} \right] \ . \end{aligned} \tag{2.15}$$

Only the first term depends on the chain conformation; the second term is a constant of the system and can essentially be ignored. In fact, as we're dealing with free energies, it is really differences that we are interested in, so this constant term will explicitly drop out soon.

One thing that we see from Eq. 2.15 is that the minimum free energy occurs for $\mathbf{h} = 0$. We will use this as a reference point and ask how the free energy changes as we pull one end of the chain away from the other along some arbitrary vector $\mathbf{h}$. We have

$$\begin{aligned} \Delta F(N, \mathbf{h}) &\equiv F(N, \mathbf{h}) - F(N, 0) \\ &= -k_B T \ln \left[\frac{P(N, \mathbf{h})}{P(N, 0)} \right] \\ &= \frac{3k_B T h^2}{2nl^2} \ . \end{aligned} \tag{2.16}$$

Interestingly, the energy increases harmonically, just as if we were pulling a spring. More interesting still is that this effect arises solely from entropic considerations. Comparing Eq. 2.16 with Hooke's Law ($E = \frac{1}{2}k(\Delta x)^2$), we find that the effective force constant for the FJC is $k = 3k_B T/nl^2$. Thus, the restoring force that pulls the chain ends together increases with temperature but decreases with chain length.

We can extend our analysis even further by asking how the restoring force is expected to depend on the spatial separation between beads. From Eq. 2.9, we can see (if N is large so that $N \approx N - 1$) that $k = 3k_B T/\langle h^2 \rangle$. This result shows us that the restoring force itself follows an inverse-square law, so that if we know the distance between chain ends, we can guess how strong the restoring force is if we pull the ends apart. If we consider replacing N with the chemical distance $|j - i|$ between two arbitrary beads, we are very close to having ourselves an elastic network model. That is the topic of another lecture, but the important point is that a very – almost stupidly – simple model of a chain can yield useful results.

Problem 2.7. *Consider two freely jointed chains, A and B. The ends of chain A, having N_A residues, are separated by a distance of r_0 when pulled apart by a force of magnitude F. What is the expected separation between the end of chain B, having $N_B = 2N_A$ residues, when pulled apart by a force of the same magnitude?*

2.3.6 How does FJC compare to proteins?

This model does not represent proteins well. We find that for globular proteins (see Fig. 2.2), $R_g \sim N^{0.380}$.

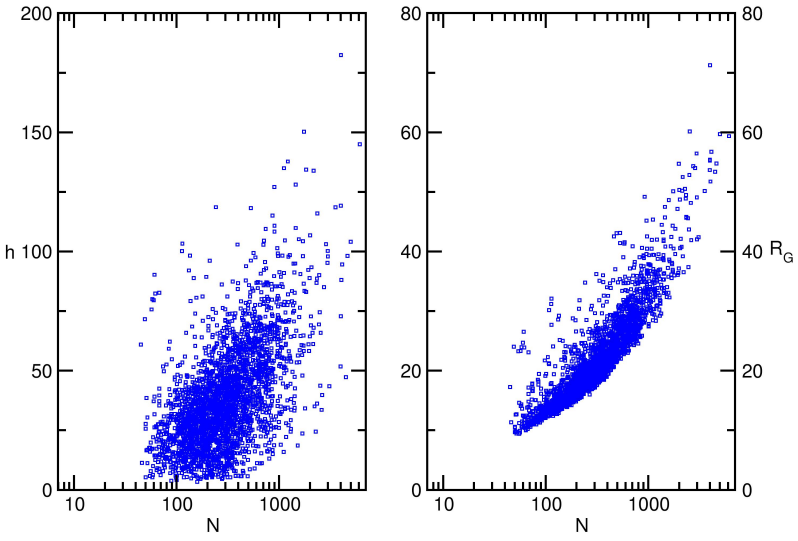


Figure 2.2: End-to-end distance and gyration radius for 2674 non-homologous globular proteins.

Among the many differences between proteins and the FJC are

- Proteins have restricted bond angles
- Proteins have side chains
- Proteins cannot self-intersect
- Proteins are solvated
- Protein residue-residue interactions have non-zero potential energies

We will look next at how fixed bond angles alter the properties of the model chain. The other issues are difficult to address and will have to wait.

2.4 Freely rotating chain

The FJC is truly a minimalist representation of reality, but it's a start. Perhaps the simplest bit of complexity that we can tack on to it is a constraint on bond angles. Assume that we have a chain in which all bond angles have the fixed value θ. This is called the *Freely Rotating Chain*. The methods developed here can be generalized to more realistic systems, such a proteins, in which the bond angles are not all identical, but we will start with the simplest case. The convention that will be used is that the bond angle at bead i satisfies

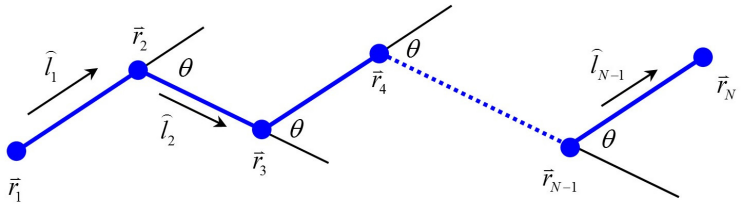


Figure 2.3: The freely jointed chain, wherein all bond angles are identical.

(see Fig. 2.3)

$$\mathbf{l}_{i-1} \cdot \mathbf{l}_i = |\mathbf{l}_i||\mathbf{l}_{i+1}| \cos\theta_i \ .$$

The $N-2$ bond angles for a chain of N beads have indices $2 \ldots (N-1)$. Let's start by considering h for the FRC, and let us assume that all links have the same length and all bonds have the same angle. That is, $|\mathbf{l}_i| = l, \theta_i = \theta \ \forall \ i$. The second term in Eq. 2.3, $\sum_i \sum_j \langle \mathbf{l}_i \cdot \mathbf{l}_j \rangle$, does not vanish in this case; however, we can find it by recursion. By definition,

$$\mathbf{l}_i \cdot \mathbf{l}_{i+1} = l^2 \cos\theta \ .$$

Then,

$$\begin{aligned} \langle \mathbf{l}_i \cdot \mathbf{l}_{i+2} \rangle &= \mathbf{l}_i \cdot (\hat{\mathbf{l}}_{i+1} \cdot \mathbf{l}_{i+2}) \hat{\mathbf{l}}_{i+1} \\ &= l^2 \cos^2\theta \ , \end{aligned}$$

and in general

$$\langle \mathbf{l}_i \cdot \mathbf{l}_{i+k} \rangle = l^2 \cos^k\theta \ .$$

In the above, hats represent unit vectors. One can show that all terms orthogonal to the links average to zero owing to the freedom of the dihedral angles. We won't do that here. Going back to Eq. 2.3, we find

$$\begin{aligned} h^2 &= nl^2 + \sum_{i=1}^{n} \sum_{j \neq i} \langle \mathbf{l}_i \cdot \mathbf{l}_j \rangle \qquad (2.17) \\ &= nl^2 + \sum_{i=1}^{n} \sum_{j \neq i} l^2 \cos^{|j-i|}\theta \\ &= nl^2 + 2 \sum_{i=1}^{n} \sum_{j > i} l^2 \cos^{|j-i|}\theta \end{aligned}$$

The second term is just l^2 times a sum of powers of $\cos\theta$. By restricting the sum to $j > i$, we are effectively looking at the upper triangle of the matrix of inner products of link vectors. The number of times that the k^{th} power of $\cos\theta$ appears is equal to the number of elements in the k^{th} diagonal above the main diagonal of the $n \times n$ matrix. Thus,

$$\begin{aligned}\sum_{i=1}^{n}\sum_{j\neq i}\langle \mathbf{l}_i \cdot \mathbf{l}_j\rangle &= 2\sum_{k=1}^{n-1}(n-k)l^2\cos^k\theta \\ &= 2nl^2\sum_{k=1}^{n-1}\cos^k\theta - 2l^2\sum_{k=1}^{n-1}k\cos^k\theta \\ &\approx 2l^2\cos\theta\left[\frac{n}{1-\cos\theta} - \frac{1}{(1-\cos\theta)^2}\right],\end{aligned} \tag{2.18}$$

where we have made use of the identities

$$\begin{aligned}\sum_{k=1}^{n}x^k &= \frac{x(1-x^n)}{1-x} \\ &\approx \frac{x}{1-x} \\ \sum_{k=1}^{n}kx^k &= \frac{x(1-x^{n+1})}{(1-x)^2} \\ &\approx \frac{x}{(1-x)^2}\end{aligned}$$

Then,

$$\begin{aligned}h^2 &= nl^2\left[\frac{1+\cos\theta}{1-\cos\theta} - \frac{2\cos\theta}{n(1-\cos\theta)^2}\right] \\ &\approx nl^2\left[\frac{1+\cos\theta}{1-\cos\theta}\right]\end{aligned}$$

Note that $\langle h^2\rangle \sim nl^2$, just like in the FJC; however, there is now an additional term of $(1+\cos\theta)/(1-\cos\theta)$, called the *stiffness*. For $\theta = \pi/2$, the result is exactly the same as the FJC. For $\theta < \pi/2$, $\langle h^2\rangle > Nl^2$, and the chain is more extended than the FJC. For $\theta > \pi/2$, $\langle h^2\rangle < nl^2$, and the chain is compact. At $\theta = 0$, the chain *should be* ballistic (i.e., $h^2 = n^2l^2$). Instead, we find that it has infinite length. This is a fault of the large n approximations that we have made. Returning to the last exact expression, Eq. 2.18, it can be seen that the chain is indeed ballistic for $\theta = 0$.

Problem 2.8. *Calculate $\langle h^2\rangle$ for a freely rotating chain that has alternating bond angles θ_A and θ_B, and alternating bond lengths l_A and l_B.*

2.4.1 Characteristic ratio

We have seen that for the FJC, $h^2 = nl^2$, whereas for the FRC, $h^2 = nl^2(1+\cos\theta)/(1-\cos\theta)$. Although we don't see this explicitly here, for a general chain molecule one might expect the *measured* square end-to-end distance, $\langle h^2\rangle_0$, (note the subscript) to scale as $\langle h^2\rangle_0 = C_n nl^2$. In the limit $n \to \infty$, we find for the FRC $C_\infty = (1+\cos\theta)/(1-\cos\theta)$. The constant C_∞ is called the *characteristic ratio*, and it can be experimentally determined for a variety of chain molecules. For carbon-based polymers, $4 \leq C_\infty \leq 12$, in general. By defining an *effective length* of $l_{eff} \equiv \sqrt{C_\infty} l$, we see that $h^2 = nl_{eff}^2$. That is, as far as the end-to-end distance is concerned, the FRC (and many other simple models) behave as a FJC with effective link length l_{eff}.

2.4.2 FRC: Persistence length

Going back to Eq. 2.7, it can be seen that the persistence length of the FRC is

$$\begin{aligned}\xi_p &= \frac{l}{2}(C_\infty + 1) \\ &\approx \frac{C_\infty l}{2} \quad \text{for } l \to 0\end{aligned}$$

Or, starting from Eq. 2.6, it can be seen for the FRC:

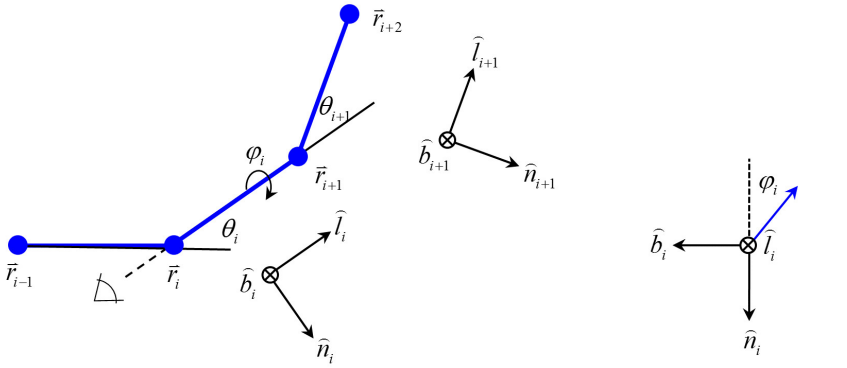
$$\begin{aligned}\xi_p &= \frac{1}{l}\sum_{j=i}^{n}\langle \mathbf{l}_i \cdot \mathbf{l}_j\rangle \\ &= \frac{1}{l}\sum_{j=1}^{n} l^2 \cos^{j-1}\theta \\ &= l\sum_{j=0}^{n-1}\cos^j\theta \\ &= l\left[\cos^0\theta + \sum_{j=1}^{n-1}\cos^j\theta\right] \\ &= l\left[1 + \frac{\cos\theta(1-\cos^{n-1}\theta)}{1-\cos\theta}\right] \\ &= l\left[\frac{1-\cos^n\theta}{1-\cos\theta}\right] \\ &\approx \frac{l}{1-\cos\theta}\end{aligned}$$

where the final approximation is taken in the limit $n \to \infty$. We can see that when $\theta = 0$, ξ_p is infinite, which is in accord with what we would expect for a straight chain of infinitely many links. When $\theta = \pi/2$, $\xi_p = l$, once again agreeing with the FJC.

2.5 Local coordinates

In general, chain molecules are not as simple as the FJC or FRC. Usually bond lengths and angles can vary in some range, as can dihedral angles. Frequently these values are correlated, as is the case for the ϕ and ψ backbone dihedrals in proteins. We can address the general case using matrices.

Suppose that we have a chain of N beads $[1 \ldots N]$. There are $n = (N-1)$ links $[1, \ldots, (N-1)]$, $N-2$ bond angles $[2, \ldots, (N-1)]$, and $N-3$ dihedral angles $[2, \ldots, (N-2)]$. With the exception of the dihedral angles, all of these have been previously defined. The dihedral ϕ_i is the clockwise rotation about link i from the *cis* conformation, and will be defined mathematically below.



(a) Local coordinate systems (b) View along $\hat{l}_i$.

Figure 2.4: Local coordinates defined using backbone conformation.

For each internal bead i, define a local orthogonal coordinate basis $\hat{\mathbf{l}}_i, \hat{\mathbf{b}}_i, \hat{\mathbf{n}}_i$ as follows:

$$\hat{\mathbf{l}}_i \equiv \frac{\mathbf{r}_{i+1} - \mathbf{r}_i}{|\mathbf{r}_{i+1} - \mathbf{r}_i|}$$
$$\hat{\mathbf{b}}_i \equiv \frac{\hat{\mathbf{l}}_i \times \hat{\mathbf{l}}_{i-1}}{\sin\theta_i}$$
$$\hat{\mathbf{n}}_i \equiv \hat{\mathbf{b}}_i \times \hat{\mathbf{l}}_i$$

The selection of coordinates is not unique; there are other possibilities that are equally valid, such as that discussed by Flory [1]. The coordinates here have the following interpretation: $\hat{\mathbf{l}}_i$ is the unit link vector that we have been using all along; $\hat{\mathbf{b}}_i$ is a *binormal* vector that defines the plane containing beads $i-1$, i and $i+1$; $\hat{\mathbf{n}}_i$ is normal to the curve and perpendicular to both $\hat{\mathbf{l}}_i$ and $\hat{\mathbf{b}}_i$. The three form a right-handed coordinate system at bead i: $\hat{\mathbf{l}}_i \times \hat{\mathbf{n}}_i = \hat{\mathbf{b}}_i$.

Nicely, we can construct a recursion relation to form coordinate systems from earlier systems:

$$\begin{aligned}
\hat{\mathbf{l}}_{i+1} &= \cos\theta_{i+1}\hat{\mathbf{l}}_i + \sin\theta_{i+1}[-\cos\phi_i\hat{\mathbf{n}}_i - \sin\phi_i\hat{\mathbf{b}}_i] \\
\hat{\mathbf{b}}_{i+1} &= \frac{\hat{\mathbf{l}}_i \times \hat{\mathbf{l}}_{i-1}}{\sin\theta_i} \\
&= \frac{1}{\sin\theta_{i+1}} \begin{vmatrix} \hat{\mathbf{l}}_i & \hat{\mathbf{n}}_i & \hat{\mathbf{b}}_i \\ \cos\theta_{i+1} & -\sin\theta_{i+1}\cos\phi_i & -\sin\theta_{i+1}\sin\phi_i \\ 1 & 0 & 0 \end{vmatrix} \\
&= -\sin\phi_i\hat{\mathbf{n}}_i + \cos\phi_i\hat{\mathbf{b}}_i \\
\hat{\mathbf{n}}_{i+1} &= \hat{\mathbf{b}}_i \times \hat{\mathbf{l}}_i \\
&= \begin{vmatrix} \hat{\mathbf{l}}_i & \hat{\mathbf{n}}_i & \hat{\mathbf{b}}_i \\ 0 & -\sin\phi_i & \cos\phi_i \\ \cos\theta_{i+1} & -\sin\theta_{i+1}\cos\phi_i & -\sin\theta_{i+1}\sin\phi_i \end{vmatrix} \\
&= \sin\theta_{i+1}\hat{\mathbf{l}}_i + \cos\theta_{i+1}\cos\phi_i\hat{\mathbf{n}}_i + \cos\theta_{i+1}\sin\phi_i\hat{\mathbf{b}}_i
\end{aligned}$$

Or,

$$\begin{pmatrix} \hat{\mathbf{l}}_{i+1} \\ \hat{\mathbf{n}}_{i+1} \\ \hat{\mathbf{b}}_{i+1} \end{pmatrix} = \begin{pmatrix} \cos\theta_{i+1} & -\sin\theta_{i+1}\cos\phi_i & \sin\theta_{i+1}\sin\phi_i \\ \sin\theta_{i+1} & \cos\theta_{i+1}\cos\phi_i & \cos\theta_{i+1}\sin\phi_i \\ 0 & -\sin\phi_i & \cos\phi_i \end{pmatrix} \begin{pmatrix} \hat{\mathbf{l}}_i \\ \hat{\mathbf{n}}_i \\ \hat{\mathbf{b}}_i \end{pmatrix} \tag{2.19}$$

The matrix in Eq. 2.19 is a transformation matrix between the local coordinate system at i and that at $i+1$. It is an orthogonal matrix, so its transpose,

$$\mathbf{T}_{i+1} = \begin{pmatrix} \cos\theta_{i+1} & \sin\theta_{i+1} & 0 \\ -\sin\theta_{i+1}\cos\phi_i & \cos\theta_{i+1}\cos\phi_i & -\sin\phi_i \\ \sin\theta_{i+1}\sin\phi_i & \cos\theta_{i+1}\sin\phi_i & \cos\phi_i \end{pmatrix}$$

is also its inverse. Multiplying both sides of Eq. 2.19 by $\mathbf{T}_{i+1}$ gives the coordinates of $\hat{\mathbf{l}}_{i+1}, \hat{\mathbf{b}}_{i+1}, \hat{\mathbf{n}}_{i+1}$ in the basis of $\hat{\mathbf{l}}_i, \hat{\mathbf{b}}_i, \hat{\mathbf{n}}_i$:

$$\begin{pmatrix} \hat{\mathbf{l}}_i \\ \hat{\mathbf{n}}_i \\ \hat{\mathbf{b}}_i \end{pmatrix} = \mathbf{T}_{i+1} \begin{pmatrix} \hat{\mathbf{l}}_{i+1} \\ \hat{\mathbf{n}}_{i+1} \\ \hat{\mathbf{b}}_{i+1} \end{pmatrix}$$

Or, in general,

$$\begin{pmatrix} \hat{\mathbf{l}}_i \\ \hat{\mathbf{n}}_i \\ \hat{\mathbf{b}}_i \end{pmatrix} = \left(\prod_{j=1}^{k} \mathbf{T}_{i+j} \right) \begin{pmatrix} \hat{\mathbf{l}}_{i+k} \\ \hat{\mathbf{n}}_{i+k} \\ \hat{\mathbf{b}}_{i+k} \end{pmatrix}$$

Note that $\left(\prod_{j=1}^{k} \mathbf{T}_{i+j}\right)$ is itself a matrix of inner products:

$$\left(\prod_{j=1}^{k} \mathbf{T}_{i+j} \right) = \begin{pmatrix} \hat{\mathbf{l}}_i \cdot \hat{\mathbf{l}}_k & \hat{\mathbf{l}}_i \cdot \hat{\mathbf{n}}_k & \hat{\mathbf{l}}_i \cdot \hat{\mathbf{b}}_k \\ \hat{\mathbf{n}}_i \cdot \hat{\mathbf{l}}_k & \hat{\mathbf{n}}_i \cdot \hat{\mathbf{n}}_k & \hat{\mathbf{n}}_i \cdot \hat{\mathbf{b}}_k \\ \hat{\mathbf{b}}_i \cdot \hat{\mathbf{l}}_k & \hat{\mathbf{b}}_i \cdot \hat{\mathbf{n}}_k & \hat{\mathbf{b}}_i \cdot \hat{\mathbf{b}}_k \end{pmatrix}$$

All of the above analysis is applicable only to chains with fixed conformations. Usually we will be interested in models in which the chain is free to move within some constraints, as in the FJC and FRC. In such cases, the matrices $\mathbf{T}_i$ must be replaced by their ensemble averages, $\langle \mathbf{T}_i \rangle$.

Now consider as an example the FRC with the special case that $\theta_i = \theta \; \forall \; i$. All internal beads are under identical conditions, and $\mathbf{T}_i$ is the same for all i. Further, $\langle \mathbf{T}_i \rangle$ is the same for all i, so the transformation matrix between i and j is just $\langle \mathbf{T} \rangle^{j-i}$.

To calculate $\langle \mathbf{T} \rangle$ for the FRC, we let ϕ vary freely:

$$\langle \mathbf{T} \rangle = \frac{\int_0^{2\pi} \mathrm{d}\phi \mathbf{T}}{\int_0^{2\pi} \mathrm{d}\phi}$$

yielding

$$\langle \mathbf{T} \rangle = \begin{pmatrix} \cos\theta & \sin\theta & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

and

$$\langle \mathbf{T}^k \rangle = \begin{pmatrix} \cos^k\theta & \cos^{k-1}\theta \sin\theta & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Problem 2.9. *Consider a random chain of N beads on a 2D square lattice with spacing l. Beads can only be placed on lattice sites, and sequential beads must be on neighboring sites (i.e., the distance between beads i and $i+1$ is always l). Assume that the chain is allowed to self-intersect.*

a. Find expressions for the squared end-to-end distance, h^2, and the radius of gyration, R_g, under the assumption that bead $i+1$ can occupy any of the four lattice sites adjacent to bead i with equal probability (i.e., the chain is a random walk on a 2D lattice). It may be helpful to follow the approach taken for the Freely Rotating Chain, and to make use of the geometric series in the notes.

b. Find expressions for the same values for a chain that cannot turn back on itself. In this case, bead $i+1$ can occupy any of the three lattice sites that are adjacent to bead i and are not occupied by bead $i-1$.

