CS5238 Combinatorial Methods in Bioinformatics 2004/2005 Semester 1

Lecture 5 : Geometric and Kinematic Models of Proteins

Lecturer: Jean-Claude Latombe

Scribes: Ng Chuming, Rong Guodong, Koh Yeow Nam

1 Introduction

A protein is a complex, high molecular weight organic compound that consists of a chain of amino acids that are joined together by bonds called peptide bonds. In molecular biology, this compound has gained the focus of several researchers as they are the *workhorses* of all living organisms, performing several vital functions, which includes

- Catalysis and inhibitions of reactions
- Transport of molecules between extracellular space and the cells themselves, as well as within the cells
- Building blocks for muscles for movement
- Storage of energy
- Signal transduction that are necessary for cell-cell interactions and cell processes such as growth and apoptosis
- Defense against foreign intruders

1.1 Proteins

The *Central Dogma of Molecular Biology* is a concept which states that hereditary information is carried by DNA[1] molecules. This information is passed down from an organism to its offspring by *replication*, which produces an identical DNA molecule, during cell division. To use the encoded information, mRNA[2] must be *transcribed* from the DNA molecules where it will be subsequently *translated* by ribosomes to form proteins. The proteins will then, depending on their amino acid sequences, fold into highly compact structures[6] which will determine their functions and hence the *phenotype* of the organism.

[1] Deoxyribonucleic Acid

[2] messenger Ribonucleic Acid

1.2 What's in a protein?

Amino acids are biochemical building blocks. An amino acid is a molecule that contains an amino(NH_2) group and a carboxylic acid($COOH$) group that are attached to the same tetrahedral carbon atom, also known as the α-carbon. The properties, and hence the identities, of the amino acids are determined by distinct R-groups, which are also attached to the α-carbon. Figure 1 shows the chemical composition of two such amino acids - alanine and valine, and one can see that the only difference between them is the chemical makeup of the R-group component.

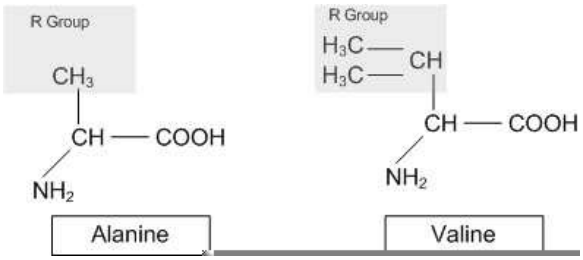

Figure 1: Chemical composition of Alanine and Valine

Amino acid molecules can bind to each other and their bonds are called *peptide bonds*. This bond is formed when the carboxylic acid group from one amino acid binds to the amino group of another amino acid, releasing a water(H_2O) molecule as the by-product. Hence in this manner, the amino acids can form chains, called polypeptides. Long chains of polypeptides are known as proteins, with each protein containing dozens to thousands of amino acids, also known as residues in this context.

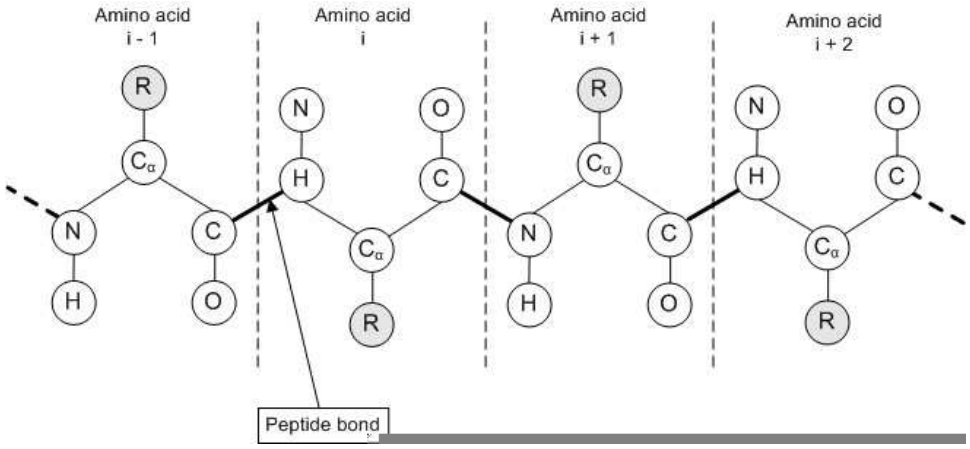

Figure 2: Amino acids and their peptide bonds that make up the protein molecule

There are over 500 amino acids that have been found in nature, but only 20 of them are relevant to the makeup of mammalian proteins. They are coded in the standard *genetic code* and are called proteinogenic.

1.3 Protein Structure

Several conformations of the amino acids in a protein are possible due to the rotation of the chain around the bonds. These conformations are responsible for the three dimensional structures and hence, the functions, of the proteins. A protein has multiple levels of structure, from primary to quaternary.