Problem 2.10. *Find an expression for the radius of gyration, R_g, for a chain on a 2D square lattice. Assume that the chain can self-intersect.*

2.6 Wormlike chain

In many cases, when $n \gg l$, it becomes nonsensical to sum over n elements. Porod and Kratky [5] introduced the idea of taking the continuum limit of a discrete chain by letting $n \to \infty$ and $l \to 0$. The chain length, $L = nl$, remains constant, and the chain essentially becomes a smooth curve in space. The model of Kratky and Porod is often called the

"Wormlike Chain", although it has been pointed out that unlike a worm, the chain is not extensible.

Consider a FRC. We find that the persistence length is

$$\xi_p = \frac{l}{\cos\theta}$$

so

$$\cos\theta = 1 - l/\xi_p \ .$$

Let's look at h^2, starting from Eqs. 2.17 and 2.18:

$$\begin{aligned} h^2 &= nl^2 + 2nl^2 \sum_{k=1}^{n-1} \cos^k\theta - 2l^2 \sum_{k=1}^{n-1} k\cos^k\theta \\ &= nl^2 + 2nl^2 \left(\frac{\cos\theta}{1-\cos\theta}\right) - 2l^2 \left(\frac{1-\cos^n\theta}{(1-\cos\theta)^2}\right) \\ &= nl^2 \left(\frac{2-l/\xi_p}{l/\xi_p}\right) - 2l^2(1-l/\xi_p)\left(\frac{1-(1-l/\xi_p)^n}{(l/\xi_p)^2}\right) \\ &= nl\xi_p(2-l/\xi_p) - 2\xi_p^2(1-l/\xi_p)(1-\exp\{-nl/\xi_p\}) \ , \end{aligned}$$

where we have used the approximation $\mathrm{e}^{-nl/\xi_p} \approx 1 - nl/\xi_p$ in the last line. This holds as long as $nl \gg \xi_p$, which is the case if our chain is long and thin. Defining $L \equiv nl$ and taking the small l limit,

$$\lim_{n\to 0} h^2 = 2L\xi_p - 2\xi_p^2(1-\mathrm{e}^{-L/\xi_p}) \ . \tag{2.20}$$

This final result demonstrates that the end-to-end distance can be found without knowledge of microscopic details of the chain (n,l,θ): Only ξ_p and the total length are needed. This is an early hint at universality, or scale invariance, in chains. By defining an overall scaling factor $a \equiv L/\xi_p$,

$$\begin{aligned} h^2 &= 2a\xi_p^2 - 2\xi_p^2(1-\mathrm{e}^{-a}) \\ &= 2\xi_p^2(a-1-\mathrm{e}^{-a}) \end{aligned}$$

Even though the end-to-end distance scales with length, it is qualitatively the same for different chains with the same $L : \xi_p$ ratio.

Problem 2.11. *Show that the expression for the mean square end-to-end distance of a wormlike chain (Eq. 2.20) reduces to the expected answers for a random coil and for a rigid rod, respectively, in the limits $L \gg \xi_p$ and $L \ll \xi_p$.*

2.6.1 Bending of WLC

To go from a discrete chain model to a continuous chain, consider a chain of equal length links bent at an angle θ_i at bead i. If the chain has the tendency to remain straight, we might impose upon it a harmonic potential that penalizes deflections from $\theta_i = 0$, such as

$$U_i = \frac{k}{2}\theta_i^2 \ .$$

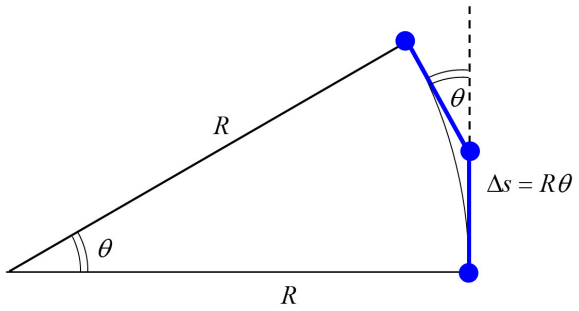


Figure 2.5: Bending a discrete chain with equally spaced beads. Bending the chain by an angle θ is equivalent to bending the chain around a radius $R \approx 2l/\theta$.

This is related to a harmonic spring restoring a linear displacement if we think of the displacement $\Delta s_i = R_i\theta_i$ as the length of the arc that connects bead $i-1$ to bead $i+1$ when bending the chain by θ_i about bead i produces an arc with radius R_i. It's safe to say that $\Delta s_i \approx 2l$, but here we are trying to avoid using l and θ_i in favor of the continuous variables R and s. Let us define s as a continuous variable representing the position along the chain. For the whole chain, the energy of bending is

$$\begin{aligned} U &= \frac{k}{2}\sum_{i=1}^{n-1}\theta_i^2 \\ &= \frac{k}{2}\sum_{i=1}^{n-1}\left(\frac{\Delta s}{R_i}\right)^2 \\ &= \frac{k\Delta s}{2}\sum_{i=1}^{n-1}\Delta s\left(\frac{1}{R_i}\right)^2 \\ &= \frac{K}{2}\int_0^L \mathrm{d}s\left(\frac{1}{R(s)}\right)^2 \\ &= \frac{K}{2}\int_0^L \mathrm{d}s\left|\frac{\mathrm{d}\mathbf{t}}{\mathrm{d}s}\right|^2, \end{aligned}$$

where $R(s)$ is the local curvature at s, $\mathbf{t}(s)$ is the tangent vector to the curve at s and K is the *bending modulus*. This is engineering here. Materials science stuff. The bending modulus has units of energy times distance.

Problem 2.12. *Follow the steps below to approximate the elastic energy that is stored in DNA packed into a viral capsid. The result from this approximation can be combined with an electrostatic calculation (not done here) to estimate the free energy cost of packing DNA into a capsid. Assume the capsid is a cylinder of radius R_{out} and height z, and that the DNA is a cylinder of diameter d_s and length L that behaves like a wormlike chain of persistence length $\xi_p > R_{out}$. Answers should be in terms of these values. Recall that the energy of bending a*

flexible beam of length l through a radius R is

$$U = \frac{Kl}{2R^2} \ , \tag{2.21}$$

where $K = \xi_p k_b T$ is the bending modulus. Assume that the DNA packs into well-ordered helices around the cylindrical axis, starting from the outer radius and progressing into helices of smaller and smaller radii.

a. Instead of helices, we can use the approximation that the DNA forms concentric circles within the capsid. Find an expression for the energetic cost, $U(R)$, of a circle of DNA with radius R.

b. As the DNA is packed tightly, we can use the approximation that inside the capsid it forms a uniformly dense cylindrical shell of inner radius R_{in} and outer radius R_{out}. Find an expression for R_{in} as a function of the known dimensions of the DNA and capsid.

c. The total elastic energy of the DNA is the sum over the energy of the rings:

$$U_{DNA} = \sum_{k=0}^{k_{max}} U(R_{in} + k\Delta R) N(R_{in} + k\Delta R) \tag{2.22}$$

where ΔR is the distance between successive rings, $N(R)$ is the number of rings with radius R, and k_{max} is simply $(R_{out} - R_{in})/\Delta R$, the number of steps to get to the capsid wall. When the DNA is packed tightly, each ring (except for those on the surface) will be touching six other rings. This type of packing, like a cells in a honeycomb or logs on a truck, is called hexagonal packing. Show that if the rings are packed tightly the distance between successive rings will be $\Delta R = \sqrt{3} d_s/2$.

d. Noting that there is one ring radius every ΔR, we can say that the density of radii is $1/\Delta R$. This can be used to convert the sum in Eq. 2.22 to an integral:

$$U_{DNA} \approx \frac{2}{\sqrt{3} d_s} \int_{R_{in}}^{R_{out}} \mathrm{d}R U(R) N(R) \ . \tag{2.23}$$

Note that

$$\frac{2}{\sqrt{3} d_s} \int_{R_{in}}^{R_{out}} \mathrm{d}R = k_{max} \ , \tag{2.24}$$

so $2/\sqrt{3} d_s$ is indeed a density. Assume that $N(R) = z/d_s$ for all R, and calculate U_{DNA} from Eq. 2.23.

e. Calculate the force $F(L) = -\mathrm{d}U_{DNA}/\mathrm{d}L$ required to put a length L of DNA into the capsid.

2.7 Self-avoiding walk

Another ingredient that has been missing from our models is self-avoidance. To get a sense of the conformational ensemble adopted by self-avoiding chains, we can follow Flory [1] and

construct some simple arguments based on scaling. We will minimize free energy with respect to the chain's size.

Start by introducing a potential that imposes a penalty for any contacting pairs:

$$U = \sum_{i=1}^{N} \sum_{j \neq i} \Theta(R_c - |\mathbf{r}_j - \mathbf{r}_i|) U_{ij} \ , \tag{2.25}$$

where $\Theta(x)$ is the Heavyside function, equal to 1 if its argument is positive and 1 otherwise. The potential of Eq. 2.25 contains one non-zero term for each contacting pair of beads. Assume that the density of contacts is more or less uniform throughout the volume occupied by the chain (i.e., $\rho = \#$ of contacts/volume is constant). The number of contacts scales with the square of the number of beads, and the volume scales as a power of the chain radius, depending on dimension. In two dimensions, the chain volume is approximated by the area of a circle: $V_2 = 2\pi R^2$. In three dimensions, it is approximated by the volume of a sphere: $V_3 = 4/3\pi R^3$. In d dimensions,

$$\rho \sim N^2/R^d \ .$$

The free energy is given by

$$F = U - TS \ ,$$

where $S = k_B \ln W$. We have seen from previous models that the chain parameters tend to follow Gaussian distributions. Here we will continue this approximation

$$\begin{aligned} S &= k_B \ln W \\ &= k_B \ln P(h^2) + \text{const.} \\ &\approx k_B \ln(\exp\{-h^2/N\}) \end{aligned}$$

Yielding

$$F = c_1 N^2/R^d - c_2 R^2/N$$

Minimizing with respect to R,

$$\begin{aligned} 0 &= \partial F/\partial R \\ &= -c_1 d N^2 R^{-d-1} - 2c_2 R N^{-1} \\ R &\sim N^{\frac{3}{d+2}} \ . \end{aligned} \tag{2.26}$$

Eq. 2.26 indicates that the end-to-end distance (or R_g) will scale with the chain dimension. In one dimension, we see that $R \sim N$, and the chain is ballistic. In two dimensions, $R \sim N^{3/4}$. This is an exact result. In three dimensions, $R \sim N^{3/5}$. Computational results indicate that the scaling in three dimensions is more accurately $R \sim N^{0.588}$, which is remarkably close to the value arrived at through this simple exercise. The scaling of $R \sim N^{1/2}$ for a four-dimensional chain is again exact. Beyond four dimensions, the self-avoidance causes attraction ($R < N^{1/2}$), indicating that this simple scaling is not valid in higher dimensions.

Bibliography

[1] Paul J. Flory. *Statistical Mechanics of Chain Molecules*. Hanser-Gardner, Cincinnati, Ohio, 1989.

[2] Ken A. Dill and Sarina Bromberg. *Molecular Driving Forces*. Garland Science, New York, NY, 2003.

[3] Rob Phillips, Jane Kondev, and Julie Theriot. *Physical Biology of the Cell*. Garland Science, New York, NY, 2009.

[4] Paul C. Hiemenz and Timothy P. Lodge. *Polymer Chemistry*. CRC Press, Boca Raton, FL, second edition, 2007.

[5] O. Kratky and G. Porod. Rontgenuntersuchung geloster fadenmolekule. *Recueil des Travaux Chimiques des Pays-Bas-Journal of the Royal Netherlands Chemical Society*, 68(12):1106–1122, 1949.

Chapter 3

Small Oscillations

An easy way to approach normal mode analysis is through the study of small oscillations. The ideas that are central to the theory of small oscillations are the same as those seen in equilibrium protein structural dynamics, but we can investigate small oscillations in systems of arbitrary size and shape. Material presented here has been discussed in greater detail in a number of mechanics textbooks, like Goldstein [1], Fetter and Walecka [2] and Fowles and Cassiday [3].

3.1 The one-body one-dimensional harmonic system

About the simplest oscillatory mechanical system is a particle connected to an immobile wall by means of a massless spring. We will say that the mass can only move in one dimension and apply the usual unrealistic approximations of no gravity and zero friction (Fig. 3.1). The particle has mass m and when the spring is in its equilibrium conformation, the mass is at position x^0. The spring has force constant k. If the mass moves to a new position, x, then the spring will push or pull on it with a force

$$F = -k(x - x^0) \ .$$

This is Hooke's law, and you've hopefully seen it before. For simplicity, we can define $\delta \equiv x - x^0$ as the displacement of the mass from its equilibrium position.

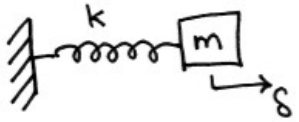


Figure 3.1: A mass m connected to an immobile wall via a massless spring of constant k.

It is worth pointing out that the potential energy of the system is

$$V = \frac{1}{2}k\delta^2 \ , \tag{3.1}$$

and that the restoring force is the negative of the derivative of the potential with respect to x: $F = -\partial V/\partial x$. The second derivative is just the force constant: $\partial^2 V/\partial x^2 = k$.

According to Newton, the force (because it is the only external force on the mass) causes the mass to accelerate back toward x^0. Specifically, Newton tells us

$$m\ddot{\delta} = -k\delta \ , \tag{3.2}$$

where $\ddot{\delta} = \frac{\mathrm{d}^2\delta}{\mathrm{d}t^2}$ is the second derivative of δ with respect to time. The solution is oscillatory, taking the general form

$$\delta(t) = \delta_+ \mathrm{e}^{\mathrm{i}\omega t} + \delta_- \mathrm{e}^{-\mathrm{i}\omega t} \ , \tag{3.3}$$

where δ_+ and δ_- are complex constants. This is the most general solution to Eq. 3.2, and it should appeal to mathematicians, but it contains a lot of complex numbers that are not useful in our analysis of real systems. We will instead work with the more specific and completely real solution

$$\delta(t) = A\sin(\omega t + \phi) \ , \tag{3.4}$$

where A is a constant specifying the amplitude of the motion, and ϕ is a phase. We know that $-1 \leq \sin(\omega t + \phi) \leq 1$, so the maximum displacement of the mass is A. The phase ϕ tells us when this maximum amplitude is reached. If $\phi = 0$, then at time $t = 0$ the mass is at the origin. If $\phi = \pm\pi/2$, then at time $t = 0$ the mass is a distance A from the origin. When ϕ is somewhere in-between, then in-between behavior is expected.

Problem 3.1. *Show that Eqs. 3.3 and 3.4 are solutions of 3.2.*

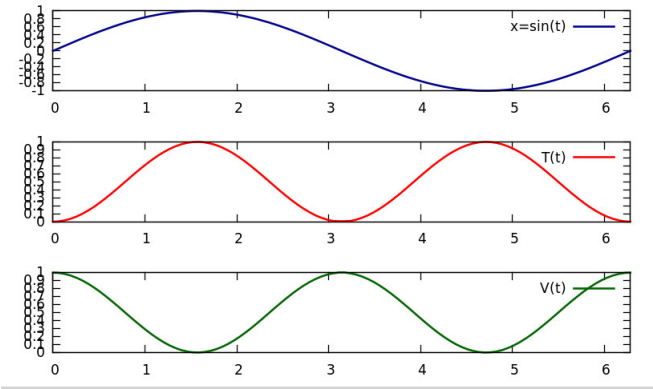


Figure 3.2: Harmonic motion. The top panel shows oscillatory position as a function of time, and the two lower panels show kinetic and potential energy in arbitrary units.

The total energy E is the sum of the potential energy (Eq. 3.1) and the kinetic energy $T = \frac{1}{2}m\dot{x}^2$. Substituting $m\omega^2$ for k gives the energy in terms of frequency and amplitude,

$$E = \frac{m\omega^2 A^2}{2} \ . \tag{3.5}$$

Because all the terms on the right-hand-side of Eq. 3.5 are constants, the total energy of the mass-spring system is conserved. We have given it no place to go in our toy system. A

more realistic model would include interactions with the environment that would allow the internal energy of the system to change over time, but not here. An oscillator that has total energy E will have amplitude $A = \sqrt{\frac{2E}{m\omega^2}}$ and a phase set by initial conditions. The energy itself oscillates between kinetic and potential (Fig. 3.2). When $\omega t + \phi = 0$, the potential is zero and $T = E$. When $\omega t + \phi = \pm\frac{\pi}{2}$, there is instantaneous velocity and $V = E$. The phase might then be thought of as controlling the mixture of kinetic to potential energy at a time t.

Problem 3.2. *Find an expression for $\rho(x)$, the probability of finding the mass at posistion x. Start by thinking of the probability $\rho(x)\mathrm{d}x$ as proportional to the amount of time that the particle spends in an interval of size $\mathrm{d}x$ centered at x.*

3.2 The many-body harmonic system

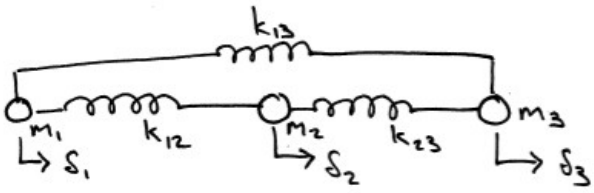


Figure 3.3: Three particles connected by springs.

Now consider a slightly less simple system of three masses constrained to one dimension and connected to each other via three springs, as shown in Fig. 3.3. We will say that all of the springs are happily at their rest lengths in the configuration shown. This three-particle system is just a simple example of the N-body system, and what follows can be applied to similar systems of any number of interacting particles. To see how the system moves, we can analyze each mass independently. The spring k_{12} exerts a rightward ($+x$) force on particle 1 if $\delta_2 > \delta_1$, it exerts a force to the left if $\delta_2 < \delta_1$, and it exerts no force if $\delta_2 = \delta_1$. A similar statement can be made for k_{13} and δ_3. The equation of motion of particle 1 is

$$m_1\ddot{\delta_1} = k_{12}(\delta_2 - \delta_1) + k_{13}(\delta_3 - \delta_1) \ .$$

For particle 2, spring k_{12} imposes a force that is equal and opposite the force it imposes on particle 1, and k_{23} imposes a force in the direction of $\delta_3 - \delta_2$:

$$m_2\ddot{\delta_2} = -k_{12}(\delta_2 - \delta_1) + k_{23}(\delta_3 - \delta_2) \ .$$

Finally, particle 3:

$$m_3\ddot{\delta_3} = -k_{13}(\delta_3 - \delta_1) - k_{23}(\delta_3 - \delta_2) \ .$$

The right-hand sides of these three equations can be rearranged to give the pretty form

$$\begin{aligned} m_1\ddot{\delta_1} &= -(k_{12}+k_{23})\delta_1 &&+ k_{12}\delta_2 &&+ k_{13}\delta_3 \\ m_2\ddot{\delta_2} &= k_{12}\delta_1 &&- (k_{12}+k_{23})\delta_2 &&+ k_{23}\delta_3 \\ m_3\ddot{\delta_3} &= k_{12}\delta_1 &&+ k_{23}\delta_2 &&- (k_{13}+k_{23})\delta_3 \ , \end{aligned}$$

which can be re-written using matrices:

$$\begin{pmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ 0 & 0 & m_3 \end{pmatrix} \begin{pmatrix} \ddot{\delta_1} \\ \ddot{\delta_2} \\ \ddot{\delta_3} \end{pmatrix} = - \begin{pmatrix} (k_{12}+k_{23}) & -k_{12} & -k_{13} \\ -k_{12} & (k_{12}+k_{23}) & -k_{23} \\ -k_{13} & -k_{23} & (k_{13}+k_{23}) \end{pmatrix} \begin{pmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \end{pmatrix} \ ,$$

or, in *really* compact notation,

$$\mathbf{M}\ddot{\boldsymbol{\delta}} = -\mathbf{K}\boldsymbol{\delta} \ . \tag{3.6}$$

The 3-component vector $\boldsymbol{\delta}$ contains the displacements of the 3 particles from their rest positions, and the matrix $\mathbf{M}$ is a diagonal matrix containing our particles' masses. It is appropriately referred to as the *mass matrix*. On the right-hand side is a matrix, $\mathbf{K}$, of force constants. It is real, symmetric and positive semi-definite, and each of its rows or columns sums to zero. It is a *Hessian* matrix, and warrants some discussion.

3.2.1 The Hessian matrix

The system's potential energy is just the sum of potentials from the individual springs:

$$V = \frac{k_{12}}{2}(\delta_2-\delta_1)^2 + \frac{k_{13}}{2}(\delta_3-\delta_1)^2 + \frac{k_{23}}{2}(\delta_3-\delta_2)^2 \ . \tag{3.7}$$

The student can verify that this is equivalent to

$$\begin{aligned} V &= \frac{1}{2}\begin{pmatrix} \delta_1 & \delta_2 & \delta_3 \end{pmatrix} \begin{pmatrix} (k_{12}+k_{23}) & -k_{12} & -k_{13} \\ -k_{12} & (k_{12}+k_{23}) & -k_{23} \\ -k_{13} & -k_{23} & (k_{13}+k_{23}) \end{pmatrix} \begin{pmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \end{pmatrix} \\ &= \frac{1}{2}\tilde{\boldsymbol{\delta}}\mathbf{K}\boldsymbol{\delta} \ . \end{aligned} \tag{3.8}$$

The potential of the system is a quadratic function of its coordinates, and the matrix $\mathbf{K}$ holds the coefficients. Its elements are the second partial derivatives of the potential with respect to the system's coordinates:

$$K_{ij} = \frac{\partial^2 V}{\partial\delta_i \partial\delta_j} \ . \tag{3.9}$$

Matrices like Eq. 3.9 are called Hessian matrices, and they are important in harmonic analysis. In the present simple case the Hessian is just a matrix of force constants, but when we look at more complicated systems in later chapters, the form of the Hessian will become increasingly complex.

Equation 3.6 looks very much like Eq. 3.2, but in bold face. We can try a solution that is similar to Eq. 3.4, but using a vector out in front rather than a scalar: $\boldsymbol{\delta}(t) = \boldsymbol{\delta}^0 \sin(\omega t + \phi)$. Then $\ddot{\boldsymbol{\delta}}(t) = -\omega^2 \boldsymbol{\delta}(t)$, and we have

$$\omega^2 \mathbf{M}\boldsymbol{\delta} = \mathbf{K}\boldsymbol{\delta} \tag{3.10}$$

It may be tempting to follow the 1D example of the previous section and eliminate the $\boldsymbol{\delta}$ from both sides of this equation, but that's not allowed in matrix algebra: Removing a variable from both sides of an equation is actually dividing each side by the variable, and there is no such thing as vector divsion. We could try multiplying by the inverse of $\boldsymbol{\delta}$, but we don't yet know it. We do know the inverse of $\mathbf{M}$, so we can multiply both sides of Eq. 3.10 by that, giving

$$\omega^2 \boldsymbol{\delta} = \mathbf{M}^{-1}\mathbf{K}\boldsymbol{\delta} \; .$$

In this equation, the scalar ω^2 multiplies a vector on the left-hand side, and the matrix $\mathbf{M}^{-1}\mathbf{K}$ multiplies the same vector on the right-hand side. It is an eigenvalue equation. But in multiplying through by $\mathbf{M}^{-1}$, we have merged the two symmetric matrices $\mathbf{M}$ and $\mathbf{K}$ into one non-symmetric matrix. We would like to preserve the symmetry, as we have seen that the eigenvectors of symmetric matrices can be easily interpreted as a special orthogonal basis. Thus, we need a way to get rid of $\mathbf{M}$ yet retain the symmetry of the problem.

Problem 3.3. *Show that* $\mathbf{K}$ *is obtained from Eqs. 3.7 and 3.9.*

Problem 3.4. *Show that the product of two symmetric matrices is not necessarily symmetric. What conditions must be satisfied for the product of two symmetric matrices to be symmetric?*

3.2.2 Mass weighting

For our second shot at Eq. 3.6, let's see what happens when we multiply by $\mathbf{M}^{-1/2}$ from the left:

$$\begin{aligned} \mathbf{M}^{-1/2}\mathbf{M}\ddot{\boldsymbol{\delta}} &= -\mathbf{M}^{-1/2}\mathbf{K}\boldsymbol{\delta} \\ \mathbf{M}^{1/2}\ddot{\boldsymbol{\delta}} &= -\mathbf{M}^{-1/2}\mathbf{K}\boldsymbol{\delta} \\ \ddot{\mathbf{q}} &= -\mathbf{M}^{-1/2}\mathbf{K}\mathbf{M}^{-1/2}\mathbf{q} \; . \end{aligned}$$

In the last step we introduced the *mass-weighted coordinates*, $\mathbf{q} \equiv \mathbf{M}^{1/2}\boldsymbol{\delta}$. Using these coordinates allows us to keep the Hessian matrix symmetric because it gets multiplied from both the left and the right by the same symmetric matrix. In the mass-weighted coordinate system, our equations of motion reduce to the symmetric eigenvalue equation

$$\omega^2 \mathbf{q} = \mathbf{K}'\mathbf{q} \; , \tag{3.11}$$

where $\mathbf{K}' = \mathbf{M}^{-1/2}\mathbf{K}\mathbf{M}^{-1/2}$ is the *mass-weighted Hessian matrix*. Whereas the Hessian matrix has units of force constant (e.g., kg/s^2), the mass-weighted Hessian has units of force constant divided by mass, or s^{-2}. The solutions of Eq. 3.11 are the eigenvectors of $\mathbf{K}'$, and

their corresponding eigenvalues are the squares of the vibrational frequencies. We can find the modes of motion by constructing the mass-weighted Hessian matrix, diagonalizing it, and then performing the inverse of the mass-weighting transformation.

If some vector $\mathbf{v}$ is a solution of Eq. 3.11, then the system satisfies the equation

$$\mathbf{v}(t) = A\mathbf{v}\sin(\omega t + \phi) \ ,$$

where A is an arbitrary amplitude. All particles in the system oscillate with the same frequency ω, and their relative amplitudes are given by the components of $\mathbf{v}$. As $\mathbf{v}$ contains mass-weighted displacements from the minimum-energy conformation, all components of $\mathbf{v}$ pass through zero simultaneously.

The amplitude A was included explicitly even though the magnitude of $\mathbf{v}$ is unspecified. Why not just wrap A up into $\mathbf{v}$? The answer is that, as a matter of convention and convenience, eigenvectors are generally assigned a magnitude of 1. They are unit vectors, pointing in the direction of instantaneous displacements from equilibrium. They describe not just position, but higher derivatives as well: $\mathbf{v}$ describes the direction of displacements, velocities and accelerations for the oscillating system.

The mass-weighted Hessian $\mathbf{K}'$ for a system having N degrees of freedom will be $N \times N$. We expect that there will be N solutions to Eq. 3.11, corresponding to the eigenvectors of $\mathbf{K}'$. Each solution describes a vibration of a particular frequency, ω, with the eigenvectors pointing in the direction of the vibration. Eigenvectors are frequently called the *modes* of motion: they are allowed and independent vibrational directions. The condition of orthogonality leads to the descriptive *normal modes*, meaning that the phase and amplitude of motion in one mode is independent of the phase and amplitude of motion in the other modes. As such, the modes are *uncoupled*: There is no energy transferred between modes, and once the system commences oscillating along one mode, it continues to do so indefinitely. This is, of course, totally unrealistic. It results from the assumptions that the particles interact only through their harmonic constraints (springs), and that they don't collide with each other or with their environment. Real systems are far more complex, but this provides a good place to start.

3.2.3 Finding the modes

Assume for the system above that all particles have equal mass ($m_1 = m_2 = m_3 = m$) and all force constants are equal ($k_{12} = k_{13} = k_{23} = k$). Equation 3.6 is then

$$m \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \ddot{\delta}_1 \\ \ddot{\delta}_2 \\ \ddot{\delta}_3 \end{pmatrix} = -k \begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix} \begin{pmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \end{pmatrix} \ ,$$

and the mass-weighted Hessian matrix is

$$\mathbf{K}' = \frac{k}{m} \begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix} \ .$$

It is diagonalized by solving

$$|\mathbf{K}' - \lambda\mathbf{1}| = 0 \ ,$$

which yields the characteristic polynomial

$$\lambda^3 - 6\lambda^2 + 9\lambda = 0 \ .$$

There are two solutions: $\lambda = 0$ and $\lambda = 3k/m$, which will be labeled λ_1 and λ_2, respectively. The eigenvalues have units of k/m, or $1/(time)^2$, just as should be expected for squared frequencies. The system oscillates along each mode, i, independently, with an oscillatory frequency of $\omega_i = \sqrt{\lambda_i}$.

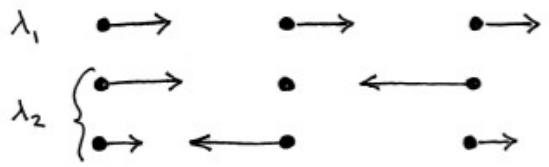


Figure 3.4: Modes of the 3-particle mass-spring system with equal masses and equal force constants.

Once we have a mode, $\mathbf{v}$, we can transform it back to the Cartesian coordinate system with the inverse transformation

$$\mathbf{u} = \mathbf{M}^{-1/2}\mathbf{v} \ .$$

Or, writing the mass-weighted modes as the columns of the matrix $\mathbf{V}$,

$$\mathbf{U} = \mathbf{M}^{-1/2}\mathbf{V} \ , \tag{3.12}$$

where $\mathbf{U}$ is the matrix of modes in the Cartesian coordinate system. Eq. 3.12 indicates that particles with large mass will have reduced motion in the Cartesian system, relative to their motions in the mass-weighted system. The orthonormality relationship $\tilde{\mathbf{V}}\mathbf{V} = \mathbf{1}$ can be rewritten to show that in the Cartesian coordinates,

$$\tilde{\mathbf{U}}\mathbf{M}\mathbf{U} = \mathbf{1} \ ,$$

so that the Cartesian modes are "normal" only when the mass matrix is sandwiched between them. This is an important point that can be easily overlooked. So-called *normal modes* are not necessarily normal. Most ENMs wave away this technicality by assuming that all amino acid residues have roughly equal mass, reducing the mass matrix to a multiple of the identity matrix. If all particle masses are equal, then the mass-weighted modes and the Cartesian modes are equivalent.

The eigenvector associated with $\lambda_1 = 0$ is $\mathbf{v}^{(1)} = \frac{1}{\sqrt{3}}[1\ 1\ 1]^T$, corresponding to simultaneous and equivalent translation of all three particles. The factor of $1/\sqrt{3}$ is in place to satisfy the normalization condition $\tilde{\mathbf{v}}\mathbf{v} = 1$. This vector changes the system's center of mass, but does not alter its internal energy, as the inter-particle distances remain fixed. There are no

oscillations along this mode, so its frequency is understandably zero. The mode stretches no springs and therefore has zero energy. It is quite common to find eigenvalues of zero in elastic network models, and they serve a diagnostic purpose. There will be one *zero mode* for each degree of freedom of the rigid system. In the present case, each of the particles has one degree of freedom, but if all particles are displaced equally along that degree, then there is no net change in their relative positions. Other examples will follow. Modes that move the system's center of mass should have zero eigenvalue, but not all modes with zero eigenvalue will move the system's center of mass.

The only other eigenvalue of $\mathbf{K}'$ is $\lambda_2 = 3k/m$, even though there are two more eigenvectors. Modes that share an eigenvalue are referred to as *degenerate* because they are not unique. Mode degeneracies arise from symmetries in the system. In the present example, the particle masses are equal and each particle has the same influence on the positions of the other particles. The result is a *twofold degeneracy* in the second mode (Fig. 3.5). The nature of the degeneracy can be intuited by exploring an example. A valid eigenvector with $\lambda_2 = 3k/m$ is $\mathbf{v}^{(2)} = \frac{1}{\sqrt{2}}[1\ 0\ -1]^T$, corresponding to motion of the outer points (1 and 3) in opposite directions and no motion of the central point (#2). Note that displacements along this mode do not alter the system's center of mass.

Problem 3.5. *Calculate the components of the mode* $\mathbf{v}'$ *in Fig. 3.5. Find another mode,* $\mathbf{v}''$*, that is orthogonal to and degenerate with* $\mathbf{v}'$ *. Sketch* $\mathbf{v}''$*.*

Defining $\mathbf{v}^{(2)}$ also specifies that $\mathbf{v}^{(3)} = \pm\frac{1}{\sqrt{6}}[1\ -2\ 1]^T$, because this is the unit vector that is orthogonal to both $\mathbf{v}^{(1)}$ and $\mathbf{v}^{(2)}$. It is easy to verify that $\mathbf{v}^{(3)}$ is an eigenvector of $\mathbf{K}'$ with eigenvalue λ_2. The eigenvectors $\mathbf{v}^{(2)}$ and $\mathbf{v}^{(3)}$ form an orthogonal basis over the space of modes with eigenvalue λ_2, but they are not the only acceptable choice. In fact, any linear combination of $\mathbf{v}^{(2)}$ and $\mathbf{v}^{(3)}$ is itself a valid mode with eigenvalue λ_2. For example, the columns of $\mathbf{K}'$ are all eigenvectors with eigenvalue λ_2, and they can all be expressed as linear combinations of $\mathbf{v}^{(2)}$ and $\mathbf{v}^{(3)}$.

Problem 3.6. *Find the general form of the frequencies and eigenmodes for the 1D 3-body system with masses* m_1*,* m_2 *and* m_3*, and with force constants* k_{12}*,* k_{13} *and* k_{23}*.*

Problem 3.7. *Suggest a simple modification to the example that will remove the degeneracy. Show that the resulting Hessian matrix has no degeneracies.*

3.2.4 Energy

Let us consider the specific case of the system oscillating along mode i alone, where $i \neq 0$ Its mass-weighted position will be given by

$$\mathbf{q}(t) = A_i \mathbf{v}^{(i)} \sin(\omega_i t + \phi_i) \ , \tag{3.13}$$

where $\omega_i = \sqrt{3k/m}$, A_i is an amplitude and ϕ_i a phase. The mass-weighted vector $\mathbf{q}$ must have dimensions of distance $\times \sqrt{\text{mass}}$, and we will assume these dimensions into the constant

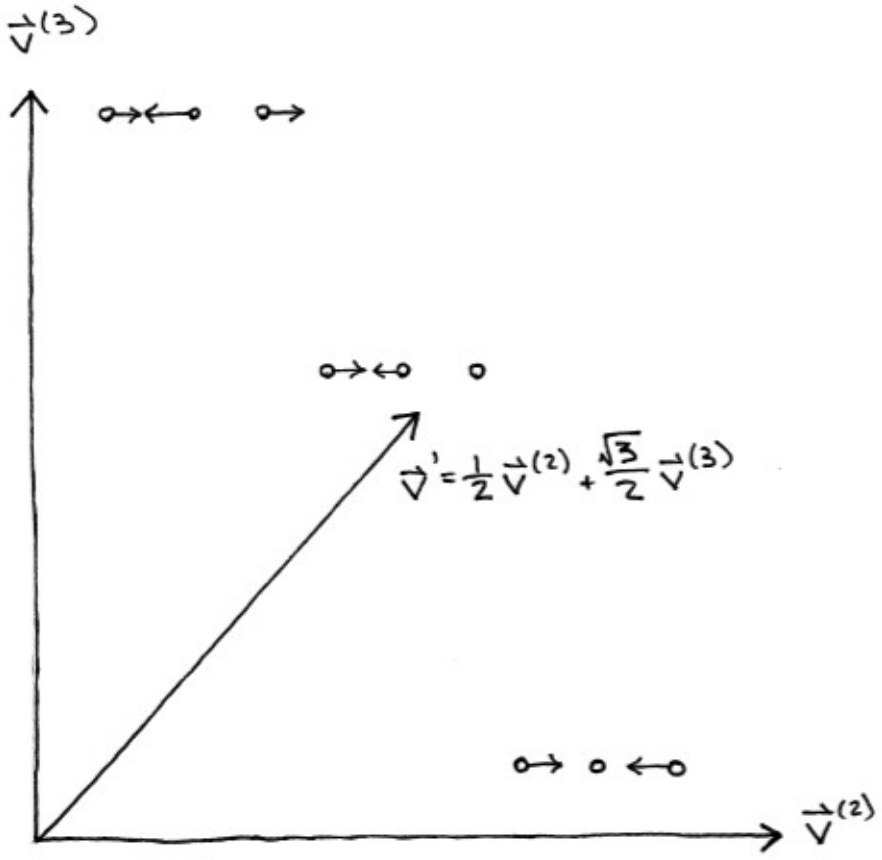


Figure 3.5: An illustration of degenerate modes for a simple 3-body system. The mode $\mathbf{v}'$ can be constructed as a linear combination of modes $\mathbf{v}^{(2)} = \frac{1}{\sqrt{2}}[1\ 0\ -1]^T$ and $\mathbf{v}^{(3)} = \frac{1}{\sqrt{6}}[1\ -2\ 1]^T$. The new mode $\mathbf{v}'$ eliminates motion of mass 3.

A_i, so that $\mathbf{v}^{(i)}$ does not carry dimensions. The potential energy at time t is given by Eq. 3.8,

$$\begin{aligned} V(t) &= \frac{1}{2}\tilde{\mathbf{q}}(t)\mathbf{K}'\mathbf{q}(t) \qquad (3.14)\\ &= \frac{1}{2}\left[A_i\tilde{\mathbf{v}}^{(i)}\sin(\omega_i t+\phi_i)\right]\mathbf{K}'\left[A_i\mathbf{v}^{(i)}\sin(\omega_i t+\phi_i)\right]\\ &= \frac{A_i^2}{2}\sin^2(\omega_i t+\phi_i)\,\tilde{\mathbf{v}}^{(i)}\mathbf{K}'\mathbf{v}^{(i)}\\ &= \frac{A_i^2\omega_i^2}{2}\,\sin^2(\omega_i t+\phi_i)\ , \end{aligned}$$

where we have used the fact that $\tilde{\mathbf{v}}^{(i)}\mathbf{K}'\mathbf{v}^{(i)} = \lambda_i$ in the last equation. The kinetic energy in mass-weighted coordinates is

$$\begin{aligned} T(t) &= \frac{1}{2}\widetilde{\dot{\mathbf{q}}}\dot{\mathbf{q}} \\ &= \frac{1}{2}\left[A_i\omega_i\tilde{\mathbf{v}}^{(i)}\cos(\omega_i t+\phi_i)\right]\left[A_i\omega_i\mathbf{v}^{(i)}\cos(\omega_i t+\phi_i)\right] \\ &= \frac{A_i^2\omega_i^2}{2}\cos^2(\omega_i t+\phi_i)\,\tilde{\mathbf{v}}^{(i)}\mathbf{v}^{(i)} \\ &= \frac{A_i^2\omega_i^2}{2}\,\cos^2(\omega_i t+\phi_i)\ . \end{aligned}$$

The total energy along mode i at time t is

$$\begin{aligned} E_i &= T+V \\ &= \frac{A_i^2\omega_i^2}{2}\,\sin^2(\omega_i t+\phi_i) + \frac{A_i^2\omega_i^2}{2}\,\cos^2(\omega_i t+\phi_i) \\ &= \frac{A_i^2\omega_i^2}{2}\ , \end{aligned} \tag{3.15}$$

which is a constant. The energy along any mode is constant, so *no energy is transferred between modes*. This is an important and unrealistic result of normal mode analysis, but one upon which we will rely to make mathematics easier. Note that by assuming mass into A_i we have lost the explicit m term from Eq. 3.5, and that the result is independent of the phase ϕ_i.

The orthogonality of mass-weighted modes provides us with the interesting mathematical result that any deformation of the system can be expressed as a combination of the modes. As there are N orthogonal modes in N dimensions, they form a complete basis over the system's degrees of freedom. Any vector in the mass-weighted coordinates is therefore expressible in exactly one way as a sum of the modes.

Suppose that we have a system that has N internal modes into which we put an amount of energy, E. Because there is no exchange of energy between modes, we know from Eq. 3.15 that

$$\begin{aligned} E &= \sum_{i=1}^{N} E_i \qquad (3.16) \\ &= \sum_{i=1}^{N}\frac{A_i^2\lambda_i}{2} \\ &= \sum_{i=1}^{N}\left(\frac{A_i}{\sqrt{2/\lambda_i}}\right)^2 \\ 1 &= \sum_{i=1}^{N}\left(\frac{A_i}{\sqrt{2E/\lambda_i}}\right)^2\ . \qquad (3.17) \end{aligned}$$

The last line may be recognized as the equation of an N-dimensional ellipse with radii $\sqrt{\frac{2E}{\lambda_i}}$. All possible dynamical states of this system with energy E fall on the isoenergetic hyperelliptical surface described by Eq. 3.17. Putting energy E into the system is equivalent to specifying a point on this surface, as well as the phase ϕ_i for each mode. One can verify that

$$\langle A_i^2 \rangle = \frac{E}{\lambda_i} \, ,$$

or the squared amplitude of motion increases linearly with energy, as would be expected for a harmonic system. It also decreases with mode frequency, showing us that low-frequency modes have larger expected amplitudes than their high-frequency counterparts.

Problem 3.8. *Show that all deformations in Cartesian coordinates can be expressed as sums of normal modes.*

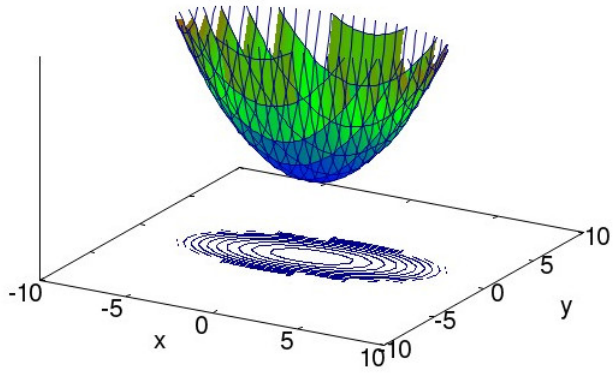
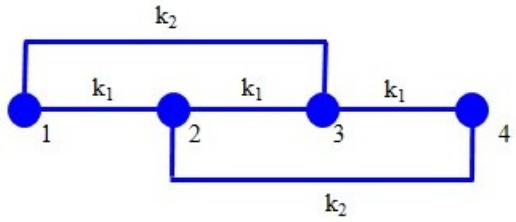


Figure 3.6: A 2D harmonic well describes the energy landscape associated with a 2D Gaussian probability distribution.

Problem 3.9. *Consider a 1 dimensional chain of four monomers as shown in the figure. Successive monomers are equally separated along the chain and connected to each other via springs with constants k_1 and k_2 as shown. Monomers 1 and 3 have mass m_1, and monomers 2 and 4 have mass m_2. In the configuration shown, all springs are at their rest length. a. Write down the mass matrix and Hessian matrix for the system.* **From this point onward, assume that all masses are equal (that is, $m_1 = m_2 = m$) and all force constants are equal ($k_1 = k_2 = k$).** *b. The eigenvalues of the mass-weighted Hessian matrix are $\lambda \in \{0, \frac{2k}{m}, \frac{4k}{m}\}$. Find the corresponding modes of oscillation (i.e, the eigenvectors). Note that one of the modes is doubly degenerate, meaning that two modes have the same frequency. c. Calculate the response of the system if monomers 2 and 4 are each displaced by an amount δ toward each other. That is, what are the displacements of the four monomers? d. Assume that the system is coarse-grained into two blocks: monomers 1 and 2 constitute a single block, and monomers 3 and 4 constitute another block. What is the*



projection matrix that transforms the system between the block space and the all-monomer space? e. Calculate the modes of vibration using this blocking scheme. Express your answer in terms of the coordinates of the original system. f. Recalculate the response of the system to the perturbation described in question 3, using the blocking scheme.