Primary The primary structure of a protein is basically the order of its amino acids. This order is often written from the amino end to the carboxyl end (much like how we write the DNA sequence from the 5' to the 3' end). A single change in the amino acid sequence can have a profound effect in the overall structure and the function of the protein.

Secondary Secondary structure refers to certain repeating features found in proteins. These features are a result of hydrogen bonding between neighboring amino acids. The two main secondary structures are the *α-helix* and the *β-sheet* (See Figure 3).

The α-helix is the most abundant type of secondary structure in proteins. Its formation is spontaneous and is stabilized by the hydrogen bonding between the amide nitrogen and the carbonyl carbon of peptide bonds that are spaced four residues apart, producing a helical coil-like structure. The β-sheets are composed of two or more different regions of stretches that are at least five to ten amino acids. The folding and alignment of the polypeptide backbone aside one another is stabilized by the hydrogen bonds between the amid nitrogen and carbonyl carbons.

Other than the α-helixes and the β-sheets, there are two more secondary structures - *loops* and *coils*. Loops are regions of the protein chain that are between α-helixes and β-sheets. They are of various lengths and three dimensional configurations. Loops can interact with the surrounding aqueous environment as well as other proteins. Lastly, any region of a secondary structure that is not a helix, sheet or recognizable turn is then commonly referred to as a coil.

Tertiary The tertiary structure of a protein refers to the complete three-dimensional structure of its polypeptide residues. A protein assumes its tertiary structure by folding.

Quartenary Quartenary structure is only present if there is more than one protein. With multiple polypeptide chains, the quartenary structure is their interconnections and organization. Examples are haemoglobin and phospholipid kinase, which are made up of multiple protein subunits.

1.4 Motion of Proteins in the Folded State

The tertiary structure that a protein assumes to carry out its physiological role in a cell is also known as its *native state* or *native conformation*. From its primary

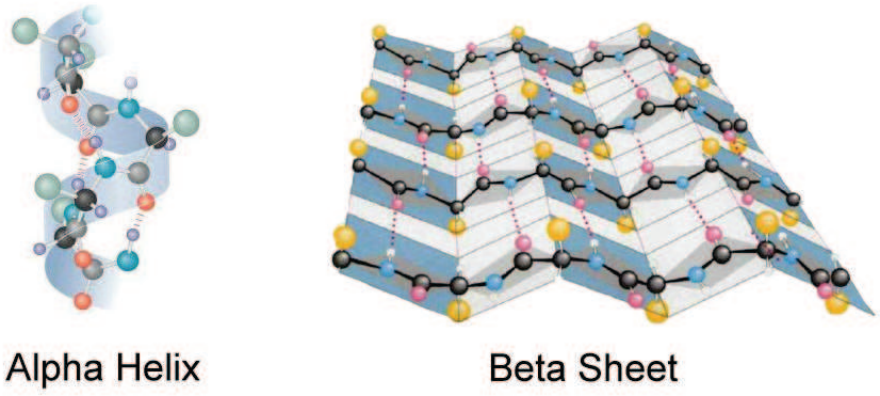

Figure 3: The secondary structures of a protein

structure, there are many pathways, or intermediate states[6] that the protein can take before achieving its native state. Certain disease are caused by misfolding in these intermediate states, such as Alzheimer and Creutzfeldt-Jakob disease[8]. But generally, the movement of the atoms in the molecule and hence the tertiary structure of the protein is stable.

2 Representing Proteins

In order to study the salient structural, kinetic and even chemical properties of proteins, a form of representation is needed so as to make it amenable to computer modeling and simulation, and ultimately, prediction. There are several ways of modeling protein molecules, depending on the feature of interest. These include geometric and kinetic modeling.

2.1 Geometric Models of Bio-Molecules

In this section we shall explain the various ways to model and represent the shape of bio-molecules, protein molecules being one of them. This form of modeling is useful in answering queries such as which atoms in a bio-molecule are close to a given atom. It can also aid in the computation of surface areas to estimate the amount of interaction with the solvent such as the water molecules. More importantly, it can be used to find shape features such as cavities which could act as binding sites for other ligands or protein molecules.

2.1.1 Hard Sphere Model

The most common approach to geometric modeling of bio-molecules is to represent each atom as a ball with fixed radius and its position relative to other atoms in the molecule is fixed[9]. This radius is taken to be the *Van der Waals* radius of the atom. This radius is defined by the Van der Waals potential, which is the consequence of induced polarization i.e. formation of electric dipoles, in atoms when they are close to

each other. These forces are weak, except at close range. The surface that is formed by the union of these spheres is also known as the Van der Waals surface.