Bibliography

[1] Herbert Goldstein. *Classical Mechanics*. Addison-Wesley, Cambridge, MA, 1953.

[2] Alexander L. Fetter and John Dirk Walecka. *Theoretical Mechanics of Particles and Continua*. Courier Corporation, Chicago, IL, 2003.

[3] Grant R. Fowles and George L. Cassiday. *Analytical Mechanics*. Thomson Brooks/Cole, Stamford, CT, 2005.

Chapter 4

The Gaussian Network Model

One of the first ENMs developed for the study of protein dynamics is the Gaussian Network Model (GNM) of Bahar and others [1]. This model has its origins in elastic polymer theory, although the exact connection may be a little difficult to see. The early GNM papers refer to the work of Flory [2] as the inspiration for the model. Flory's paper discusses the elastic properties of bulk polymer materials, such as rubber. Specifically, the paper addresses polymer networks, or random networks of long polymers that cross and connect at certain points. In the Flory model, the nodes of the network are crossings of polymer chains, and its edges are the polymer chains themselves. The elastic properties of a random network of polymers thus arises as the sum of elastic interactions between nodes that are connected by springlike chains. In the GNM, the nodes are the protein residues, each represented by it alpha carbon atom. The edges represent the abstract realization of the net pairwise forces between residues in the folded protein. In the spirit of Flory, we can derive the GNM from the elastic nature of a simple chain molecule.

4.1 From polymer networks to proteins

Recall from Chapter 2 that the mean end-to-end vector $\mathbf{h}$ of a freely jointed chain with n edges of length l is given by

$$P(n, \mathbf{h}) = \left[\frac{3}{2\pi n l^2}\right]^{3/2} \exp\left\{-\frac{3h^2}{2nl^2}\right\} .$$

We can generalize this to describe the distribution of the vector connecting nodes i and j,

$$P(\mathbf{r}_{ij}) = \left[\frac{3}{2\pi(\mathbf{r}^0_{ij})^2}\right]^{3/2} \exp\left\{-\frac{3(\mathbf{r}_{ij})^2}{2(\mathbf{r}^0_{ij})^2}\right\} , \tag{4.1}$$

where $\mathbf{r}_{ij} = \mathbf{r}_j - \mathbf{r}_i$ is the vector displacement from node i to node j and has magnitude $r_{ij} = |\mathbf{r}_{ij}|$, and the zero superscript implies the crystal structure. Inspecting Eq. 4.1, we see that the inter-residue distance fluctuates about zero, meaning that the residues tend to be right on top of each other. This is an unphysical case that Flory refers to as "Phantom

Networks". He allows the nodes to pass through each other, and then he constrains some of them in a surrounding medium and lets the remainder relax. We will take a simpler approach of straightforwardly demanding the distances between residues i and j to fluctuate about the equilibrium distance r_{ij}^0. Defining displacements from the minimum energy conformation as $\mathbf{\Delta r}_i = \mathbf{r}_i - \mathbf{r}_i^0$, Eq. 4.1 becomes

$$P(\mathbf{\Delta r}_{ij}) = \left[\frac{\gamma_{ij}^*}{\pi}\right]^{3/2} \exp\left\{-\gamma_{ij}^*(\Delta r_{ij})^2\right\} \ ,$$

where $\mathbf{\Delta r}_{ij} = \mathbf{r}_{ij} - \mathbf{r}_{ij}^0 = \mathbf{\Delta r}_j - \mathbf{\Delta r}_i$ and $\gamma_{ij}^* = 3/2(r_{ij}^0)^2$. Assuming that the distances between nodes fluctuate independently, the probability of finding some displacement $\mathbf{\Delta r}$ of all residues from their equilibrium positions can be expressed as a product of the independent probabilities of pairwise distances. That is,

$$\begin{aligned} P(\mathbf{\Delta r}) &= \prod_{i=1}^{N}\prod_{j=i+1}^{N} P(\mathbf{\Delta r}_{ij}) \\ &= \frac{1}{Z_{GNM}} \exp\left\{-\sum_{i=1}^{N}\sum_{j=i+1}^{N} \gamma_{ij}^*(\Delta r_{ij})^2\right\}. \end{aligned} \tag{4.2}$$

The student can verify that this becomes

$$P(\mathbf{\Delta r}) = \frac{1}{Z_{GNM}} \exp\left\{-\sum_{i=1}^{N}\sum_{j=1}^{N} \gamma_{ij}\mathbf{\Delta r}_i \cdot \mathbf{\Delta r}_j\right\} \tag{4.3}$$

where

$$\gamma_{ij} = \begin{cases} -\gamma_{ij}^* & \text{if} \quad i \neq j \\ \sum_{k\neq i}\gamma_{ik}^* & \text{if} \quad i = j \end{cases} \ .$$

Expanding the dot product,

$$P(\mathbf{\Delta r}) = \frac{1}{Z_{GNM}} \mathrm{e}^{-\sum_{ij}\Delta x_i \gamma_{ij} \Delta x_j} \mathrm{e}^{-\sum_{ij}\Delta y_i \gamma_{ij} \Delta y_j} \mathrm{e}^{-\sum_{ij}\Delta z_i \gamma_{ij} \Delta z_j},$$

which looks like the product of three Gaussians.

Problem 4.1. *Show that Eq. 4.3 follows from Eq. 4.2.*

4.2 Equations of the GNM

If we assume that all non-zero off-diagonal elements have a value of $\gamma_{ij} = \gamma/2k_BT \ \forall \ i, j$ and define the N-component vectors of coordinate displacements $\mathbf{\Delta x} = (\Delta x_i \dots \Delta x_N)^T$ (and similarly for $\mathbf{\Delta y}$ and $\mathbf{\Delta z}$), we get

$$P_{GNM}(\mathbf{\Delta r}) = \frac{1}{Z_{GNM}} \exp\left\{-\frac{\gamma}{2k_BT}\left[\widetilde{\mathbf{\Delta x}}\mathbf{\Gamma}\mathbf{\Delta x} + \widetilde{\mathbf{\Delta y}}\mathbf{\Gamma}\mathbf{\Delta y} + \widetilde{\mathbf{\Delta z}}\mathbf{\Gamma}\mathbf{\Delta z}\right]\right\} \ , \tag{4.4}$$

where $\mathbf{\Gamma}$ is the *Kirchhoff Matrix*, also known as the *Laplacian Matrix*. More on this later. As we have slipped our old friend $\beta = 1/k_B T$ into the exponent, we can use the above equation to define the GNM potential:

$$V_{GNM} = \frac{\gamma}{2}\left[\widetilde{\mathbf{\Delta x}}\mathbf{\Gamma}\mathbf{\Delta x} + \widetilde{\mathbf{\Delta y}}\mathbf{\Gamma}\mathbf{\Delta y} + \widetilde{\mathbf{\Delta z}}\mathbf{\Gamma}\mathbf{\Delta z}\right] . \tag{4.5}$$

The Kirchhoff matrix $\mathbf{\Gamma}$ is unitless (more on this later), so the constant γ must have units of energy per squared distance. That is, it is a classical Hookean spring constant (recall $E = \frac{1}{2}k(\Delta x)^2$ from introductory physics). Using our knowledge of Gaussian integrals, we can find the GNM partition function,

$$\begin{aligned} Z_{GNM} &= \int \mathrm{d}^{3N}\mathbf{\Delta r}\ \exp\left\{-\frac{\gamma}{2k_B T}\left[\widetilde{\mathbf{\Delta x}}\mathbf{\Gamma}\mathbf{\Delta x} + \widetilde{\mathbf{\Delta y}}\mathbf{\Gamma}\mathbf{\Delta y} + \widetilde{\mathbf{\Delta z}}\mathbf{\Gamma}\mathbf{\Delta z}\right]\right\} \\ &= \int \mathrm{d}^N\mathbf{\Delta x}\ \mathrm{e}^{-\frac{\gamma}{2k_B T}\widetilde{\mathbf{\Delta x}}\mathbf{\Gamma}\mathbf{\Delta x}} \int \mathrm{d}^N\mathbf{\Delta y}\ \mathrm{e}^{-\frac{\gamma}{2k_B T}\widetilde{\mathbf{\Delta y}}\mathbf{\Gamma}\mathbf{\Delta y}} \int \mathrm{d}^N\mathbf{\Delta z}\ \mathrm{e}^{-\frac{\gamma}{2k_B T}\widetilde{\mathbf{\Delta z}}\mathbf{\Gamma}\mathbf{\Delta z}} \\ &= (2\pi)^{3N/2}\left|\frac{k_B T}{\gamma}\mathbf{\Gamma}^{-1}\right|^{3/2} . \end{aligned}$$

The covariance in fluctuations along the x-direction for residues i and j are

$$\begin{aligned} \langle \Delta x_i \Delta x_j \rangle &= \int \mathrm{d}^{3N}\mathbf{\Delta r}\ P_{GNM}(\mathbf{\Delta r})\Delta x_i \Delta x_j \\ &= (2\pi)^{-N/2}\left|\frac{\gamma}{k_B T}\mathbf{\Gamma}^{-1}\right|^{1/2} \int \mathrm{d}^N\mathbf{\Delta x'}\ \mathrm{e}^{-\frac{\gamma}{2k_B T}\widetilde{\mathbf{\Delta x'}}\mathbf{\Gamma}\mathbf{\Delta x'}}\Delta x_i \Delta x_j \\ &= \frac{k_B T}{\gamma}(\mathbf{\Gamma}^{-1})_{ij} . \end{aligned}$$

If we want to calculate the entire matrix of x- (or y-, or $z-$) covariances

$$\langle \mathbf{\Delta x}\widetilde{\mathbf{\Delta x}}\rangle = \langle \mathbf{\Delta y}\widetilde{\mathbf{\Delta y}}\rangle = \langle \mathbf{\Delta z}\widetilde{\mathbf{\Delta z}}\rangle = \frac{k_B T}{\gamma}\mathbf{\Gamma}^{-1} .$$

Using the definition of the inner product, we can find each residue's variance, as well as its covariance with other residues:

$$\begin{aligned} \langle \mathbf{\Delta r}_i^2 \rangle &= \frac{3k_B T}{\gamma}(\mathbf{\Gamma}^{-1})_{ii} \\ \langle \mathbf{\Delta r}_i \mathbf{\Delta r}_j \rangle &= \frac{3k_B T}{\gamma}(\mathbf{\Gamma}^{-1})_{ij} . \end{aligned} \tag{4.6}$$

To recap: We started with the assumptions that fluctuations in inter-residue distances are Gaussian – as described by polymer theory – and isotropic. From there we derived a distribution function that provides the probability of a structural deviation from the native state (Eq. 4.4), and we found the potential associated with this distribution (Eq. 4.5). A key part of this potential is the Kirchhoff matrix (more on that later), the inverse of which gives the residue-residue covariances to within a multiplicative constant (Eq. 4.6). Thus, armed with knowledge of the elastic properties of chain molecules, we can infer residue covariances from a single structure.

4.3 The Kirchhoff matrix

Matrices like $\mathbf{\Gamma}$ are quite common in graph theory. For an undirected graph (i.e., one in which the edges represent mutual interactions) an off-diagonal element Γ_{ij} of the Kirchhoff matrix is -1 if nodes i and j are connected by an edge and 0 otherwise. In the GNM, residues are connected if the distance between them in the crystal structure is less than a cutoff distance r_C, generally taken to be about 7Å. The diagonal elements are determined such that the sum over any row or column is zero. Mathematically,

$$\Gamma_{ij} = \begin{cases} -1 & \text{if} \quad i \neq j \;\&\; r_{ij} \leq r_C \\ 0 & \text{if} \quad i \neq j \;\&\; r_{ij} > r_C \\ \sum_{k \neq i} \Gamma_{ik} & \text{if} \quad i = j \end{cases} .$$

Sometimes the form of $\mathbf{\Gamma}$ is explained in terms of the *degree matrix* and the *adjacency matrix*, $\mathbf{\Gamma} = \mathbf{D} - \mathbf{A}$. The diagonal element D_{ii} of the degree matrix $\mathbf{D}$ indicates the degree (number of edges) of node i. All off-diagonal elements of $\mathbf{D}$ are zero. The element A_{ij} of the adjacency matrix $\mathbf{A}$ is 1 if nodes i and j are connected and 0 otherwise. The Kirchhoff matrix is real and symmetric, so it can be decomposed with an orthogonal transformation. As discussed below, it is positive semi-definite, indicating that all of its eigenvalues are non-negative real numbers. This allows us to bestow them with physical meaning. It turns out that $\mathbf{\Gamma}$, which is constructed from a single protein structure, contains just about all of the information that we need to estimate certain properties of the protein's equilibrium dynamics.

Problem 4.2. *Show that* $\mathbf{\Gamma}$ *has one zero eigenvalue.*

As it cannot be inverted properly, we use its pseudo-inverse (Eq. 1.4) to calculate the fluctuations and covariances in Eq. 4.6. The mode associated with the zero eigenvalue represents rigid translations of the entire system. For a protein of N residues, each component of this mode has a value of $1/\sqrt{N}$. This single mode arises from the isotropic assumption of GNM: An average translation in x is accompanied by an identical average translation in y and z. It should be noted, however, that the GNM is *not* rotationally invariant [3]. Rigid rotations of the full molecule increase the system's energy, causing some to argue that the model is unphysical. It is nonetheless useful for exploring motions that do not include rigid-body rotations or translations.

4.4 Comparing the GNM to experiments

The two common ways of comparing GNM predictions to experiment are through mean squared fluctuations (MSFs) and mode shapes. MSFs are contained in the diagonal elements of the covariance matrix given by Eq. 4.6. Experimentally, MSFs can be inferred from multiple sources, most commonly X-ray B-factors. The B-factor for atom i is

$$\begin{aligned} B_i &= \frac{8\pi^2}{3} \langle (\mathbf{\Delta r}_i)^2 \rangle \\ &= \frac{8\pi^2 k_B T}{\gamma} (\mathbf{\Gamma}^{-1})_{ii} \end{aligned} \tag{4.7}$$

The Pearson correlation between GNM-predicted MSFs and experimental B-factors is generally accepted to be around 0.6 [4]. Whether or not this constitutes a "good" agreement is a topic that is still open for discussion, but the fact that such a simple model gives a non-zero correlation with experiment is impressive.

Better correlations (around 0.75) are found by comparing GNM-predicted MSFs with the sample variance of NMR models [4]. Generally, solution NMR does not permit the identification of a single best protein structure, but suggests an ensemble of structures. PDB files of proteins solved using NMR therefore contain multiple models, each of which satisfies the restraints that are measured in the experiment. If the collection of NMR models is used to approximate protein fluctuations, we can compare it with the fluctuations predicted by the GNM. It is important to note that the structures deposited to the PDB are not necessarily *all* of the structures that satisfy the constraints, but some representative subset. Further, they are fit to the experimental data using harmonic functions, so they may have a natural tendency to agree with the similarly harmonic GNM.

A second way of comparing GNM results with experiment is through the individual modes. Decomposing $\mathbf{\Gamma}$ in the standard way,

$$\mathbf{\Gamma} = \mathbf{V}\mathbf{\Lambda}\tilde{\mathbf{V}} \ ,$$

we can see that the MSFs (Eq. 4.6) are the weighted sums of MSFs from the individual modes:

$$\begin{aligned} \langle \mathbf{\Delta r}_i^2 \rangle &= \frac{3k_BT}{\gamma}(\mathbf{\Gamma}^{-1})_{ii} \\ &= \frac{3k_BT}{\gamma}\sum_{k=1}^{N-1}\frac{(V_{ik})^2}{\lambda_k} \ . \end{aligned}$$

The contribution of mode k to the MSF of any residue goes like $1/\lambda_k$, so the smallest eigenvalues of $\mathbf{\Gamma}$ have the largest influence on dynamics. It so happens that $1/\lambda_k$ decreases quickly with k, and usually only the top 10 or so modes need to be used to estimate MSFs. It has even been suggested through rigorous comparison with random matrices that only about the top three modes are significant [5].

Analysis of theoretical MSFs is straightforward. Large MSFs indicate high mobility, and small MSFs indicate low mobility. When two regions of high mobility are separated by a region of low mobility, a *hinge* is suggested: That is, a stable point between two moving regions. This same analysis applies to the individual modes. Mode k *mobilizes* residue i if $(V_{ik})^2$ is large, and it *stabilizes* the residue if $(V_{ik})^2$ is small. But we can get more information from analyzing modes individually. The contribution $(V_{ik})^2$ is always non-negative, even though V_{ik} itself has a sign. The choice of sign is arbitrary, but it indicates the motion of the residue relative to other residues.

Consider two residues, i and j. If $V_{ik}V_{jk} > 0$, then the residues are correlated in mode k. If $V_{ik}V_{jk} < 0$, then they are anticorrelated in the mode. By plotting the modes themselves rather than their squares, we can distinguish true hinges from regions of damped motions. A zero-crossing in the mode indicates that motions of neighboring regions in the protein are

anti-correlated. Hinge regions identified in this manner have shown to be associated with ligand binding and catalytic activity.

Bibliography

[1] I. Bahar, A. R. Atilgan, and B. Erman. Direct evaluation of thermal fluctuations in proteins using a single-parameter harmonic potential. *Folding & Design*, 2(3):173–181, 1997.

[2] P. J. Flory. Statistical thermodynamics of random networks. *Proc. R. Soc. Lond. A.*, 351:351–380, 1976.

[3] M. F. Thorpe. Comment on elastic network models and proteins. *Phys. Biol.*, 4:60–63, 2007.

[4] Lee-Wei Yang, Eran Eyal, Chakra Chennubhotla, JunGoo Jee, Angela M. Gronenborn, and Ivet Bahar. Insights into equilibrium dynamics of proteins from comparison of nmr and x-ray data with computational predictions. *Structure*, 15(6):741–749, JUN 2007.

[5] Raffaello Potestio, Fabio Caccioli, and Pierpaolo Vivo. Random matrix approach to collective behavior and bulk universality in protein dynamics. *Physical Review Letters*, 103:268101, 2009.

Chapter 5

The Anisotropic Network Model

The ANM is a model for exploring the equilibrium dynamics of macromolecules. It is similar in nature to the GNM, but does not assume spatial isotropy of residue fluctuations. A protein of N residues has $3N$ degrees of freedom in the ANM: One degree for each Cartesian coordinate of each residue. The same mathematical approach is taken for the ANM as for simpler systems, with a few critical adjustments.

5.1 ANM potential

The ANM assumes that nearby residues interact through harmonic potentials that are all at their minumum in the equilibrium conformation. It is a coarse-grained model in which the residue positions are approximated as coincident with the coordinates of the C_α atoms. The distance between two residues i and j with positions $[x_i\ y_i\ z_i]^T$ and $[x_j\ y_j\ z_j]^T$ is

$$R_{ij} = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2 + (z_j - z_i)^2}\ .$$

Their equilibrium distance – that is, their distance in the crystal structure – is defined as R^0_{ij}, with the superscript zero indicating equilibrium conformation throughout what follows. If these residues are joined via a Hookean spring with force constant γ, the potential associated with any conformation of residues i and j is

$$V = \gamma/2(R_{ij} - R^0_{ij})^2\ .$$

Thus, the potential is not directly dependent on the locations of the residues, but on the distance between them. Specifically, it depends on the difference between the inter-residue distance in the instantaneous and the equilibrium conformation.

The ANM potential,

$$V_{ANM} = \frac{\gamma}{2}\sum_{i=1}^{N}\sum_{j=i+1}^{N}(R_{ij} - R^0_{ij})^2\Theta(r_c - R^0_{ij})\ , \tag{5.1}$$

is just a sum of potentials that are harmonic in the distance between residues. In Eq. 5.1, the double summation covers all $N(N-1)/2$ pairs of residues in the protein, allowing each

pair to contribute to the energy. The Heavyside step function $\Theta(r_c - R_{ij}^0)$ is 0 if $R_{ij}^0 > r_c$ and 1 otherwise. This function ensures that only residue pairs that are within a cutoff distance of r_c in the crystal structure are connected by springs. This is a very simple model that can be constructed using only a single crystal structure.

5.1.1 The second order approximation

A key difference between the ANM and the models in Chapters 3 and 4 is that the ANM allows particles to move in three dimensions instead of just one. Although this may seem like something that can be handled without much difficulty, it has a significant impact on the mathematics of the model because the potential is no longer a quadratic function of the particle positions, as it was in Eq. 3.7. To get such a function, we first define a $3N$-dimensional coordinate vector $\mathbf{r} = [x_1 \;\; y_1 \;\; z_1 \ldots x_N \;\; y_N \;\; z_N]^T$. Based on this definition, the x-, y- and z-components of residue i are r_{3i-2}, r_{3i-1} and r_{3i}, respectively. Eq. 5.1 can then be Taylor expanded about the equilibrium conformation $\mathbf{r^0}$ as

$$V_{ANM}(\mathbf{r}) = V_{ANM}(\mathbf{r^0}) + \sum_{i=1}^{3N} \left.\frac{\partial V}{\partial r_i}\right|_{\mathbf{r^0}} (r_i - r_i^0) + \frac{1}{2}\sum_{i=1}^{3N}\sum_{j=1}^{3N} \left.\frac{\partial^2 V}{\partial r_i \partial r_j}\right|_{\mathbf{r^0}} (r_i - r_i^0)(r_j - r_j^0) + \ldots$$

The first term is a constant that can be set to zero. The second term is the sum over first derivatives of the potential with respect to the coordinates, evaluated at $\mathbf{r^0}$. By definition, $\mathbf{r^0}$ is a minimum of the potential, so all terms in the sum vanish. To leading order, this leaves

$$\begin{aligned} V_{ANM}(\mathbf{r}) &\approx \frac{1}{2}\sum_{i=1}^{3N}\sum_{j=1}^{3N} \left.\frac{\partial^2 V}{\partial r_i \partial r_j}\right|_{\mathbf{r^0}} (r_i - r_i^0)(r_j - r_j^0) \\ &= \frac{1}{2}\widetilde{\mathbf{\Delta r}}\mathbf{H}\mathbf{\Delta r} \;, \end{aligned} \tag{5.2}$$

where $\mathbf{\Delta r} \equiv \mathbf{r} - \mathbf{r^0}$ and $\mathbf{H}$ is the Hessian matrix (Eq. 3.9) evaluated at $\mathbf{r^0}$. When expanded to second order, the ANM potential becomes harmonic in the C_α displacements from the crystal structure. This is convenient because the dynamics in a harmonic potential can be solved analytically.