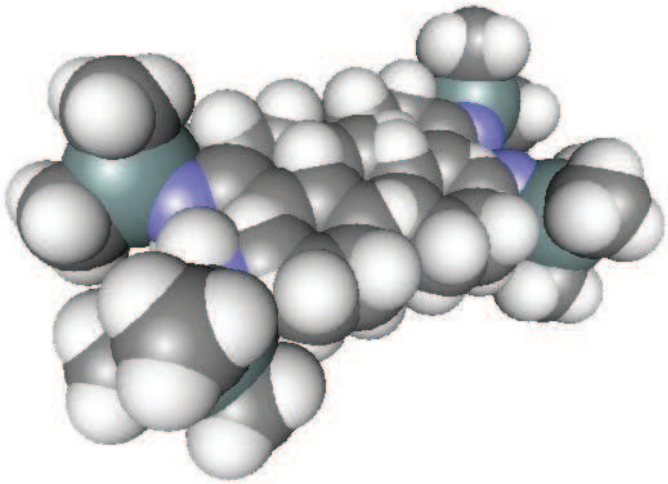

Figure 4: A Hard Sphere model of a bio-molecule

The Van der Waals potential is often described by a potential energy function and the most commonly used potentials are the Lennard-Jones(12-6) potential and the Exponential-6 potential[5]. The Lennard-Jones(12-6) potential is given by the expression

$$V_{12-6}(r) = 4\epsilon_{XY}\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right]$$

for the interaction potential between a pair of atoms[1]. This potential has an attractive tail at large r (i.e. when two atoms are a distance apart from each other), reaches a minimum at around 1.122σ and is strongly repulsive at a shorter distance, increasing steeply as r is decreased further (i.e. When two atoms are forced too close to each other). The graph of the potential is shown in figure 5. ϵ_{XY} and σ are empirical parameters. The term $\sim 1/r^{12}$ dominating at short distance, models the repulsion between atoms when they are brought very close to each other while the term $\sim 1/r^{6}$, dominating at large distance, constitute the attraction part. One limitation of this potential is that it cannot represent situations with open shells, where strong localized bonds may form (as in covalent system), or where there is a delocalized "sea of electrons" where the ions sit (such as in metals). But for bio-molecules, the Van der Waals potential is a sufficient representation. Table 1 shows the Van der Waals radius (in Ångstrøm) for some elements.

Element	H	C	N	O	F	P	S	Cl
Van der Waals radius (Å)	1.2	1.7	1.5	1.4	1.35	1.9	1.85	1.8

Table 1: Van der Waals radii of some elements

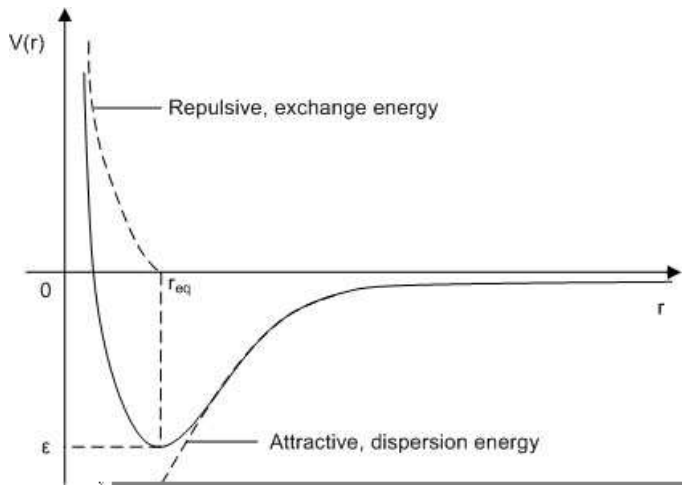

Figure 5: The Van der Waals potential

2.1.2 Solvent Accessible Surface and Molecular Surface

The Van der Waals surface is one way of representing the molecule surface. Another model would be the *Solvent-Accessible Surface*. It takes into account the surface of the molecule which can come into contact with the molecules of the solvent, such as water molecules. The solvent molecules (or probes) are usually represented by single spheres and the accessible surface is the closed surface traced by a solvent molecule as it goes through all the possible positions in which the two molecules are in surface contact. This surface is traced by the center of the probe molecule. The *Molecular Surface model*[7] is similar. It defines a surface composed of many surface patches, some of which are defined by the surface of the macromolecule, while others come from the surface of the solvent molecule in positions where the surface contact is established along the closed curve, not only at one point.

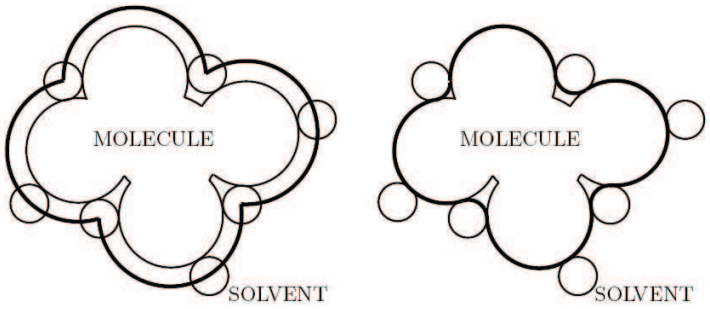

Figure 6: Solvent accessible model (left) and the Molecular surface model (right)

2.1.3 Computation of Hard-Sphere Surface

With a method of representation, the next step is to consider the computational complexity of calculating the surface area. Usually, implementing geometric algorithms is a difficult task. There is always a huge gap in the algorithms as they are

described in theoretical papers and their implementation in software. In a paper by Halperin and Shelton[4], they proposed a such that computation of molecular surface only takes $\Theta(n)$ time.

That scheme exploits the fact that the maximum number of spheres that is intersecting any sphere is a constant. The data structure for such information could be in the form of a hashtable[3]. Each entry will represent a sphere and associated with it is a bin, storing links to its intersecting spheres. Looking up the entries will take constant time (since there is a maximal constant to the number of links in the bin). Hence to compute each atom's contribution to the molecular surface only requires $O(1)$ time (Refer to Section 2.1.4). And assuming that there are n number of atoms, to compute the molecular surface will take $\Theta(n)$ time.

2.1.4 Trapezoidal Decomposition

A trapezoidal decomposition of a polygon is formed by drawing a vertical line segment up and down each vertex of a polygon, extending it until it reaches the polygon boundary. Using this method, a collection of trapezoids will be obtained.

Similarly, the surface of molecules can be subdivided. This subdivision (or arrangement) of a sphere, s, is induced by the circles of intersection of other atoms with s. The arrangement is constructed incrementally by adding one circle at a time and maintaining the trapezoidal decomposition of the current arrangement. As mentioned above, the maximum number of circles that a sphere can have is a constant.

A collection of circles (caused by intersection) on a sphere s will induce a partitioning of the sphere into vertices, edges and faces. We can apply trapezoidal decomposition to make each face homeomorphic to a disc that has at most *four* edges on its boundary. To do this, two antipodal points are declared as *poles*. Great circles through the poles are called *polar circles* and their arcs are called *polar arcs*. Also, any point on the little circles that are tangent to the polar arcs are called *polar tangencies*. For every polar tangency we extend a polar arc in either direction until it hits the poles or another little circle. This step is repeated for any points of intersection between the small circles.

The result is a set of surfaces, each containing at most four edges, that contributes to the molecular surface. To compute the molecular surface area, we just have to repeat for each of the n molecules and hence the total complexity for computing the molecular surface area takes $\Theta(n)$ time.

2.2 Kinematic Models of Bio-Molecules

Kinematics is the study of movement independent of the forces that cause them. Another term related to it is dynamics, which considers the force. The goal of kinematics is to analytically describe the position and the movement of objects.

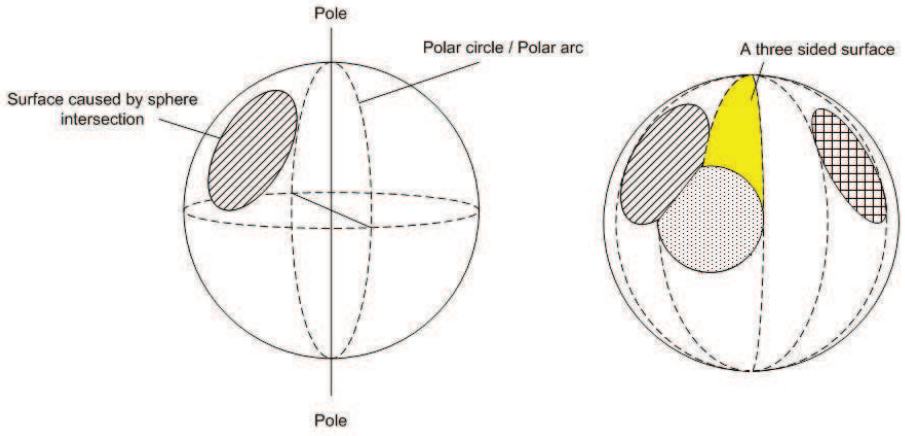

Figure 7: Sphere with the poles and polar arcs (left). A surface that is created as a result of trapezoidal decomposition (right)

When we consider bio-molecules, the objects of interest are the atoms that composed the molecules. Kinematic models of bio-molecules describe their positions and movement. There are two main kinematic models:

1. **Atomistic model**: In this model, the position of atoms are described directly by their 3-D coordinates. So, in this model, p atoms require $3p$ parameters.

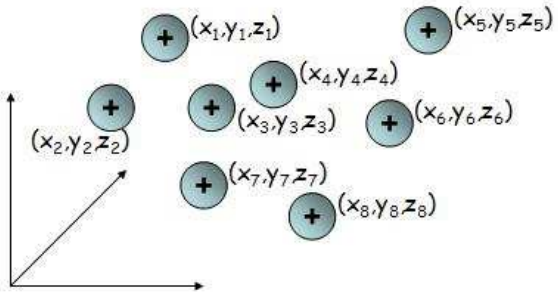

Figure 8: Atomistic model

   The drawback of this model is it only records the positions of the atoms. It does not record the information of bonds between them.