Following the simple example outlined in Chapter 3, we can write the equations of motion as

$$\mathbf{M}\ddot{\mathbf{\Delta r}} = -\mathbf{H}\mathbf{\Delta r} \;, \tag{5.3}$$

where $\mathbf{M}$ is now a $3N \times 3N$ diagonal matrix of residue masses. The mass, m_i, of residue i appears on the diagonal elements of the rows corresponding to x_i, y_i and z_i, which are rows $3i-2$, $3i-1$ and $3i$, respectively. Mathematically,

$$M_{3i-k,j} = m_i \delta_{3i-k,j} \;\; k \in \{0, 1, 2\} \;.$$

In the earlier example of small oscillations, we solved the equations of motion in mass-weighted coordinates. In the ANM, we avoid complications associated with mass-weighting

by assuming that all residues have equal mass, m_0. Under this assumption, the mass matrix becomes a multiple of the identity matrix: $\mathbf{M} = m_0\mathbf{1}$. The scalar mass m_0 is easy to move around in equations, without needing to invert or ensure symmetries. Under this assumption, Eq. 5.3 becomes

$$\ddot{\mathbf{\Delta r}} = -\frac{1}{m_0}\mathbf{H}\mathbf{\Delta r} \ ,$$

and a solution of the form $\mathbf{\Delta r}(t) = \mathbf{\Delta r^0}\sin(\omega t + \phi)$ yields the eigenvalue equation

$$m_0\omega^2\mathbf{\Delta r} = \mathbf{H}\mathbf{\Delta r} \ .$$

This is satisfied when $\mathbf{\Delta r}$ is an eigenvector of $\mathbf{H}$, in which case $m_0\omega^2$ is its eigenvalue. The eigenvectors, $\{\mathbf{v}^{(1)}, \ldots, \mathbf{v}^{(3N)}\}$, of $\mathbf{H}$ are the ANM modes. Each has $3N$ components, corresponding to the displacements along the three coordinates of each residue. The first three components correspond to the x-, y- and z-motions of the first residue, the next three components correspond to the second residue, and so on. The eigenvalues $\{\lambda_1, \ldots, \lambda_{3N}\}$ are the effective force constants associated with stretching or squeezing the protein along the modes.

5.1.2 Hessian matrix: Superelements n'at

The form of the Hessian matrix $\mathbf{H}$ is worth exploring. We have already seen that its elements are the second partial derivatives,

$$H_{ij} = \left.\frac{\partial^2 V}{\partial r_i \partial r_j}\right|_{\mathbf{r^0}} \ ,$$

and the student can verify that this reduces to

$$\left.\frac{\partial^2 V}{\partial \alpha_i \partial \beta_j}\right|_{\mathbf{r^0}} = -\frac{\gamma(\alpha_j^0 - \alpha_i^0)(\beta_j^0 - \beta_i^0)}{(r_{ij}^0)^2} \qquad i \neq j \qquad (5.4)$$

$$= \gamma \sum_{k \neq j} \frac{(\alpha_k^0 - \alpha_j^0)(\beta_k^0 - \beta_j^0)}{(r_{kj}^0)^2} \qquad i = j \qquad (5.5)$$

where $\alpha, \beta \in \{x, y, z\}$.

The ANM Hessian matrix is $3N \times 3N$ and is best thought of as an $N \times N$ matrix of 3×3 matrices. Associated with each pair of residues, i and j, is a 3×3 matrix describing how relative motions of those residues influence the potential. These 3×3 matrices are called *superelements*, and the superelement of residues i and j is written $\mathbf{H_{ij}}$, where the subscripts are in bold type. When residue i is displaced along the x-axis by some amount $\mathrm{d}x_i$ from its equilibrium position, and residue j is displaced along the y-axis by some amount $\mathrm{d}y_i$, the corresponding increase in energy is given by $\mathbf{H_{ij}}^{xy}\mathrm{d}x_i\mathrm{d}y_j = H_{3(i-1)+1,3(j-1)+2}\mathrm{d}x_i\mathrm{d}y_j$.

We can see from the equations above that the diagonal superelements of the Hessian are constructed such that the sum over the superelements in a superrow or supercolumn will be zero (or, more precisely, a 3×3 matrix of zeros). As a result, the elements of the

Hessian are not all linearly independent, and the Hessian has some eigenvalues that are zero. If the system is properly connected, ANM will always give exactly six eigenvalues of zero, corresponding to rigid-body translations and rotations of the system. The reason for this is that rigid motions of the system as a whole do not affect the relative positions of the particles, and therefore have no effect on the internal energy. Translations along the three Cartesian coordinates and rotations along the three principal axes of inertia do nothing to the system's energy. This is independent of the number of particles in the system, but depends on the dimension. We saw earlier that a system of N particles in one dimension has a single degree of freedom – translation along the coordinate axis – and a single zero eigenvalue. The eigenvector confirmed that translations along the coordinate axis leave the energy unchanged.

In two dimensions, there are three types of motion that do not change the energy: Translations along the two axes that define the plane containing the system, and a rotation about the normal to the plane. Conceptually, each motion corresponds to an element in the Hessian superelement. The Hessian superelements of a two-dimensional system contain four components each,

$$\mathbf{H_{ij}} = \frac{\gamma}{(r_{ij}^0)^2} \begin{pmatrix} (x_j^0 - x_i^0)^2 & (x_j^0 - x_i^0)(y_j^0 - y_i^0) \\ (y_j^0 - y_i^0)(x_j^0 - x_i^0) & (y_j^0 - y_i^0)^2 \end{pmatrix} .$$

The diagonal elements of the super-element correspond to rigid translations: If there is no motion along y, then the only terms in the potential will be those containing $(x_j^0 - x_i^0)^2$. The form of the diagonal superelements ensures that a rigid translation in the x direction will cancel these terms. Similarly, a rigid translation in the y direction will cause no overall contribution from the $(y_j^0 - y_i^0)^2$ terms in the Hessian. The mixed terms correspond to the rotation. As the Hessian superelement is symmetric, there is only one unique mixed term per superelement, or one rotation. Only motions that have the proper combination of x and y can lead to vanishing of the mixed terms. These are the rigid rotations.

In three dimensions, the Hessian superelements each has nine elements, six of which are unique by symmetry. The three diagonal elements of the superelement correspond to rigid translations along the Cartesian coordinates, and the off-diagonal mixed elements correspond to three rotations. Again, the correct combination of x y and z motions will result in canceling the mixed terms in the Hessian, leading to no change in energy.

What if something other than six zero modes are found? This indicates that the system is not properly constrained. Fewer than six zero modes result from excessive constraints. This usually does not happen, but can stem from tethering the molecule to a specific point in space, or to a particular orientation. A more common occurrence is finding more than six zero modes, indicating an underconstrained system. Perhaps the most obvious case of an underconstrained system is one consisting of two disjointed subsystems. Consider a PDB file containing two structures that are spatially separated by a distance greater than r_c. When the ENM is constructed over the entire system, there will be no connection between the two sub-structures, resulting in two independent systems, each with 6 degrees of freedom. This will clearly show up as 12 zero modes in the ANM, but it is not the only situation that can lead to 12 zero modes.

The six zero modes stem from motions that do not change the conformation of the system in any way that alters its internal energy. If the system is under-constrained, there may be additional motions that do not increase its internal energy. Consider a residue that is connected to only two other residues. The two connections restrain the residue to a fixed distance from two points in space. Mathematically, the residue is allowed to be anywhere on the circle defined by these two distances and these two points. Any motion of the residue along the circle will leave the internal energy of the system unchanged, as no springs are stretched by the motion. This situation leads to an additional degree of freedom. In general, each residue must have at least 3 connections in order for the system to be properly restrained. The minimum number of connections depends on the dimensionality of the system. For a system of n dimensions, each node must have at least n edges, in general. There are exceptions to this rule, but they only occur in special cases. Additional zero modes will arise if the system has other freedoms.

Problem 5.1. *Show that the superelements of the ANM Hessian matrix reduce to the form given in Eqs. 5.4 and 5.5.*

Problem 5.2. *Describe a 3D system in which a residue that is connected to only two other residues is properly restrained. Describe a case in which a residue connected to three other residues is unadequately restrained.*

Problem 5.3. *Describe situations leading to 8 and 9 zero modes.*

5.1.3 The Hessian-covariance connection

As we have seen with GNM, we can calculate the partition function and dynamical information from the ANM potential (Eq. 5.2). To get the partition function, we integrate the potential over all conformations of the structure, applying Boltzmann weighting:

$$\begin{aligned} Z_{ANM} &= \int \mathrm{d}^N\mathbf{r} \exp\left\{-\frac{1}{2k_\mathrm{B}T}\tilde{\mathbf{r}}\mathbf{H}\mathbf{r}\right\} \\ &= (2\pi k_\mathrm{B}T)^{N/2}\left(\det(\mathbf{H}^\dagger)\right)^{1/2} . \end{aligned}$$

Because $\det(\mathbf{H}^\dagger)$ is the product of the reciprocal nonzero eigenvalues of $\mathbf{H}$, the lowest frequency modes contribute most to Z_{ANM}.

The correlations between the system's coordinates,

$$\begin{aligned} \langle \Delta r_i \Delta r_j \rangle &= \frac{1}{Z}\int \mathrm{d}^N\mathbf{r} \exp\left\{-\frac{1}{2k_\mathrm{B}T}\tilde{\mathbf{r}}\mathbf{H}\mathbf{r}\right\} \Delta r_i \Delta r_j \\ &= k_\mathrm{B}T(\mathbf{H}^\dagger)_{ij} , \end{aligned}$$

can be solved through Gaussian integration and show us that the $3N \times 3N$ covariance matrix is the inverse of the Hessian to within a factor of $k_\mathrm{B}T$.

5.2 Comparing to experiment

5.2.1 MSFs

Both the ANM and the GNM predict the amount by which each residue's position fluctuates under equilibrium conditions. That is, they both provide estimates for residue mean squared fluctuations. As we have seen earlier, MSFs are, within a multiplicative constant, captured by the diagonal elements of the inverse of the GNM Kirchhoff matrix. In ANM, the $3N \times 3N$ covariance is the pseudoinverse of the Hessian, and its diagonal elements represent the directional fluctuations of the residues. The MSF of residue i is given in terms of the displacements along the Cartesian coordinates by

$$\langle(\Delta\mathbf{r}_i)^2\rangle = \langle\Delta x_i^2\rangle + \langle\Delta y_i^2\rangle + \langle\Delta z_i^2\rangle \ . \tag{5.6}$$

Thus, in order to find the MSF of residue i, we add the diagonal elements of the covariance matrix that correspond to residue i. As the diagonal elements of the covariance matrix are in fact variances (no "co" needed), this is simply stating that the total variance of a residue's position is the sum of its variances along the three Cartesian coordinates. It is the trace of the superelement of the Hessian pseudo-inverse corresponding to residue i, which we will represent as

$$\langle(\Delta\mathbf{r}_i)^2\rangle = (k_{\mathrm{B}}T)\ \mathrm{Tr}(\mathbf{H}_{\mathbf{ii}}^{\dagger}) \ ,$$

where the bold subscript indicates a 3×3 superelement.

Studies have shown that MSFs calculated from ANM do not agree as well with experimentally determined motions as do MSFs calculated from GNM; however, ANM predictions go well beyond those of the GNM. MSFs are really not well-suited for assessing the performance of the ANM, largely because they do not account for directionality. Equation 5.6 shows that the MSF is a sum of squares, so that a given MSF is associated with an infinite number of vectors. The MSF is telling us the magnitude of a residue's motion, but it is neglecting the direction. Once we know residue MSFs, we can randomly reassign the x-, y- and z-components of the variance such that they have the same sum but different individual values. That is to say, the mapping of ANM Hessians to MSFs is many-to-one, so we lose information when going from ANM to MSFs. The ANM has much more to offer than estimates of fluctuation magnitudes.

Problem 5.4. *Show that* $\langle(\Delta\mathbf{r}_i)^2\rangle = \langle\Delta x_i^2\rangle + \langle\Delta y_i^2\rangle + \langle\Delta z_i^2\rangle$*, where* $\Delta\mathbf{r}_i = \mathbf{r}_i - \mathbf{r}_i^0$ *is the displacment of residue i from its equilibrium position.*

5.2.2 ADPs

One way to retain some of the information provided by ANM is to use anisotropic displacement parameters (ADPs) that are present in some PDB files. For structures of sufficiently high resolution, the spatial anisotropy of atomic fluctuations can be determined, and each atom's spatial distribution is described by a set of six ADPs representing the upper triangle of its variance-covariance matrix. Using the relationship between Hessian and covariance

matrices, the ADPs of atom i correspond to the upper-triangle of the diagonal superelement of the Hessian pseudoinverse

$$k_{\mathrm{B}}T(\mathbf{H}^{\dagger}_{\mathbf{ii}}) \; .$$

ADPs are essentially Gaussian ellipsoids centered on the atoms, and a number of methods can be used to quantitatively compare ADPs to ANM-predicted fluctuations. As with the MSFs, one can simply calculate the correlation between ADPs and ANM fluctuations. Instead of comparing N MSFs with N b-factors, the comparison here is of $6N$ covariances with $6N$ ADPs. Alternatively, one might consider the extent to which the ellipsoids defined by ADPs overlap with those defined by ANM fluctuations. One way to do this is using the principal axes of motion. The ADP matrix for residue i can be decomposed, revealing eigenvectors that indicate the principal directions of motion for the residue. The largest such motion can be directly compared to its analagous vector in the ANM-approximated distribution using the inner product. In essence, this is comparing two covariance matrices, which will be discussed in greater detail later.

A caveat of using ADPs is that they require structural alignment. Although MSFs have no directionality, ADPs depend on the spatial orientation of the molecule. In order to compare ADPs from two molecules, it is essential that they are properly aligned. This is addressed later. Note that there is still a many-to-one mapping of Hessians to ADPs: The ADPs tell how each residue moves individually, but they tell us nothing about correlations between residue motions. For this, we need to look at the individual modes.

5.2.3 Deformations

One of the great things about ANM is that its results can be compared to deformations between two known structures. The PDB has high sequence redundancy, and many proteins have more than one crystal structure. The structures may correspond to different bound states, such as *apo* vs. *holo* forms, or the same protein bound to different ligands. They may also correspond simply to structures solved by different laboratories, or using different methods. In any case, it is assumed that each structure represents a local free energy minimum and therefore represents a local population maximum within the ensemble of conformations accessible to the protein. We can say that each structure represents a stable state of the protein, and the modes generated through the ANM provide a convenient basis for studying putative transition pathways between known states.

Suppose that the $3N$-component vectors $\mathbf{a}$ and $\mathbf{b}$ contain the C_α coordinates for two structures of the same N residue protein. The $3N$-component *deformation vector* that describes the transition from state $\mathbf{a}$ to state $\mathbf{b}$ is given by

$$\mathbf{d}_{ab} = \mathbf{a} - \mathbf{b} \; .$$

The identity $\mathbf{b} = \mathbf{a} + \mathbf{d}_{ab}$ shows that, starting from structure $\mathbf{a}$, we can arrive at structure $\mathbf{b}$ by deforming the structure by an amount $\mathbf{d}_{ab}$. Using the completeness of the orthogonal basis described by the ANM modes, $\mathbf{d}_{ab}$ can be expressed in terms of the modes as

$$\mathbf{d}_{ab} = \sum_{k=1}^{3N} \left(\hat{\mathbf{v}}^{(k)} \cdot \mathbf{d}_{ab} \right) \hat{\mathbf{v}}^{(k)} \; ,$$

where the inner product $\hat{\mathbf{v}}^{(k)} \cdot \mathbf{d}_{ab}$ is the projection of $\mathbf{d}_{ab}$ onto mode k. It tells how much of the deformation is accounted for by mode k. This implies that if we have ANM modes calculated about the structure **a**, we can deform the structure a little bit along each of the modes and by doing so reach the state **b**. This is interesting because our modes are not just arbitrary directions in the conformation space, but are aligned with directions of energetically favorable deformations and ordered according to importance. If the ANM properly captures the equilibrium dynamics of the molecule, one might expect the least energetic modes to account for the bulk of the deformation. We will claim that this is the case, and quantify the extent to which this claim may or may not hold.

Note that if **a** and **b** are properly aligned to eliminate rotations and translations, only the $3N-6$ internal modes contribute to $\mathbf{d}_{ab}$. A proper alignment is here essential, as the deformation vector will be affected by adding rotations or translations. The term *deformation* implies that it is the structure, not its position or orientation in space, that is changing.

Normalizing by the magnitude of $\mathbf{d}_{ab}$ provides the unit vector that points in the direction of the deformation,

$$\hat{\mathbf{d}}_{ab} = \mathbf{d}_{ab}/|\mathbf{d}_{ab}|^2 \ .$$

Expressed in terms of the modes,

$$\hat{\mathbf{d}}_{ab} = \sum_{k=1}^{3N-6} \left(\hat{\mathbf{v}}^{(k)} \cdot \hat{\mathbf{d}}_{ab}\right) \hat{\mathbf{v}}^{(k)} \ ,$$

and using the normalization of the unit vector leads to

$$\begin{aligned}
1 &= \hat{\mathbf{d}}_{ab} \cdot \hat{\mathbf{d}}_{ab} \\
&= \left[\sum_{k=1}^{3N-6} \left(\hat{\mathbf{v}}^{(k)} \cdot \hat{\mathbf{d}}_{ab}\right) \hat{\mathbf{v}}^{(k)}\right] \cdot \left[\sum_{k'=1}^{3N-6} \left(\hat{\mathbf{v}}^{(k')} \cdot \hat{\mathbf{d}}_{ab}\right) \hat{\mathbf{v}}^{(k')}\right] \\
&= \sum_{k=1}^{3N-6} \sum_{k'=1}^{3N-6} \left(\hat{\mathbf{v}}^{(k)} \cdot \hat{\mathbf{d}}_{ab}\right) \left(\hat{\mathbf{v}}^{(k')} \cdot \hat{\mathbf{d}}_{ab}\right) \hat{\mathbf{v}}^{(k)} \cdot \hat{\mathbf{v}}^{(k')} \\
&= \sum_{k=1}^{3N-6} \sum_{k'=1}^{3N-6} \left(\hat{\mathbf{v}}^{(k)} \cdot \hat{\mathbf{d}}_{ab}\right) \left(\hat{\mathbf{v}}^{(k')} \cdot \hat{\mathbf{d}}_{ab}\right) \delta_{kk'} \\
&= \sum_{k=1}^{3N-6} \left(\hat{\mathbf{v}}^{(k)} \cdot \hat{\mathbf{d}}_{ab}\right)^2 \ ,
\end{aligned}$$

which is just a long way of showing that the sums of the squares of components of a unit vector is unity. The inner product $\hat{\mathbf{v}}^{(k)} \cdot \hat{\mathbf{d}}_{ab}$ is the cosine of the angle between the deformation and mode k, and its square is the fraction of the deformation that is captured by mode k.

We can now quantify the extent to which any mode overlaps with a known deformation. Assuming that the structure fluctuates about its equilibrium position in a random walk that is weighted by the mode energies, we expect the largest structural excursions to take place along the softest modes. We therefore also expect that, if an observed deformation results

from allowed fluctuations along normal modes, it will overlap most with the modes of least energy. There are countless examples of proteins with known structural deformations that are captured well by only a few ANM modes, and such claims indeed are responsible for the popularity of ANM.

A common way to present ANM results in terms of a known deformation between two structures is to plot the *cumulative overlap*,

$$\begin{aligned} O^m &= \left[\sum_{k=1}^{m} \left(\hat{\mathbf{v}}^{(k)} \cdot \frac{\mathbf{d}}{|\hat{\mathbf{d}}|} \right)^2 \right]^{1/2} \\ &= \frac{1}{|\mathbf{d}|} \left[\sum_{k=1}^{m} \left(\hat{\mathbf{v}}^{(k)} \cdot \mathbf{d} \right)^2 \right]^{1/2} , \end{aligned} \tag{5.7}$$

for the first m modes. Recall that the ANM modes are ordered from lowest to highest energy, or eigenvalue. But the ANM eigenvalues cannot be taken too seriously, because they don't account for the damping effects of the solvent. Although ANM calculations are based on the structures of solvated proteins, the ANM potential represents a system *in vacuo*. The solvent that surrounds the protein should dampen motion along the ANM modes, realistically. It will actually overdamp the motions, so that proteins do not oscillate along the modes, but randomly explore them. Nonetheless, the relative lengths of excursions along the modes are expected to follow the same order as the eigenvalues. All this means is that we know the order of the modes, but we don't necessarily know exactly their magnitudes. The cumulative overlap measures how much of the vector $\mathbf{d}$ is accounted for by the first m modes. It depends on the modes occurring in a certain order, but it does not depend on the magnitudes of their eigenvalues. Usually, to show that the first few slow modes contribute disproportionately to an inferred deformation, researchers will plot the cumulative overlap for several small values of m.

A problem with this approach is that it is not symmetric. ANM will more easily predict open-to-closed conformational changes than closed-to-open changes. When the conformation is open, Note that assuming equal residue mass eliminated the need to mass-weight the Hessian. it has some outlying regions that have few inter-residue contacts. These regions are loosely restrained and highly mobile. They can move in two directions, as indicated by the mathematical invariance of eigenvectors to a change in sign: Either they open more or they close. We select the mode in the direction of closing, so there is high overlap with the deformation. In the case of the closed structure, there are no highly mobile residues, so the energetic differences among the global modes are smaller.

5.2.4 Overlap vs. RMSD

It is common practice to compare the predicted modes to some known vector, such as the deformation between two alternate crystal structures of a protein, and we have seen above how overlap can be used to quantify the agreement between modes and deformations. But what does some overlap or cumulative overlap value really mean, in terms of the structure? How much overlap is needed before there is "good" agreement between the model and

the observed deformation? Here we take a brief look at how the overlap between modes and displacement vectors relates to the reduction of root mean squared distance (RMSD) between the structures. We start by assuming that we have a protein of N residues that has two crystal structures, represented by the $3N$-dimensional vectors $\mathbf{a} = (a_1 \dots a_{3N})^T$ and $\mathbf{b} = (b_1 \dots b_{3N})^T$.