2. **Linkage model**: In this model, the position of atoms are described by *internal* parameters. These parameters include: bond length, bond angle and torsion angle. In this model, under certain assumptions, p atoms only require *p-3* parameters. However, there is no possibility of fine-tuning in this model.

Following, we will explain how to build a linkage model. Before that, we will first introduce how to using homogenous coordinate matrix to represent rigid-body transformation.

2.2.1 Rigid-body transform

Rigid-body transform give a one-to-one mapping form one set of points to another set of points in Euclidean space, while all the distance between any pair of points remain unchanged. There are many methods to represent such transformations. Here, we will introduce one widely used method – *homogenous coordinate matrix*. In the next section, we will talk about another more efficient way – *quaternion*.

1. **2-D case**
   In 2-D Euclidean space, if we want to rotate (counterclockwise) a point (a, b) by an angle θ to a new point (a', b'), we can simply multiply a rotation matrix $\mathbf{R}$ to the point.

$$\mathbf{R} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$$

$$\begin{pmatrix} a' \\ b' \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix}$$

   If we want first rotate a point by an angle θ, then translate it along a vector $t = [t_x, t_y]^T$, we can first multiply the rotation matrix $\mathbf{R}$, then add translation vector t.

$$\begin{pmatrix} a' \\ b' \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix}$$

   However, this method uses multiplication and addition together. So we cannot easily combine many transformation into one multiplication. Homogenous coordinate can solve this problem. When using homogenous coordinate, a 2-D point (x, y) can be represented by a vector $(x, y, 1)$. Generally, a vector (x, y, w) represents a 2-D point $(x/w, y/w)$.

   So the rotation and translation can be represented by one homogenous coordinate matrix as following (figure 9):

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta & t_x \\ \sin\theta & \cos\theta & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} x * \cos\theta - y * \sin\theta + t_x \\ x * \sin\theta + y * \cos\theta + t_y \\ 1 \end{pmatrix}$$

   When using homogenous coordinate, we can easily combine many transformations. For example, if we want to do n transformations, we can just multiply all the transformation matrices in order.

   If vectors $\vec{i} = (i_1, i_2)^T$ and $\vec{j} = (j_1, j_2)^T$ are the two supporting vectors of the new coordinate system, we can rewrite the transform matrix as following:

$$\mathbf{R} = \begin{pmatrix} i_1 & j_1 & t_x \\ i_2 & j_2 & t_y \\ 0 & 0 & 1 \end{pmatrix}$$

2. **3-D case**
   Similarly, in 3-D Euclidian space, if the supporting vectors of the new coordinate

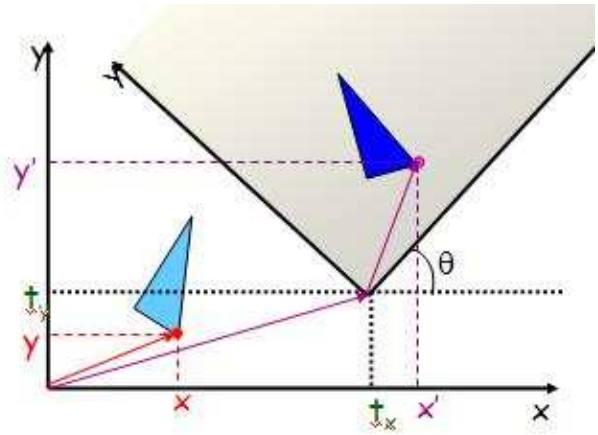

Figure 9: 2D rigid-body transformation

system are $\vec{i} = (i_1, i_2, i_3)^T, \vec{j} = (j_1, j_2, j_3)^T, \vec{k} = (k_1, k_2, k_3)^T$ and the origin of the new coordinate system is at (t_x, t_y, t_z), we can use following transform matrix:

$$\mathbf{R} = \begin{pmatrix} i_1 & j_1 & k_1 & t_x \\ i_2 & j_2 & k_2 & t_y \\ i_3 & j_3 & k_3 & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

2.2.2 Serial linkage model

At first, we will explain how to build a serial linkage model. In a serial linkage model, all the atoms are assumed to be linked to one another to form a long chain.

We first place the first atom at an arbitrary position in space. Then, we place the second atom anywhere at bond length. That means no matter where the second atom is, the distance between the first atom and the second atom is same to the bond between them. In the third step, we place the third atom any where according to the bond length and the bond angle. That means, the distance between the second atom and the third atom is same to the bond between them, and the angle formed by the first three atoms is same to the bond angle.

After having placed the first three atoms (we call them atom -2, -1, 0), we can use them to build a coordinate frame. The origin is at the atom 0; x axis point to atom 1; y axis is in the plane defined by atom -2,-1,0, so that the y-coordinate of the atom -2 is positive. z axis is built according to right-hand rule.

Then we can put the fourth atom, atom 1, into this coordinate system. We first place atom 1 at $(d, 0, 0)$, where d equals to the bond length between atom 0 and 1 (figure 10). Then we rotate atom 1 around z axis by the bond angle β (figure 11). At last, we rotate atom 1 around x axis by the torsion angle τ (figure 12).

After inserted the atom 1, we can build another coordinate system using atom -1,0,1, then insert atom 2. By using this method, we can insert all the atoms one by one. Atom $i+1$ is placed in the coordinate system decided by atom $i, i-1, i-2$ (figure 13). We call this coordinate system "frame i". We denote the transformation matrix of atom $i+1$ in frame i by $\mathtt{T}_i$. If we want to get the transformation matrix of atom k ($k > i$) in frame i (denoted as $\mathtt{T}_k^{(i)}$), we can just multiply all the matrices from $\mathtt{T}_{i+1}$ to $\mathtt{T}_k$.

$$\mathtt{T}_k^{(i)} = \mathtt{T}_k \cdots \mathtt{T}_{(i+2)}\mathtt{T}_{(i+1)}$$

If the matrix $\mathtt{T}_{j+1}(i < j < k)$ is changed to $\mathcal{T}_{j+1}$, we can update $\mathtt{T}_k^{(i)}$ to $\mathcal{T}_k^{(i)} = \mathtt{T}_j^{(i)}\mathcal{T}_{j+1}\mathtt{T}_k^{(j+1)}$.

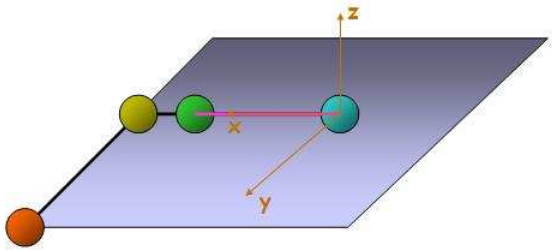

Figure 10: Place atom 1 at $(d, 0, 0)$

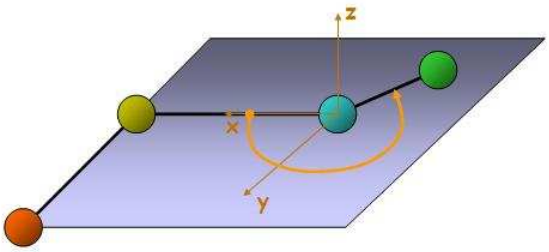

Figure 11: Rotate atom 1 around z axis by bond angle β

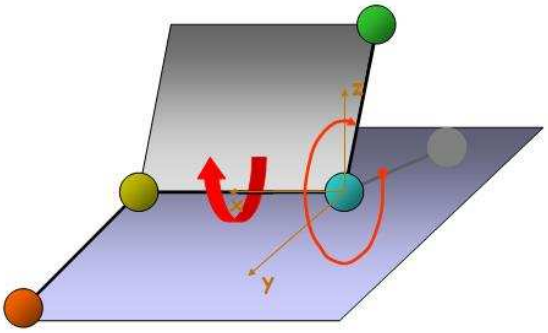

Figure 12: Rotate atom 1 around z axis by bond angle τ

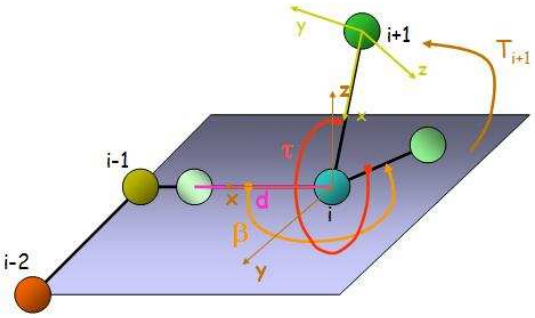

Figure 13: Place atom $i+1$

2.2.3 Tree-shaped linkage model

Since almost no real molecules are totally serial, we have to use tree-shaped linkage model to represent real molecules. A tree-shaped linkage model can be represented as figure 14. The root node represents the first three atoms. All the other nodes represent the atoms inserted after the first frame is built, one node for one atom.

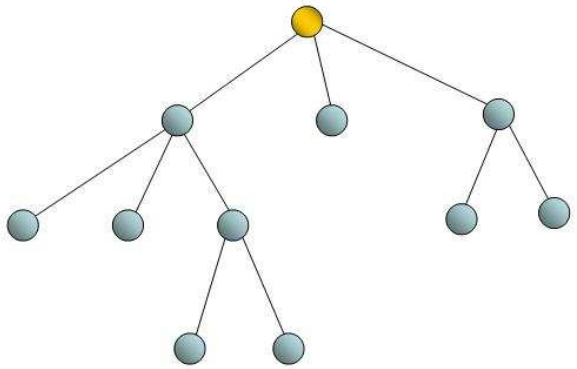

Figure 14: Tree-shaped linkage model

The building process of the tree-shaped linkage model is similar to the process of the serial linkage model. First, we place the first three atoms and build frame 0. Then we insert all the atoms connected to atom 0. These atoms are the children of the root node. Then we insert the grand children, and so on.