In terms of the displacement vector, $\mathbf{d} = \mathbf{b} - \mathbf{a}$, from $\mathbf{a}$ to $\mathbf{b}$, the RMSD is

$$\begin{aligned} D_{RMS}(\mathbf{d}) &= \left[\frac{1}{3N}\sum_{i=1}^{3N}(d_i)^2\right]^{1/2} && (5.8)\\ &= \left[\frac{1}{3N}\mathbf{d}\cdot\mathbf{d}\right]^{1/2} && (5.9)\\ &= \frac{|\mathbf{d}|}{\sqrt{3N}}\,. && (5.10)\end{aligned}$$

If the set $\{\hat{\mathbf{v}}^{(1)}, \dots, \hat{\mathbf{v}}^{(3N)}\}$ of normalized modes is calculated about the structure $\mathbf{a}$, we can define the cumulative overlap of the first m modes with $\mathbf{d}$ using Eq. 5.7. As the vectors $\{\hat{\mathbf{v}}^{(1)}, \dots, \hat{\mathbf{v}}^{(3N)}\}$ form an orthonormal basis over the deformations of the structure, we know that $O^{3N} = 1$. Using Eq. 5.7, we find

$$\sum_{k=1}^{3N}\left(\hat{\mathbf{v}}^{(k)}\cdot\mathbf{d}\right)^2 = |\mathbf{d}|^2\ , \tag{5.11}$$

from which it follows

$$\begin{aligned} \langle(\hat{\mathbf{v}}^{(k)}\cdot\mathbf{d})^2\rangle &= \frac{1}{3N}\sum_{k=1}^{3N}\left(\hat{\mathbf{v}}^{(k)}\cdot\mathbf{d}\right)^2 && (5.12)\\ &= \frac{|\mathbf{d}|^2}{3N}\,. && (5.13)\end{aligned}$$

The expected cumulative overlap of a vector $\mathbf{d}$ on m modes is then

$$\begin{aligned} \langle O^m\rangle &= \frac{1}{|\mathbf{d}|}\left[m\langle(\hat{\mathbf{v}}^{(k)}\cdot\mathbf{d})^2\rangle\right]^{1/2} && (5.14)\\ &= \sqrt{\frac{m}{3N}}\,. && (5.15)\end{aligned}$$

Thus, the cumulative overlap is expected to scale with the square root of the number of modes used. This value can be used as a comparison when evaluating how well the predicted modes reproduce known deformations.

The amount by which the RMSD is reduced by a given mode can also be calculated. Defining the coefficients $c_k = \hat{\mathbf{v}}^{(k)}\cdot\mathbf{d}$, the deformation is written in the basis of the modes as

$$\mathbf{d} = \sum_{k=1}^{3N} c_k\hat{\mathbf{v}}^{(k)}, \tag{5.16}$$

where c_k is the displacement along mode k that minimizes the RMSD. After a displacement of c_1 along mode $\hat{\mathbf{v}}^{(1)}$, the displacement vector becomes $\mathbf{d}' = \mathbf{d} - c_1\hat{\mathbf{v}}^{(1)}$ with a magnitude that satisfies $|\mathbf{d}'|^2 = |\mathbf{d}|^2 - (c_1)^2$. Using Eq. 5.10, we find

$$\begin{aligned} D_{RMS}(\mathbf{d}') &= \frac{|\mathbf{d}'|}{\sqrt{3N}} && (5.17)\\ &= \left[\frac{|\mathbf{d}|^2 - (c_1)^2}{3N}\right]^{1/2} && (5.18)\\ &= \frac{|\mathbf{d}|}{\sqrt{3N}}\sqrt{1-f^2} && (5.19)\\ &= \mathrm{RMSD}(\mathbf{d})\sqrt{1-f^2}\,, && (5.20) \end{aligned}$$

where $f = \hat{\mathbf{v}}^{(1)} \cdot \hat{\mathbf{d}} = c_1/|\mathbf{d}|$ is the normalized overlap of $\mathbf{d}$ with $\hat{\mathbf{v}}^{(1)}$. Thus, a deformation along a single mode that has an overlap of 0.6 with the initial displacement vector will only reduce the RMSD by 20%.

Finally, we can gauge the statistical significance of the observed overlap between the displacement vector and a slow mode. That is, how unlikely is it that a random mode will have a particular overlap with the displacement vector? We approach the question as follows: The set of all unit vectors in a $3N$-dimensional space describes the surface of the unit $3N$-sphere. A subset of these vectors have a projection of magnitude p or greater on the basis vector $\hat{\mathbf{v}}^{(1)}$. The ratio of the area of the $3N$-sphere that is described by this second set to the area of the full $3N$-sphere is the probability that a random vector will have overlap of at least $p : 0 \le p \le 1$ with $\hat{\mathbf{v}}^{(1)}$. The surface area of a $3N$-sphere is given by

$$S_{3N} = \int_0^\pi d\phi_1 \sin^{3N-2}\phi_1 \int_0^\pi d\phi_2 \sin^{3N-3}\phi_2 \ldots \int_0^\pi d\phi_{3N-2}\sin\phi_{3N-2}\int_0^{2\pi} d\phi_{3N-1}\,, \quad (5.21)$$

where the ϕ_is are directional cosines, equivalent to the angles θ and ϕ in spherical polar coordinates. Specifically, the angle ϕ_1 is the angle of departure from the $\hat{\mathbf{v}}^{(1)}$ axis. The projection of a unit vector on the $\hat{\mathbf{v}}^{(1)}$ axis is then $\cos\phi_1$, and for every unit vector $\hat{\mathbf{u}}$: $|\hat{\mathbf{u}} \cdot \hat{\mathbf{v}}^{(1)}| = p$, $\phi_1(\hat{\mathbf{u}}) = \varphi = \arccos p$. The surface area of the $3N$-sphere that is covered by only those vectors $\hat{\mathbf{u}} : |\hat{\mathbf{u}} \cdot \hat{\mathbf{v}}^{(1)}| \ge p$ is given by

$$S^p_{3N} = 2\int_0^\varphi d\phi_1 \sin^{3N-2}\phi_1 \int_0^\pi d\phi_2 \sin^{3N-3}\phi_2 \ldots \int_0^\pi d\phi_{3N-2}\sin\phi_{3N-2}\int_0^{2\pi} d\phi_{3N-1}\,. \quad (5.22)$$

The limit of the integral over ϕ_1 ensures that only a subsurface is integrated over, and the factor of two is included to account for the case when $\hat{\mathbf{u}} \cdot \hat{\mathbf{v}}^{(1)} < 0$. The probability of finding a random vector with projection of at least p on the first mode is the ratio of Eq. 5.22 to Eq. 5.21:

$$P(p, 3N) = \frac{2\int_0^\varphi d\phi_1 \sin^{3N-2}\phi_1}{\int_0^\pi d\phi_1 \sin^{3N-2}\phi_1}\,. \quad (5.23)$$

Eq. 5.23 gives a concise form of the probability of finding some overlap between a displacement vector and a basis vector. Using the recursion relation

$$\int \sin^n x dx = -\frac{1}{n}\sin^{n-1} x \cos x + \frac{n-1}{n}\int \sin^{n-2} x dx \ , \tag{5.24}$$

this function has been plotted against dimensionality, d, for various values of overlap, p in Figure 5.1.

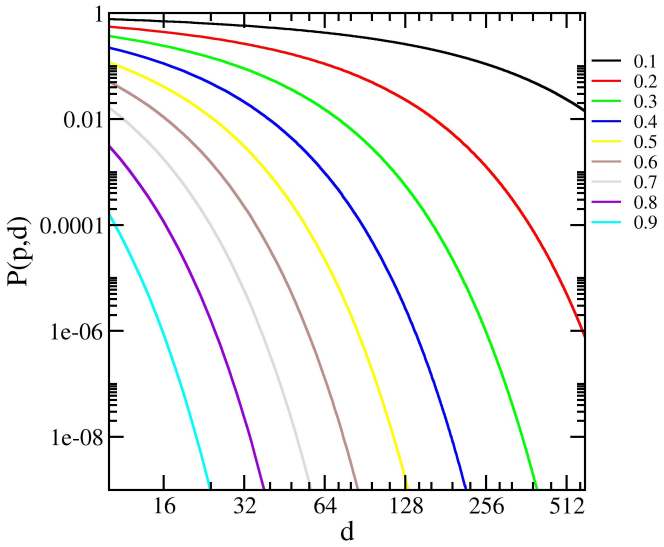


Figure 5.1: Eq. 5.23 plotted for various values of overlap. The dimensionality, $d = 3N$ varies between 10 and 600, corresponding to chains of length 3 to 200 residues. The probability of finding high overlap drops very quickly with increasing protein size.

The probability that a random vector will overlap well with a displacement vector drops quickly with the protein size, particularly for high overlaps. As shown in Figure 5.1, the probability that a purely random global motion of a 200 residue protein has 0.2 overlap with a known deformation vector is about one in a million. An important consideration is that these values apply only to *single* vectors; that is, the probability that the slowest mode has a given overlap with the deformation. Because the modes are orthogonal, the probabilities taken over several modes are not independent. The probability that one of the first ten modes of a 200 residue protein has a 0.2 overlap with the displacement vector is not simply 10 in a million, because the selection of the each mode greatly reduces the pool from which subsequent modes may be selected.

5.3 Modeling with ANM

ANM can be compared to a variety of experimental data, including fluctuations and observed deformations. But what good does it do to repeatedly show that a model compares well with known results? The true power of a model is not in reproducing observations, but in generating novel predictions. For the case of deformations, the ANM might be used to suggest an initial direction for the transition pathway from one state to another. The enterprising scientist may then be able to develop a means of physically blocking this transition pathway and thereby reduce or eliminate the population of the final state. Another possible use of the ANM is to predict unknown conformations of proteins from known conformations. By deforming the structure along the slow modes, one may explore the global motions of the protein and predict new energetic minima. Here we will look at some of the steps necessary to more realistically model protein dynamics with the ANM.

5.3.1 Estimating the size of the force constant

In order to make an accurate model from an ENM, we must first make sure that the scale of the predicted motions agrees with the scale of the motions that we are modeling. In the case of ANM, all dynamics are controlled by two parameters: The cutoff distance r_c and the force constant γ. The shapes of the modes are determined by the cutoff distance, but their magnitudes depend on γ. We have seen above how experimental data such as X-ray B factors or ADPs correlate with ANM-predicted fluctuations, and we will now use the same data to adjust the scale of γ. The isotropic B factors associated with structures solved via X-ray crystallography relate to the MSFs of the individual atoms as shown in Eq. 4.7,

$$B_i = \frac{8\pi^2}{3}\langle(\mathbf{\Delta r}_i)^2\rangle \ ,$$

where $\mathbf{\Delta r}_i$ is the vector displacement of residue i from its average position. The ANM-predicted fluctuations can be calculated from the Hessian pseudoinverse, which has the force constants embedded in it. We define the dimensionless matrix $\boldsymbol{\mathcal{H}} = \frac{1}{\gamma}\mathbf{H}$ to represent the Hessian without explicitly containing the force constant. The fluctuation of residue i is then

$$\langle(\Delta \mathbf{r}_i)^2\rangle = \frac{k_{\mathrm{B}}T}{\gamma}\ \mathrm{Tr}(\boldsymbol{\mathcal{H}}^{\dagger}_{\mathbf{ii}}) \ .$$

Having separated the force constant from the Hessian, we can now find a value that agrees best with experiments. The ANM spring constants that best account for the experimental values are found by minimizing a distance between experimental and theoretical fluctuations. There are at least two methods in use for fitting γ to fluctuation data, and they differ by the distance metric used. The first method [1] minimizes

$$f_1 = \left| \sum_{i=1}^{N} \left(\frac{3b_i}{8\pi^2} - \frac{k_{\mathrm{B}}T}{\gamma}\mathrm{Tr}(\boldsymbol{\mathcal{H}}^{\dagger}_{\mathbf{ii}}) \right) \right| \ , \tag{5.25}$$

where b_i is the B factor of residue i and $\frac{k_B T}{\gamma}\mathrm{Tr}(\boldsymbol{\mathcal{H}}^{\dagger}_{\mathbf{ii}})$ is its ANM-predicted MSF. The quantity f_1 is the total error, or the difference in area under the curves given by experimental and theoretical fluctuations. It is minimized when

$$\gamma = \frac{8\pi^2 k_B T \sum_{i=1}^{N} \mathrm{Tr}(\boldsymbol{\mathcal{H}}^{\dagger}_{\mathbf{ii}})}{3\sum_{i=1}^{N} b_i} \,, \tag{5.26}$$

which might not appeal to some because it does not link individual B factors to individual residue fluctuations. Instead, γ as calculated above depends separately on the sum of B factors and on the sum of ANM predicted fluctuations. If we randomly re-order the B factors, we will wind up with the same force constant. A second distance metric is

$$f_2 = \sum_{i=1}^{N} \left(\frac{3b_i}{8\pi^2} - \frac{k_B T}{\gamma}\mathrm{Tr}(\boldsymbol{\mathcal{H}}^{\dagger}_{\mathbf{ii}}) \right)^2 \,, \tag{5.27}$$

which measures the squared error. This form is minimized when

$$\gamma = \frac{8\pi^2 k_B T}{3} \frac{\sum_{i=1}^{N} [\mathrm{Tr}(\boldsymbol{\mathcal{H}}^{\dagger}_{\mathbf{ii}})]^2}{\sum_{i=1}^{N} \mathrm{Tr}(\boldsymbol{\mathcal{H}}^{\dagger}_{\mathbf{ii}}) b_i} \,. \tag{5.28}$$

Note that the denominator links ANM-predicted fluctuations with B factors for individual residues. We'll use this form because of personal preference. The value obtained for the ENM force constant can vary considerably based on the type of data to which it is fit [2, 3]. For example, MSFs derived from NMR ensembles tend to be larger than those calculated from X-ray B factors, and not just because of temperature differences. A number of studies [4, 5, 6, 7] investigating force constants using MSFs provide values that range from 0.1 to 10 kcal/mol/Å^2.

For structures of sufficiently high resolution, the spatial anisotropy of atomic fluctuations can be determined, and each atom's spatial distribution is described by a set of six ADPs. The ADPs of atom i define the trivariate Gaussian distribution and correspond to the upper-triangle of $\frac{k_B T}{\gamma}\boldsymbol{\mathcal{H}}^{\dagger}_{\mathbf{ii}}$. When fitting force constants using ADPs, the $6N$ components of ANM-predicted fluctuations are taken from the diagonal super-elements of $\mathbf{H}^{\dagger}$, the $6N$ components of experimental fluctuations are the corresponding anisotropic temperature factors from the PDB file, and the force constant that minimizes f_2 is

$$\gamma = k_B T \frac{\sum_{i=1}^{N} \sum_{j=1}^{6} [\boldsymbol{\mathcal{H}}^{\dagger}_{\mathbf{ii}}]_j^2}{\sum_{i=1}^{N} \sum_{j=1}^{6} [\boldsymbol{\mathcal{H}}^{\dagger}_{\mathbf{ii}}]_j b_{ij}} \,. \tag{5.29}$$

Here $[\mathbf{A}]_j$ indicates the j^{th} element of the upper triangle of the 3×3 matrix $\mathbf{A}$, and b_{ij} is its experimental analogue. The formulas in Eqs. 5.28 and 5.29 are valid so long as all of the motion captured by the experiment arises from the internal modes of the molecule. When fitting to B factors or ADPs, the possibility exists that some of the measured fluctuations are accounted for by rigid-body motions, necessitating another approach to finding the force constants from these values [8].

Problem 5.5. *Show that the values of γ given in Eqs. 5.26 and 5.28 respectively minimize the functions f_1 and f_2.*

5.3.2 How big is too big?

Once we have a value for the ANM force constant, we can predict global motions. To do this, we must be able to provide a reasonable size for an excursion along a mode. A simple way to estimate this is to assume that excursions are allowed if they stand a reasonable chance of getting sampled through thermal fluctuations. Invoking equipartition, we will say that each mode of our system contains one $k_{\mathrm{B}}T$ of energy, all of which is in the potential when the amplitude is maximum. Then any $\mathbf{\Delta r}$ that satisfies

$$\frac{1}{2}\widetilde{\mathbf{\Delta r}}\mathbf{H}\mathbf{\Delta r} \leq k_{\mathrm{B}}T$$

represents a conformational state that has a realistic chance of getting sampled. Suppose we wish to deform the structure along eigenvector $\mathbf{v}^{(k)}$ of $\mathbf{H}$ with eigenvalue λ_k. The maximum extent of the deformation, a_k, satisfies

$$\begin{aligned} 2k_{\mathrm{B}}T &= (\widetilde{a_k\mathbf{v}^{(k)}})\mathbf{H}(a_k\mathbf{v}^{(k)}) \\ &= a_k^2\lambda_k \ , \end{aligned}$$

or

$$a_k = \sqrt{2k_{\mathrm{B}}T/\lambda_k} \ .$$

As expected, the magnitude of fluctuations increases with the (square root of) temperature and decreases with the mode frequency. This order-of-magnitude estimate of how large excursions may be is a rule of thumb. It is quite conceivable – in fact, likely – that larger excursions will take place, given enough time. Once we have a Hessian, we can assign a probability for observing the system in any imaginable state, but we don't know exactly how much time it will take before the system samples rare states because the true dynamics of the molecule are damped by solvent. Nonetheless, we can calculate something like transition rates in the context of the ANM.

Suppose that we want to deform a protein a distance d_M along mode i. The harmonic ANM potential tells us that the probability of traveling a distance x along mode i is

$$p_i(x) = \sqrt{\frac{\lambda_i}{2\pi}}\exp\left\{-\frac{\lambda_i x^2}{2}\right\} \ ,$$

where λ_i is the eigenvalue of the Hessian (not of the covariance matrix). The probability of an excursion of at least d_M will be the area under the tails of this distribution,

$$\begin{aligned} p_i(|x| > d_M) &= \int_{-\infty}^{-d_M} p_i(x')\mathrm{d}x' + \int_{d_M}^{\infty} p_i(x')\mathrm{d}x' \\ &= 2\int_{-\infty}^{d_M} p_i(x')\mathrm{d}x' \\ &= \frac{1}{2} + \frac{1}{2}\mathrm{erf}\left(\frac{d_M\sqrt{\lambda_i}}{\sqrt{2}}\right) \ . \end{aligned}$$

We can imagine that if we randomly sampled this distribution, it would take on average $1/p_i(|x| > d_M)$ samples before we drew one with $|x| > d_M$. Assuming that our samples are drawn with the frequency of oscillation along mode i, the period between successive samples is $\tau_i = \sqrt{m_0/\lambda_i}$. The mean first passage time to a state that is a distance d_M along mode i is then

$$\tau = \frac{2m_0}{\sqrt{\lambda_i}\left[1 + \mathrm{erf}\left(\frac{\sqrt{d_M\lambda_i}}{\sqrt{2}}\right)\right]} .$$

Keep in mind that this isn't a rigorous result. It's just a quick example to show how real[1] values can be obtained from ANM results.

There are other factors that need to be considered when looking at the size of an ANM motion. The ANM modes are officially tangents to the direction of motion and should only be believed for displacements that are small enough to be approximated by a tangent vector. An easy way to see this is to consider a structure pivoting about a hinge. If the structure is moving rigidly, the tangent vector will be larger at the tip than near the hinge. Were we to displace the structure along the tangent vector, we would find that it does not maintain its shape, but becomes distorted because the rotational motion is approximated as linear displacements. Song and Jernigan [9] addressed this by defining a new overlap, but basically it is good practice to avoid deformations that distort the structure. Interestingly, every mode except rigid translations will cause the structure to explode if taken to an extreme. This is probably not realistic.

5.3.3 Tip effect and collectivity

Another problem that occurs from time to time in ANM analysis is the *tip effect*. This occurs when some part of a protein – generally a long loop or one of the termini – is extended away from the main mass of the protein. In terms of the elastic network, the distant residues will have fewer contacts than most of the residues in the protein, so the extended "tip" will be poorly constrained. The slowest modes of such systems usually involve wild fluctuations of the tip and often have little to do with the biological function of the protein. The easiest way to address the tip effect is to cut off the offending bit of protein. It's not an elegant solution, but it solves the problem.

An alternative strategy is to select modes based not just on their eigenvalues, but on how much they mobilize global motions in the protein. Although high-frequency motions are almost always localized, low-frequency motions are not always global. The extent to which a mode is "global" can be calculated using well-known quantities, like entropy. The x-, y- and z- components of the motion of residue i from eigenvector $\mathbf{v}$ are v_{3i-2}, v_{3i-1} and v_{3i}, respectively. The mobility that residue i gets from mode k is

$$(\mathbf{\Delta r_i}^{(k)})^2 = (v_{3i-2}^{(k)})^2 + (v_{3i-1}^{(k)})^2 + (v_{3i}^{(k)})^2 .$$

Noting that $\sum_{i=1}^{N}(\mathbf{\Delta r_i}^{(k)})^2 = 1$, the distribution of residue mobilities within a mode is analogous to a discrete probability distribution. We can then define the Shannon entropy of

[1] or real*ish*

mode k as

$$S_k = -\sum_{i=1}^{N} (\mathbf{\Delta r_i}^{(k)})^2 \log(\mathbf{\Delta r_i}^{(k)})^2 \ .$$

S_k is the information entropy of a distribution of residues that are weighted by their mobilities. We can see that S_k is a good measure of how global our mode is. A highly global mode will mobilize all residues to the same extent $(1/N)$, and S_k has its maximum value at $\log N$. This is exactly what we see for rigid-body translations, which are clearly global and also low energy. A completely local mode mobilizes only a single residue while the rest remain fixed, giving a lower limit of $S_k = 0$.

Problem 5.6. *Why can $S_k = 0$ never occur for an internal mode?*

The limits on S_k make it difficult to interpret, so the common measure of the *collectivity* [10] of mode k is

$$\kappa_k = \frac{1}{N} \exp S_k \ .$$

By exponentiating the entropy, we have converted it to an effective number: $\exp S_k$ is the effective number of residues that are mobile in mode k. The collectivity κ_k is then the fraction of residues that are effectively mobilized by mode k. Collectivity is often used as a filter to eliminate localized low-energy motions like the tip effect. The downside of collectivity is that clever researchers can concoct combinations of eigenvalues and collectivity to "choose" modes that seem interesting. That is, one might select the slowest mode with a collectivity above an arbitrary threshold as the mode of interest. There is nothing to stop researchers from finding the mode and then retroactively defining the collectivity threshold. Nothing except research ethics. So there.

Bibliography

[1] Eran Eyal, Lee-Wei Yang, and Ivet Bahar. Anisotropic network model: systematic evaluation and a new web interface. *Bioinformatics*, 22(21):2619–2627, 2006.

[2] Lee-Wei Yang, Eran Eyal, Chakra Chennubhotla, JunGoo Jee, Angela M. Gronenborn, and Ivet Bahar. Insights into equilibrium dynamics of proteins from comparison of nmr and x-ray data with computational predictions. *Structure*, 15(6):741–749, JUN 2007.

[3] Reza Soheilifard, Dmitrii E. Makarov, and Gregory J. Rodin. Critical evaluation of simple network models of protein dynamics and their comparison with crystallographic b-factors. *Phys. Biol.*, 5(2), JUN 2008.

[4] K Hinsen, AJ Petrescu, S Dellerue, MC Bellissent-Funel, and GR Kneller. Harmonicity in slow protein dynamics. *Chem. Phys.*, 261(1-2, SI):25–37, NOV 1 2000.

[5] D Ming and ME Wall. Allostery in a coarse-grained model of protein dynamics. *Phys. Rev. Lett.*, 95(19), NOV 4 2005.

[6] Kei Moritsugu and Jeremy C. Smith. Coarse-grained biomolecular simulation with reach: Realistic extension algorithm via covariance hessian. *Biophys. J.*, 93(10):3460–3469, NOV 2007.

[7] Timothy R. Lezon and Ivet Bahar. Using entropy maximization to understand the determinants of structural dynamics beyond native contact topology. *PLoS Comp. Biol.*, 6(6), JUN 2010.

[8] Timothy R. Lezon. The effects of rigid motions on elastic network model force constants. *Proteins*, 80:1133–1142, 2012.

[9] Guang Song and Robert L. Jernigan. vgnm: A better model for understanding the dynamics of proteins in crystals. *J. Mol. Biol.*, 369(3):880–893, 2007.

[10] Rafael Brüschweiler. Collective protein dynamics and nuclear spin relaxation. *J. Chem. Phys.*, 102(8):3396–3403, 1994.

Chapter 6

Ensemble Analysis

Normal mode analysis (NMA) is a common technique for statistically inferring small motions of complex systems. The idea behind NMA is that equilibrium fluctuations of many-body systems can be explained – up to second order – as the sum of orthogonal modes of vibration. NMA is useful for inferring dynamics near a potential minimum, as is the case with elastic network models or energy-minimized molecular dynamics simulations. It is also useful for characterizing systems dynamics from data, in which case it may take on a different name. When applied to structural ensembles, NMA is usually called principal component analysis (PCA); when applied to molecular dynamics trajectories, it is referred to as essential dynamics (ED). Here the concepts and mathematics behind NMA will be presented in the context of covariance analysis, which provides a somewhat intuitive framework for the theory.

6.1 The one-dimensional ensemble

We will begin with the example of a one dimensional ensemble. This is a distribution of discrete points in space, but we know from probability theory that this sample of points is drawn from an underlying population distribution. We will not concern ourselves with analysis of the population distribution, but will look only at the sample distribution. The difference is subtle. Our sample is a collection of points drawn from some unknown population. We can estimate the population distribution by taking many samples, and our estimate improves with the number of samples that we take. Statisticians are often concerned with finding *unbiased estimators* of the properties of the population distribution. We, on the other hand, will consider that the whole of reality is contained within our data, and we will not try to infer the properties of the unknown population distribution. It really comes down to a matter of preference, and the fact that dividing things by $N - 1$ instead of by N is aesthetically unappealing.

Suppose we have a one dimensional distribution, $\rho(x)$. How do we characterize it? We may start by finding its mean,

$$\mu = \int_{-\infty}^{\infty} \mathrm{d}x \; \rho(x) x = \langle x \rangle \; . \tag{6.1}$$

In practice, this is approximated via sampling:

$$\mu \approx \frac{1}{N}\sum_{i=1}^{N} x_i \ . \tag{6.2}$$

The problem with the mean is that it just gives one point, so it's not very informative. If we want some more detailed information on the distribution, we can proceed to second order and look at the variance,

$$\begin{aligned} \sigma^2 &= \langle (x-\mu)^2 \rangle & (6.3)\\ &= \int_{-\infty}^{\infty} \mathrm{d}x \ \rho(x)(x-\mu)^2 \\ &= \langle x^2 \rangle - \langle x \rangle^2 \ . & (6.4) \end{aligned}$$

In practice, the averages are usually estimated from a sample, giving

$$\sigma^2 \approx \frac{1}{N}\sum_{i=1}^{N} x_i^2 - \left(\frac{1}{N}\sum_{i=1}^{N} x_i \right)^2 \ . \tag{6.5}$$

Higher moments can also be calculated, but we usually stop at second order because it permits neat analytical solutions. In fact, if there are no higher moments to our distribution, then it is Gaussian and subject to the mathematical techniques described in Appendix A.

Problem 6.1. *Show the steps to get from Eq. 6.3 to Eq. 6.4.*

6.2 The two-dimensional ensemble

Before jumping into multivariate distributions we'll consider the two-dimensional case. Consider a space of two variables, x_1 and x_2. Any point in the space can be described using a two-component vector $\mathbf{x} = (x_1 \ x_2)^T$. Now consider the distribution $\rho(\mathbf{x})$ in $\mathbf{x}$. We can calculate the mean of the distribution by taking the probability-weighted sum of all vectors $\mathbf{x}$,

$$\langle \mathbf{x} \rangle = \int_{-\infty}^{\infty} \mathrm{d}x_1 \int_{-\infty}^{\infty} \mathrm{d}x_2 \, \rho(\mathbf{x})\mathbf{x} \ .$$

Because we are integrating a two-component vector over the entire space, the mean is also going to be a two-component vector. We can calculate each component separately, so the mean vector is simply $\langle \mathbf{x} \rangle = (\langle x_1 \rangle \ \langle x_2 \rangle)^T$, where

$$\langle x_i \rangle = \int_{-\infty}^{\infty} \mathrm{d}x_1 \int_{-\infty}^{\infty} \mathrm{d}x_2 \, \rho(\mathbf{x})x_i \ .$$

The variance can be calculated similarly. If we want to find the variance of x_i, we can use

$$\sigma_i^2 = \int_{-\infty}^{\infty} \mathrm{d}x_1 \int_{-\infty}^{\infty} \mathrm{d}x_2 \, \rho(\mathbf{x})(x_i - \langle x_i \rangle)^2 \ ,$$

just like before. The variance σ_i^2 of x_i tells us how much the distribution varies in direction i; however, the presence of more than one variable gives us an additional quantity, the *covariance* between x_i and x_j:

$$\sigma_{ij}^2 = \int_{-\infty}^{\infty} \mathrm{d}x_1 \int_{-\infty}^{\infty} \mathrm{d}x_2 \, \rho(\mathbf{x})(x_i - \langle x_i \rangle)(x_j - \langle x_j \rangle) \ .$$

The covariance σ_{ij}^2 tells us how much x_i and x_j are related. If $\sigma_{ij}^2 = 0$ then x_i and x_j are independent: Knowing x_i tells us nothing about x_j, and vice versa. If $\sigma_{ij}^2 \neq 0$, then the two variables carry some information on each other.

Problem 6.2. *Show that σ_{ij}^2 falls in the range $\left[-\sqrt{\sigma_i^2 \sigma_j^2}, \sqrt{\sigma_i^2 \sigma_j^2}\right]$.*

Problem 6.3. *Show that $\sigma_{ij}^2 = \langle x_i x_j \rangle - \langle x_i \rangle \langle x_j \rangle$.*

The variances and covariances typically appear in a matrix, called the *variance-covariance* matrix. We will just call it a *covariance matrix*, because a variance is just a special case of a covariance. The matrix in two dimensions looks like this:

$$\mathbf{C} = \begin{bmatrix} \sigma_{11}^2 & \sigma_{12}^2 \\ \sigma_{21}^2 & \sigma_{22}^2 \end{bmatrix} \ .$$

The covariance matrix is sometimes given the name $\mathbf{\Sigma}$, but we will call it $\mathbf{C}$ (for covariance) to avoid confusion with the summation symbol. Note that this matrix is symmetric: $\sigma_{21}^2 = \sigma_{12}^2$. The diagonal elements are squares of real numbers and are therefore positive-valued. They are also greater than or equal to the off-diagonal elements (see Problem 6.2), so $\mathbf{C}$ is non-negative definite.

In practice, we will not know the true form of $\rho(\mathbf{x})$, but will have to estimate it from a sample set. The first point is $\mathbf{x}^{(1)} = (x_1^{(1)} \ x_2^{(1)})^T$, the second is $\mathbf{x}^{(2)} = (x_1^{(2)} \ x_2^{(2)})^T$, and so on. We can gather all of these vectors into a single $2 \times N$ matrix, $\mathbf{X} = (\mathbf{x}^{(1)} \ \ldots \ \mathbf{x}^{(N)})$. This is our data matrix. In terms of $\mathbf{X}$, the components of mean vector are

$$\mu_i = \frac{1}{N} \sum_{k=1}^{N} X_{ik} \ , \tag{6.6}$$

and the components of the covariance matrix are

$$\begin{aligned} C_{ij} &= \frac{1}{N} \left(\mathbf{X} \tilde{\mathbf{X}} \right)_{ij} - \left[\frac{1}{N} \sum_{k=1}^{N} X_{ik} \right] \left[\frac{1}{N} \sum_{p=1}^{N} X_{jp} \right] \\ &= \frac{1}{N} \left(\mathbf{X} \tilde{\mathbf{X}} \right)_{ij} - \mu_i \mu_j \ . \end{aligned} \tag{6.7}$$

It's often convenient to center our data so that $\boldsymbol{\mu} = 0$, which will provide us with a matrix $\mathbf{\Delta X}$ of the differences from the average, with components given by

$$(\Delta X)_{ij} = X_{ij} - \mu_i \ .$$

Mean-centering the data like this removes the second term from Eq. 6.7 and lets us write the covariances as

$$\begin{aligned} C_{ij} &= \frac{1}{N}\left(\mathbf{\Delta X}\widetilde{\mathbf{\Delta X}}\right)_{ij} \\ &= \langle \Delta x_i \Delta x_j \rangle \ . \end{aligned}$$

Once again, the covariances are just the expected products of displacements of the coordinate variables from their average values. The diagonals are the mean-squared fluctuations of the variables.

6.3 Analysis of distributions of n dimensions

Our findings for the two-dimensional distribution can be generalized to distributions of n dimensions. The mean vector $\boldsymbol{\mu}$ and covariance matrix $\mathbf{C}$ for an n-dimensional distribution has components given by Eqs. 6.6 and 6.7. The only conceptual leap (more of a hop) that needs to be taken to go from the two-dimensional case described above to the general n-dimensional case is to understand that $\boldsymbol{\mu}$ has n components and $\mathbf{C}$ is $n \times n$.

We have seen in Chapter 1 that symmetric matrices like $\mathbf{C}$ can be decomposed with the orthogonal transformation

$$\mathbf{C} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T \ ,$$

where $\mathbf{V}$ is an $n \times n$ orthogonal matrix of eigenvectors and $\mathbf{\Lambda}$ is a diagonal matrix of eigenvalues. The matrix $\mathbf{V}$ is taken to be dimensionless, so $\mathbf{\Lambda}$ has the same units as $\mathbf{C}$, i.e., distance squared. Using Eq. 6.7,

$$\begin{aligned} \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T &= \frac{1}{N}\mathbf{\Delta X}(\mathbf{\Delta X})^T \\ \mathbf{\Lambda} &= \frac{1}{N}\mathbf{V}^T\mathbf{\Delta X}(\mathbf{\Delta X})^T\mathbf{V} \\ &= \frac{1}{N}\mathbf{\Delta Z}(\mathbf{\Delta Z})^T \ , \end{aligned}$$

where we have defined the matrix $\mathbf{\Delta Z} \equiv \mathbf{V}^T\mathbf{\Delta X}$. So $\mathbf{V}$ transforms $\mathbf{C}$ into new coordinates that have only variances, and no covariances. What this means is that if we were to take our distribution and look at it along the direction of $\mathbf{v}^{(i)}$, the i^{th} eigenvector (given by the i^{th} column of $\mathbf{V}$, we would see a variance of λ_i. We would also see that the fluctuations in the direction of $\mathbf{v}^{(i)}$ are totally uncorrelated with those along any of the other directions. The same goes for all other eigenvectors. Diagonalizing the covariance matrix enables us to define a set of n independent directions, such that fluctuations along any one of these directions are independent of the fluctuations along all of the others. The eigenvectors of $\mathbf{C}$ are called the *principal components* of the distribution.

When only the mean vector and covariance matrix are known (and here we have conveniently chosen to ignore all higher moments), the distribution can be represented as a Gaussian,

$$\rho(\mathbf{x}) = (2\pi)^{-m/2}|\mathbf{C}|^{-1/2}\, \mathrm{e}^{-\frac{1}{2}(\widetilde{\mathbf{x}-\boldsymbol{\mu}})\mathbf{C}^{-1}(\mathbf{x}-\boldsymbol{\mu})} \ .$$

Defining $\mathbf{\Delta z} \equiv \mathbf{V}^T(\mathbf{x} - \boldsymbol{\mu})$ and $Z = (2\pi)^{m/2}|\mathbf{C}|^{1/2}$, this can be re-written as

$$\begin{aligned}\rho(\mathbf{\Delta z}) &= (2\pi)^{-m/2}|\mathbf{\Lambda}|^{-1/2}\, \mathrm{e}^{-\frac{1}{2}\widetilde{\mathbf{\Delta z}}\mathbf{\Lambda}^{-1}\mathbf{\Delta z}} \\ &= \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi\lambda_i}} \mathrm{e}^{-(\Delta z_i)^2/2\lambda_i} \\ &= \prod_{i=1}^{m} \rho(\Delta z_i) \ ,\end{aligned}$$

or just the product of probabilities of the individual variables Δz_i.

6.4 Structural alignment

We conclude our discussion of ensembles at the beginning. The first step of ensemble analysis is structural alignment, which removes rotational and translational degrees of freedom from the ensemble. The reason for doing this is that we don't want the relative positions of ensemble members to influence our analysis. All that we are interested in is how the structures in our ensemble vary in shape, not how they vary in spatial location or orientation.

Let's take a second to explore aligning structures using the Kabsch method [1, 2]. Nobody really knows how this works anymore, because there is software that does it for you in any language that you use. Even when there isn't software, the method is generally laid out as an algorithm rather than a mathematically sound solution for optimimally aligning structures. But you should still see it somewhere, and it might as well be here.

Consider the $3 \times N$ matrices $\mathbf{A^0}$ and $\mathbf{B^0}$, the columns of which contain the C_α coordinates for an N residue protein. The structures differ by some amount, either because of different orientations in space or because of some small changes in residue positions relative to one another. Our goal is to align the structures such that we minimize the RMSD between them. In terms of normal modes, we seek to remove any differences in the structures that can be accounted for by zero modes. We are allowed to move the structures along these modes, as such changes carry no energetic cost.

The first step is to eliminate translations between the two structures by making sure that their centers of mass coincide. This is easily accomplished by calculating the center of mass of $\mathbf{A^0}$, with components

$$a_i^{CM} = \frac{1}{N}\sum_{k=1}^{N} A_{ik}^0 \ ,$$

and subtracting it from each column of $\mathbf{A^0}$,

$$A_{ij} = A_{ij}^0 - a_i^{CM} \ ,$$

and then doing likewise for $\mathbf{B^0}$. Now $\mathbf{A}$ and $\mathbf{B}$ both have their mass centers at the origin, leaving us with only the task of finding the rotation that will minimize the RMSD between them. For this purpose, we will assume that $\mathbf{B}$ is fixed and that $\mathbf{A}$ will be rotated to a position $\mathbf{A'} = \mathbf{RA}$ that minimizes its RMSD with $\mathbf{B}$. The rotation is accomplished with the

3×3 matrix $\mathbf{R}$, the components of which we seek to determine. Calling f the RMSD, we have

$$f = \sum_{i=1}^{N} \sum_{j=1}^{3} \left[B_{ji} - A'_{ji} \right]^2 \tag{6.8}$$

as our function to minimize. Differentiating with respect to each element of $\mathbf{R}$, we find

$$\begin{aligned} \frac{\partial f}{\partial R_{\alpha\beta}} &= -2 \sum_{i=1}^{N} \sum_{j=1}^{3} \left[B_{ji} - A'_{ji} \right] \frac{\partial A'_{ij}}{\partial R_{\alpha\beta}} \\ &= -2 \sum_{i=1}^{N} \sum_{j=1}^{3} \left[B_{ji} - A'_{ji} \right] \delta_{j\alpha} A_{\beta i} \\ &= -2 \sum_{i=1}^{N} \left[B_{\alpha i} - A'_{\alpha i} \right] A_{\beta i} \\ &= -2(\mathbf{B}\tilde{\mathbf{A}})_{\alpha\beta} + 2(\mathbf{R}\mathbf{A}\tilde{\mathbf{A}})_{\alpha\beta} \ . \end{aligned}$$

Setting this equal to zero (which we really shouldn't do, for reasons that will be explained shortly), we find

$$\mathbf{B}\tilde{\mathbf{A}} = \mathbf{R}\mathbf{A}\tilde{\mathbf{A}} \ , \tag{6.9}$$

or

$$\mathbf{R} = \mathbf{B}\tilde{\mathbf{A}}(\mathbf{A}\tilde{\mathbf{A}})^{-1} \ . \tag{6.10}$$

A detail that has been skipped over is that $\mathbf{R}$ is not just any matrix, but strictly a rotation. It therefore must satisfy the orthogonality condition $\mathbf{R}\tilde{\mathbf{R}} = \tilde{\mathbf{R}}\mathbf{R} = \mathbf{1}$. We may have imposed this constraint using Lagrangian multipliers, but instead we just found a closed form of the unconstrained $\mathbf{R}$. Let's impose the constraint now so that we can get $\mathbf{R}$ into a form that enables easy manipulation by a computer. First, we will define the matrix

$$\mathbf{C} \equiv \mathbf{B}\tilde{\mathbf{A}} \ ,$$

which contains covariances of positions in $\mathbf{A}$ and $\mathbf{B}$. Then Eq. 6.10 becomes

$$\mathbf{R} = \mathbf{C}(\mathbf{A}\tilde{\mathbf{A}})^{-1} \ .$$

Imposing the orthogonality constraint $\tilde{\mathbf{R}}\mathbf{R} = \mathbf{1}$ yields

$$\mathbf{1} = (\mathbf{A}\tilde{\mathbf{A}})^{-1}\tilde{\mathbf{C}}\mathbf{C}(\mathbf{A}\tilde{\mathbf{A}})^{-1} \ ,$$

where we used the matrix identity $(\mathbf{A}^T)^{-1} = (\mathbf{A}^{-1})^T$. Multiplying from both sides by $\mathbf{A}\tilde{\mathbf{A}}$ gives

$$(\mathbf{A}\tilde{\mathbf{A}})^2 = \tilde{\mathbf{C}}\mathbf{C} \ .$$

Putting this back into Eq. 6.4 gives

$$\mathbf{R} = \mathbf{C}(\tilde{\mathbf{C}}\mathbf{C})^{-1/2} \ . \tag{6.11}$$

Writing the optimal rotation as a function of $\mathbf{C} \equiv \mathbf{B}\tilde{\mathbf{A}}$ alone enables us to further simplify it using SVD. If we decompose $\mathbf{C}$ as $\mathbf{C} = \mathbf{U}\mathbf{\Sigma}\tilde{\mathbf{V}}$, Eq. 6.11 becomes $\mathbf{R} = \mathbf{U}\tilde{\mathbf{V}}$. In numerical implementation, usually an extra step is taken to ensure the proper handedness of the coordinate system. This is required to remove the ambiguity associated with the sign of each eigenvector.

Problem 6.4. *What happens if the system is left-handed?*

Problem 6.5. *Show that* $\mathbf{R} = \mathbf{C}(\tilde{\mathbf{C}}\mathbf{C})^{-1/2}$ *reduces to* $\mathbf{R} = \mathbf{U}\tilde{\mathbf{V}}$*, where the columns of* $\mathbf{U}$ *and* $\mathbf{V}$ *are, respectively, the left and right singular vectors of* $\mathbf{C}$*.*

This same technique is applied whether two or multiple structures are aligned. In the case of multiple structures, all structures are iteratively aligned until convergence. To align multiple structures, one is first selected as the key, and all other structures are pairwise aligned to it using the method described above. After aligning all structures, the average structure is calculated from the entire ensemble. In the next iteration, all structures are pairwise aligned to this ensemble average, and a new average is calculated. The two-step process of aligning all structures to the ensemble average followed by updating the ensemble average continues until the average does not change by some small amount. Multiple structure alignments are important in studies of structural ensembles.

Bibliography

[1] Wolfgang Kabsch. A soluton for the best rotation to relate two sets of vectors. *Acta Crystallographica*, 32:922, 1976.

[2] Wolfgang Kabsch. A discussion of the solution for the best rotation to relate two sets of vectors. *Acta Crystallographica*, A34:827–828, 1978.

Appendix A

Gaussian Distributions

If there's one distribution that everyone should know, it is the Gaussian distribution. Its bell-shaped curve is found throughout science, often as a result of the central limit theorem. Here we are interested in the Gaussian distribution mostly because we are looking at pairwise interactions. This will hopefully become clear momentarily.

A.1 The one-dimensional Gaussian distribution

In one dimension, the probability density at point x given the Gaussian distribution with mean μ and standard deviation σ is

$$\rho(x|\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\} . \tag{A.1}$$

You should memorize this equation, because Gaussian distributions are ubiquitious in this life. The quantity $p(x|\mu,\sigma)\mathrm{d}x$ is the probability that a point randomly drawn from this distribution will fall in an interval of width $\mathrm{d}x$ centered at x. Note that Eq. A.1 is normalized:

$$1 = \int_{-\infty}^{\infty} \mathrm{d}x\, \rho(x|\mu,\sigma) . \tag{A.2}$$

This is fun to show mathematically. Functions that give the probability density of a point given a distribution are appropriately called *probability density functions*, or PDFs. Often the dependencies on μ and σ will be dropped from PDFs for simplicity. There are many fascinating properties of Gaussian distributions, but here we will make use of the fact that they are characterized by two values, μ and σ.