Remember that when using atomistic model, p atoms require $3p$ parameters. Here, by using tree-shaped linkage model, p atoms require $3p - 6$ parameters. The reason why we decrease of 6 parameters is due to the fact that when we place the first three atoms, we only use three parameters (one for the atom -1 and two for the atom 0). However, in the atomistic model, these three atoms require nine parameters.

Moreover, we can further decrease the number of parameters in the linkage model. We can assume the bond lengths and bond angles are constant. Thus, for every atom, we can use only one parameter – the torsional angle. By doing so, as we have mentioned before, we can use $p - 3$ parameters to represent p atoms.

In protein molecule, the sequence of $C_\alpha - C - N - ...$ is the backbone of the protein. We often call the torsional angle between $C_\alpha - N$ the ϕ angle, the torsional angle between $N - C$ the ω angle, and the torsional angle between $C - C_\alpha$ the ψ angle. Because ω angle often equal to $180°$ (figure 15), we only record ϕ angle and ψ angle.

There are also many side-chains connected to the backbone of protein. The torsional angle of these side-chains are called χ angles. This model that uses only the three torsional angles is known as $\phi - \psi - \chi$ linkage model of protein. (See figure 16)

In different amino acid molecule, the number of different χ angles varies. There are at most four χ angles in one amino acid molecule. For example, Arginine has four χ angles (figure 17).

Although the linkage model use much less parameters than the atomistic model, it still has some drawbacks. Since we assume the bond lengths and bond angles are constant, there is no possibility of fine-tuning. Moreover, because the position of

every later atom rely on the position of its three predecessors, small local changes may have big global effects. That means the errors in the linkage model accumulate. And forces are difficult to express in linkage model.

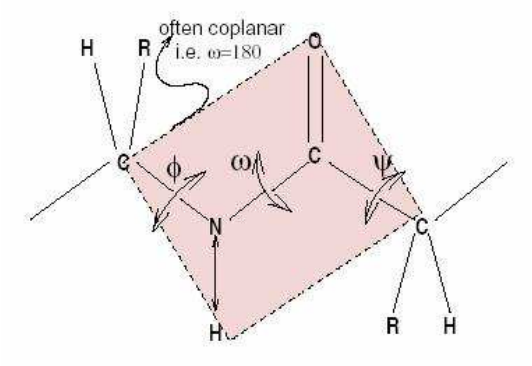

Figure 15: ϕ, ω, ψ angles

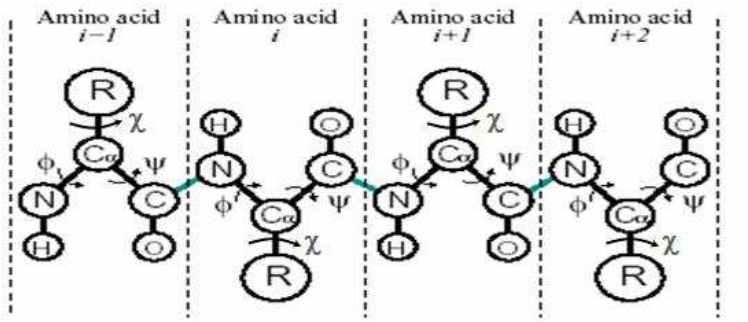

Figure 16: $\phi - \psi - \chi$ linkage model of protein

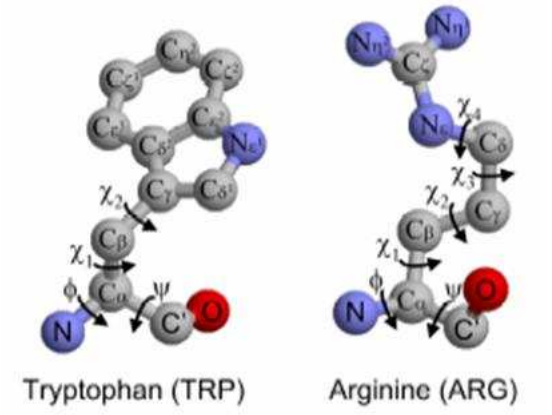

Figure 17: χ angles in different amino acid molecule

3 Quaternions

Quaternions was first invented in the nineteenth century by William Rowan Hamilton as a generalization of complex numbers into a class of so called *hypercomplex* numbers. It has been widely accepted then that real numbers have a geometric interpretation of defining points on a *line*, and that complex numbers define points lying on the *plane*. Quaternions was born as a result of the attempt to generalize the above concepts to higher dimensions.