Problem A.1. *Perform the integral in Eq. A.2. This can be done by defining distributions in two dimensions, x and y, and then changing to cylindrical coordinates.*

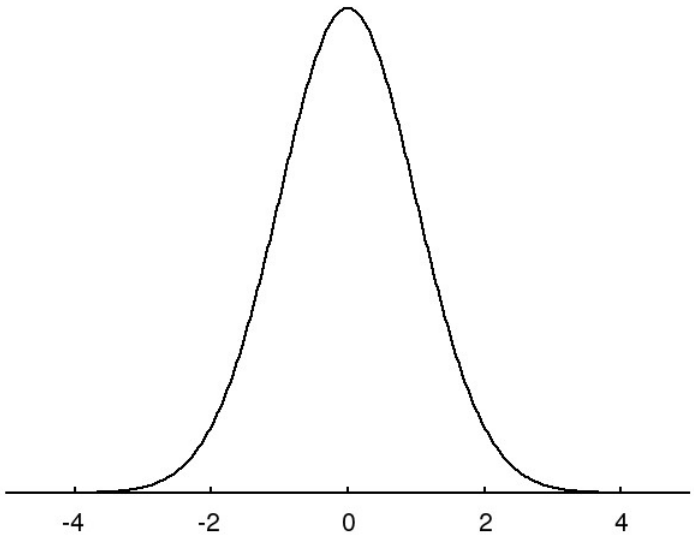


Figure A.1: The Gaussian distribution.

A.1.1 Moments of the Gaussian distribution

We have just asserted that the Gaussian distribution of Eq. A.1 has mean μ and variance σ^2(or standard deviation σ), so we should show these explicitly. First the mean:

$$\begin{aligned}\langle x\rangle &= \int_{-\infty}^{\infty} \mathrm{d}x \rho(x) x \\ &= \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} \mathrm{d}x \, \mathrm{e}^{-\frac{(x-\mu)^2}{2\sigma^2}} x \ .\end{aligned}$$

Changing variables inside the integral to $y = x - \mu$,

$$\begin{aligned}\langle x\rangle &= \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} \mathrm{d}y \, \mathrm{e}^{-\frac{y^2}{2\sigma^2}} (y+\mu) \\ &= \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} \mathrm{d}y \, \mathrm{e}^{-\frac{y^2}{2\sigma^2}} y + \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} \mathrm{d}y \, \mathrm{e}^{-\frac{y^2}{2\sigma^2}} \mu \ .\end{aligned}$$

The first integral is over an odd function of y (meaning that $f(-y) = -f(y)$), so the integral from $-\infty$ to zero cancels the integral from zero to $+\infty$. We are left only with the second integral, which, from Eq. A.2, evaluates to the constant

$$\langle x\rangle = \mu \ . \tag{A.3}$$

Moving on to $\langle x^2 \rangle$, we can try the same $y = x - \mu$ substitution,

$$\begin{aligned}
\langle x^2 \rangle &= \int \mathrm{d}x\, \rho(x) x^2 \\
&= \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} \mathrm{d}x\, \mathrm{e}^{-\frac{(x-\mu)^2}{2\sigma^2}} x^2 \\
&= \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} \mathrm{d}y\, \mathrm{e}^{-\frac{y^2}{2\sigma^2}} (y+\mu)^2 \\
&= \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} \mathrm{d}y\, \mathrm{e}^{-\frac{y^2}{2\sigma^2}} (y^2 + \mu^2 + 2y\mu) \\
&= \mu^2 + \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} \mathrm{d}y\, \mathrm{e}^{-\frac{y^2}{2\sigma^2}} y^2 + \frac{2\mu}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} \mathrm{d}y\, \mathrm{e}^{-\frac{y^2}{2\sigma^2}} y \qquad \text{(A.4)} \\
&= \mu^2 + \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} \mathrm{d}y\, \mathrm{e}^{-\frac{y^2}{2\sigma^2}} y^2 \,, \qquad \text{(A.5)}
\end{aligned}$$

where we have found the integral over the odd function of y to be zero. The remaining integral is even in y and does not simply disappear. We will use an old physics trick of adding to the exponential a *field*, h that is linear in y. We define the generating function

$$Z(h) = \int_{-\infty}^{\infty} \mathrm{d}y\, \mathrm{e}^{-\frac{y^2}{2\sigma^2} + hy} \,. \qquad \text{(A.6)}$$

Differentiating $Z(h)$ with respect to h allows us to pull down powers of y from the exponent. Because h is only being used for the purpose of taking derivatives, it can take any value we wish, including zero. At the end of our calculations, we'll set $h = 0$ to remove the influence of the field from the end result; however, this in no way prevents us differentiating with respect to it. First we look for a closed form of Eq. A.6. This can be found by completing the square in the exponent and making the substitution $z = y - \sigma h$. Now

$$\begin{aligned}
Z(h) &= \mathrm{e}^{h^2/2\sigma^2} \int_{-\infty}^{\infty} \mathrm{d}z\, \mathrm{e}^{-z^2/2\sigma^2} \\
&= \frac{1}{\sqrt{2\pi\sigma^2}}\, \mathrm{e}^{h^2/2\sigma^2} \,,
\end{aligned}$$

where we have used Eq. A.1 in the second line. Using the second derivative with respect to h, we can find

$$\begin{aligned}
\langle y^2 \rangle &= \frac{1}{Z(h)} \frac{\partial^2}{\partial^2 h} Z(h) \bigg|_{h=0} \\
&= \sigma^2 \,.
\end{aligned}$$

Along with Eqs. A.3 and A.5, this leads us to

$$\sigma^2 = \langle x^2 \rangle - \langle x \rangle^2 \,. \qquad \text{(A.7)}$$

The equations for μ and σ^2 are usually just memorized, but it is always fun to work through the steps. A fine point here is that the mean and variance of any distribution are referred to as μ and σ^2, respectively, so our result may seem trivial and the derivation unnecessary. Keep in mind that μ and σ are parameters in the Gaussian distribution, so we have just provided them with meaning beyond simply constants in an equation. We can very quickly write down the mean and variance of any distribution that has the form of Eq. A.1, without performing any calculations.

A.2 The multivariate Gaussian distribution

The above can be extended to multiple dimensions. In n dimensions, the mean is a vector $\boldsymbol{\mu} = (\mu_1 \dots \mu_n)^T$ and the variance is a matrix, called the *variance-covariance matrix*, or just the *covariance matrix*,

$$\mathbf{C} = \begin{bmatrix} C_{11} & \cdots & C_{1n} \\ \vdots & \ddots & \vdots \\ C_{n1} & \cdots & C_{nn} \end{bmatrix} .$$

Here the diagonal element C_{ii} denotes the variance of component i, which can also be denoted as σ_i^2. The off-diagonal element C_{ij} is the co-variance between components i and j. This matrix is real, symmetric and positive semi-definite.

A multivariate Gaussian distribution in n dimensions is defined by an n-component mean vector $\boldsymbol{\mu}$ and a symmetric $n \times n$ covariance matrix $\mathbf{C}$. Its PDF takes the form

$$\rho(\mathbf{x}) = (2\pi)^{-n/2} |\mathbf{C}|^{-1/2} \mathrm{e}^{-\frac{1}{2}\widetilde{(\mathbf{x}-\boldsymbol{\mu})}\mathbf{C}^{-1}(\mathbf{x}-\boldsymbol{\mu})} . \tag{A.8}$$

The normalization factor $(2\pi)^{-n/2}|\mathbf{C}|^{-1/2}$ plays the same role as the $\sigma\sqrt{2\pi}$ in the univariate case. We can see this by integrating our multivariate distribution over the n-dimensional space. This little exercise will also show us how the symmetric form of $\mathbf{C}$ makes this easy. In fact, we will just use the multivariate forms of the same tricks that we used in the previous section.

First, we will introduce some shorthand for the integral over all n dimensions:

$$\int \mathrm{d}^n\mathbf{x} = \int_{-\infty}^{\infty} \mathrm{d}x_1 \dots \int_{-\infty}^{\infty} \mathrm{d}x_n .$$

This just allows us to save space by treating the individual variables and the limits of integration implicitly. Now we define $\mathbf{y} = \mathbf{x} - \boldsymbol{\mu}$. Note that $\mathrm{d}^n\mathbf{y} = \mathrm{d}^n\mathbf{x}$. We will also replace $(2\pi)^{-n/2}|\mathbf{C}|^{-1/2}$ with $1/Z$, so we can surprise ourselves at the end of the calculation.

$$\begin{aligned} \int \mathrm{d}^n\mathbf{x}\, \rho(\mathbf{x}) &= \frac{1}{Z} \int \mathrm{d}^n\mathbf{x}\, \mathrm{e}^{-\frac{1}{2}\widetilde{(\mathbf{x}-\boldsymbol{\mu})}\mathbf{C}^{-1}(\mathbf{x}-\boldsymbol{\mu})} \\ &= \frac{1}{Z} \int \mathrm{d}^n\mathbf{y}\, \mathrm{e}^{-\frac{1}{2}\tilde{\mathbf{y}}\mathbf{C}^{-1}\mathbf{y}} . \end{aligned}$$

To proceed, we will change coordinates to a basis in which $\mathbf{C}^{-1}$ is diagonal, turning the integrand into a product of univariate Gaussians. The covariance matrix is decomposed using an orthogonal transformation, as shown in Eq. 1.3,

$$\mathbf{C} = \mathbf{V}\mathbf{\Lambda}\tilde{\mathbf{V}} \; . \tag{A.9}$$

More details about the significance of this decomposition will follow, but for now it will suffice that Eq. A.9 implies that

$$\mathbf{C}^{-1} = \mathbf{V}\mathbf{\Lambda}^{-1}\tilde{\mathbf{V}} \; .$$

Note that $|\mathbf{V}| = \pm 1$, which will appear below as the Jacobian for changing coordinates from $\mathbf{y}$ to $\mathbf{z} = \tilde{\mathbf{V}}\mathbf{y}$. Now to find Z:

$$\begin{aligned}
\int \mathrm{d}^n\mathbf{x}\, \rho(\mathbf{x}) &= \frac{1}{Z} \int \mathrm{d}^n\mathbf{y}\, \exp\left\{ -\frac{1}{2} \tilde{\mathbf{y}}\mathbf{V}\mathbf{\Lambda}^{-1}\tilde{\mathbf{V}}\mathbf{y} \right\} \\
&= \frac{1}{Z} \int \mathrm{d}^n\mathbf{z}\, |\mathbf{V}|\, \exp\left\{ -\frac{1}{2} \tilde{\mathbf{z}}\mathbf{\Lambda}^{-1}\mathbf{z} \right\} \; .
\end{aligned}$$

Expanding the product in the exponent into a sum over components,

$$\begin{aligned}
\int \mathrm{d}^n\mathbf{x}\, \rho(\mathbf{x}) &= \frac{1}{Z} \int \mathrm{d}^n\mathbf{z}\, \exp\left\{ -\frac{1}{2} \sum_{i,j=1}^{n} \delta_{ij} \frac{z_i z_j}{\lambda_i} \right\} \\
&= \frac{1}{Z} \int \mathrm{d}^n\mathbf{z}\, \exp\left\{ -\frac{1}{2} \sum_{i=1}^{n} \frac{z_i^2}{\lambda_i} \right\} \\
&= \frac{1}{Z} \prod_{i=1}^{n} \int \mathrm{d}z_i\, \exp\left\{ -\frac{z_i^2}{2\lambda_i} \right\} \; ,
\end{aligned}$$

we find that the integral in n dimensions reduces to the product of n one-dimensional integrals. Finding the solution from Eqs. A.1 and A.2 gives

$$\begin{aligned}
\int \mathrm{d}^n\mathbf{x}\, \rho(\mathbf{x}) &= \frac{1}{Z} \prod_{i=1}^{n} \sqrt{2\pi\lambda_i} \\
&= \frac{1}{Z} \left(2\pi\right)^{n/2} |\mathbf{C}|^{1/2} \; ,
\end{aligned}$$

where we have made use of the fact that the determinant of a matrix is invariant under orthogonal transformations (i.e., $|\mathbf{\Lambda}| = |\mathbf{C}|$. Finally,

$$Z = \left(2\pi\right)^{n/2} |\mathbf{C}|^{1/2} \; , \tag{A.10}$$

agreeing with Eq. A.8. This calculation made use of a nice feature of multivariate Gaussian distributions: They are just products of univariate Gaussian distributions and can be integrated easily by separation of variables using eigendecomposition of the covariance matrix.

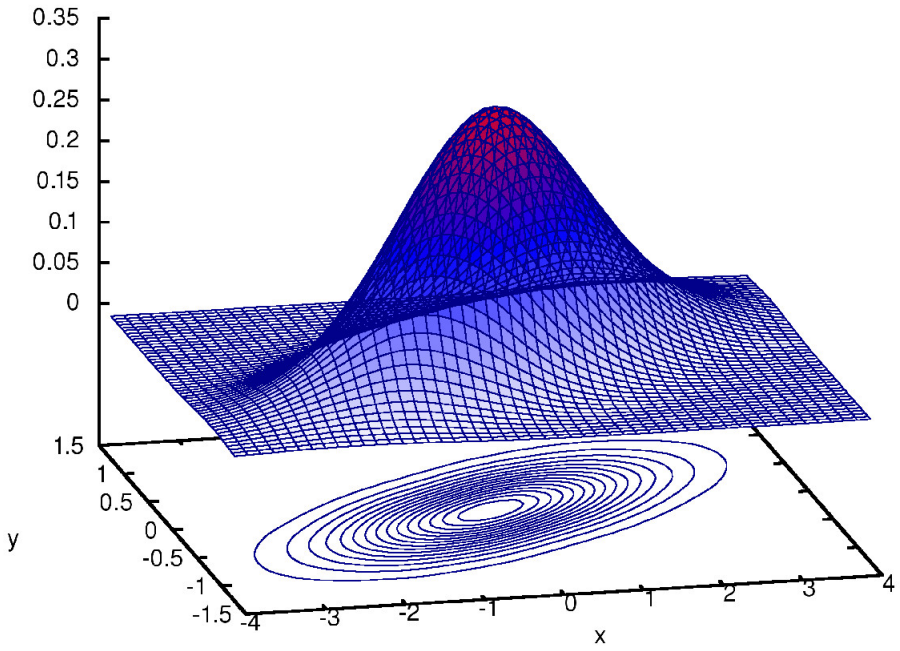


Figure A.2: A two dimensional Gaussian distribution with contours showing that surfaces of equal probability are elliptical. The principal componenents (eigenvectors) of the distribution point along the major and minor axes of the contour ellipses.

A.2.1 Moments of the multivariate Gaussian distribution

We can now calculate some expectation values from the multivariate Gaussian distribution. We'll start by looking for the expected value of one of its variables,

$$\begin{aligned}
\langle x_i \rangle &\equiv \int \mathrm{d}^n\mathbf{x}\, \rho(x) x_i \\
&= (2\pi)^{-n/2} |\mathbf{C}|^{-1/2} \int \mathrm{d}^n\mathbf{x}\, \mathrm{e}^{-\frac{1}{2}(\widetilde{\mathbf{x}-\boldsymbol{\mu}})\mathbf{C}^{-1}(\mathbf{x}-\boldsymbol{\mu})} x_i \\
&= (2\pi)^{-n/2} |\mathbf{C}|^{-1/2} \int \mathrm{d}^n\mathbf{y}\, \mathrm{e}^{-\frac{1}{2}(\tilde{\mathbf{y}})\mathbf{C}^{-1}\mathbf{y}} (y_i + \mu_i) \\
&= (2\pi)^{-n/2} |\mathbf{C}|^{-1/2} \left[\int \mathrm{d}^n\mathbf{y}\, \mathrm{e}^{-\frac{1}{2}(\tilde{\mathbf{y}})\mathbf{C}^{-1}\mathbf{y}} y_i + \int \mathrm{d}^n\mathbf{y}\, \mathrm{e}^{-\frac{1}{2}(\tilde{\mathbf{y}})\mathbf{C}^{-1}\mathbf{y}} \mu_i \right] .
\end{aligned}$$

Just like in the 1D case, we end up with two integrals: One containing a y_i and the other containing a μ_i. The integral over y_i vanishes because it is odd, leaving only the constant μ_i, or

$$\langle x_i \rangle = \mu_i \ . \tag{A.11}$$

Covariances can be calculated using the same "field" trick that we used for the one-

dimensional case. Explicitly,

$$\begin{aligned}
\langle x_i x_j \rangle &= \int d^n\mathbf{x}\, \rho(\mathbf{x}) x_i x_j \\
&= \frac{1}{\sqrt{(2\pi)^n |\mathbf{C}|}} \int \mathrm{d}^n\mathbf{x}\, \mathrm{e}^{-\frac{1}{2}\widetilde{(\mathbf{x}-\boldsymbol{\mu})}\mathbf{C}^{-1}(\mathbf{x}-\boldsymbol{\mu})} x_i x_j \\
&= \frac{1}{\sqrt{(2\pi)^n |\mathbf{C}|}} \int \mathrm{d}^n\mathbf{y}\, \mathrm{e}^{-\frac{1}{2}\tilde{\mathbf{y}}\mathbf{C}^{-1}\mathbf{y}} (y_i + \mu_i)(y_j + \mu_j) \\
&= \frac{1}{\sqrt{(2\pi)^n |\mathbf{C}|}} \int \mathrm{d}^n\mathbf{y}\, \mathrm{e}^{-\frac{1}{2}\tilde{\mathbf{y}}\mathbf{C}^{-1}\mathbf{y}} (y_i y_j + \mu_i\mu_j + y_i\mu_j + y_j\mu_i) \\
&= \mu_i\mu_j + \frac{1}{\sqrt{(2\pi)^n |\mathbf{C}|}} \int \mathrm{d}^n\mathbf{y}\, \mathrm{e}^{-\frac{1}{2}\tilde{\mathbf{y}}\mathbf{C}^{-1}\mathbf{y}} y_i y_j \ . \qquad \text{(A.12)}
\end{aligned}$$

The integral can be solved by defining the generating function

$$Z(\mathbf{J}) = \int \mathrm{d}^n\mathbf{y}\, e^{-\frac{1}{2}\tilde{\mathbf{y}}\mathbf{C}^{-1}\mathbf{y} + \tilde{\mathbf{J}}\mathbf{y}} \ , \qquad \text{(A.13)}$$

which contains the *vector field*, $\mathbf{J}$. It works just like the field h worked in the calculation of $\langle x^2 \rangle$ from the 1D Gaussian. We say $\mathbf{z} = \mathbf{y} - \mathbf{C}\mathbf{J}$, and

$$\begin{aligned}
Z(\mathbf{J}) &= \mathrm{e}^{\frac{1}{2}\tilde{\mathbf{J}}\mathbf{C}\mathbf{J}} \int \mathrm{d}^n\mathbf{z}\, \mathrm{e}^{-\frac{1}{2}\tilde{\mathbf{z}}\mathbf{C}^{-1}\mathbf{z}} \\
&= (2\pi)^{\frac{n}{2}} \sqrt{|\mathbf{C}|}\, \mathrm{e}^{\frac{1}{2}\tilde{\mathbf{J}}\mathbf{C}\mathbf{J}} \ ,
\end{aligned}$$

yielding

$$\begin{aligned}
\langle y_i y_j \rangle &= \frac{1}{Z(\mathbf{J})} \frac{\partial^2}{\partial J_i \partial J_j} Z(\mathbf{J}) \bigg|_{\mathbf{J}=0} \\
&= C_{ij} \ .
\end{aligned}$$

Along with Eqs. A.11 and A.12, this leads us to

$$C_{ij} = \langle x_i x_j \rangle - \langle x_i \rangle \langle x_j \rangle \ . \qquad \text{(A.14)}$$

The covariance between variables i and j is the difference between their average product and the product of their averages. In Gaussian distributions, these covariances are hard-coded into the covariance matrix that appears in the exponent.

If we transform to the coordinate system $\mathbf{Z} = \tilde{\mathbf{V}}\mathbf{X}$, then motions along the basis vectors are uncorrelated. Or, defining $\boldsymbol{\Delta}\mathbf{z} \equiv \mathbf{z} - \tilde{\mathbf{V}}\boldsymbol{\mu}$,

$$\begin{aligned}
\rho(\boldsymbol{\Delta}\mathbf{z}) &= (2\pi)^{-n/2} |\boldsymbol{\Lambda}|^{-1/2} \exp\left\{ -\frac{1}{2} \widetilde{\boldsymbol{\Delta}\mathbf{z}} \boldsymbol{\Lambda}^{-1} \boldsymbol{\Delta}\mathbf{z} \right\} \\
&= (2\pi)^{-n/2} |\boldsymbol{\Lambda}|^{-1/2} \prod_{i=1}^{n} \exp\left\{ -\frac{(\Delta z_i)^2}{2\lambda_i} \right\} \ .
\end{aligned}$$

The distribution is independently Gaussian in each coordinate.

Appendix B

Geometric Series

Where did Eq. 2.11 come from? It is just the simplification of a geometric series that can be derived easily as follows: We write the series for a higher power, expand it, rearrange terms, cancel, and viola!

$$\begin{aligned}
\sum_{k=0}^{n} k^3 &= \sum_{k=0}^{n} (k+1)^3 - (n+1)^3 \\
&= \sum_{k=0}^{n} k^3 + 3\sum_{k=0}^{n} k^2 + 3\sum_{k=0}^{n} k + \sum_{k=0}^{n} 1 - (n+1)^3 \\
3\sum_{k=0}^{n} k^2 &= (n+1)^3 - 3\sum_{k=0}^{n} k - \sum_{k=0}^{n} 1 \\
&= n^3 + 3n^2 + 3n + 1 - \frac{3n(n+1)}{2} - n - 1 \\
&= n^3 + \frac{3n^2}{2} + \frac{n}{2} \\
\sum_{k=0}^{n} k^2 &= \frac{n}{6}(n+1)(2n+1) \\
\sum_{k=1}^{n-1} k^2 &= \frac{n}{6}(n-1)(2n-1)
\end{aligned}$$