3.1 Preliminaries

To set the stage for a discussion of quaternions, we first review the familiar concept of a complex number. To represent a complex number, one usually denote it in the form

$$a + bi,$$

but it should be highlighted that such notation is misleading since a *real* and an *imaginery* number cannot directly added together arithmetically. A better representation of a complex number is to denote it by an *ordered pair* of reals

$$(a, b),$$

together with a set of operations such as addition and multiplication defined on them.

With knowledge of complex numbers, it seems natural then, to speculate on the possibility an extension of the number system such that its members may be interpreted as points in 3-space representable by *triplets*. The simplest extension imaginable would be to have numbers of the form

$$a + bi + cj = (a, b, c),$$

where i and j are distinct and independent square roots of -1. However, it turns out that while addition and subtraction find its common analogy, an algebra of such triplet pairs cannot capture multiplication in the same flavour as it is defined for complex numbers. We shall demonstrate why this is so in the following paragraphs.

In the algebra of complex numbers, multiplication acquires meaningful geometric interpretation that is most apparent when we look at them in the *polar* representation. In polar form, for a complex number $z = a + bi$, we have

$$\begin{aligned} r &= \sqrt{a^2 + b^2} \\ \theta &= tan^{-1}(\frac{b}{a}) \end{aligned}$$

where r is the 'modulus' and θ is the argument, Then, z is representable as

$$z = (r\cos(\theta), r\sin(\theta)).$$

Geometrically, the mulitplication of two complex numbers z_1 and z_2 is characterized by a scaling and a rotation about the origin, resulting in the new point

$$z' = z_1 z_2(r_1 r_2 \cos(\theta_1 + \theta_2), r_1 r_2 \sin(\theta_1 + \theta_2)).$$

which has modulus $r_1 r_2$ and argument $\theta_1 + \theta_2$.

However, in three dimensions, two parameters (an *azimuthal* angle θ and a *polar* angle ϕ) are required to determine the axis for a rotation, one to specify the angle of rotation and a final one for scaling of the length. This would in total require *four* parameters whereas ordered triplets only have three. Interestingly, a resolution of this problem is revealed through the attempt to extend the concept of complex conjugates to triplets.

Complex conjugates. The conjugate of a complex number $z = a + bi$ is given by $\bar{z} = a - bi$. For complex numbers, the property of multiplication is such that the result of a complex number with its conjugate is always a real number,

$$z\bar{z} = a^2 + b^2$$

which is also the square of the modulus.

Let us now examine an extension of conjugation to number triplets. We have

$$z = a + bi + cj \qquad \text{and} \qquad \bar{z} = a - bi - cj.$$

On multiplying we get

$$z\bar{z} = a^2 + b^2 + c^2 - 2ijbc,$$

which contain an extra imaginery product term $-2ijbc$. An obvious but wrong attempt to remove this term is to have the axiom $ij = 0$, since it results in the following contradiction

$$ij.ij = i^2.j^2 = (-1)(-1) = 1$$

Hamilton's insight in resolving this problem is to regard the product term $-2ijbc$ as two separate terms $-ijbc$ and $-jibc$. If we abandon the commutativity of multiplication and define $ij = -ji$, then the product term $-2ijbc$ disappears. It turns out that if we forego commutativity, we can create a number system whereby quadruples of numbers are the natural objects and at the same time retain all the familiar operators of complex numbers. The fourth number forming each quadruple arise from the fact that it is possible to derive the value of ij using the associativity of multiplication as follows

$$ij.ij = i(ji)j = -i(ij)j = -i^2 j^2 = -(-1)(-1) = -1;$$

The above implies that $(ij)^2 = -1$, which means that ij is yet another root of -1. If we denote this third root as k, then the extension to quadruples of number is straightforward.

$$a + bi + cj + dk = (a, b, c, d)$$

Using the above formulation, we then have the following basic relationships

$$ij = k, jk = i, ki = j, ji = -k, kj = -i, ik = -j, i^2 = j^2 = k^2 = ijk = -1$$

Hamilton termed these extended numbers *quaternions*.

3.2 Basic Properties of Quaternions

In this section, we present the basic properties and fundamental operations of quaternion arithmetic. In addition to the previous representations, quaternions can also be represented as consisting of a real part a and an imaginary part $bi + cj + dk$ which is written as a vector (b, c, d). This is because i, j and k act like orthogonal unit vectors. So, a quaternion z is written in this alternative representation as

$$q = (a, \boldsymbol{v}), \qquad \text{where } \boldsymbol{v} = (b, c, d)$$

Addition and Subtraction. The operations of addition and subtraction are similar to that of component wise operations of complex numbers:

$$\begin{aligned} q &= (a, b, c, d) = a + bi + cj + dk \\ p &= (x, y, z, w) = x + yi + zj + wk \\ q \pm p &= (a \pm x, b \pm y, c \pm z, d \pm k) = (a \pm x) + (b \pm y)i + (c \pm z)j + (d \pm w)k \\ &= (a, \boldsymbol{v}) \pm (x, \boldsymbol{u}) = (a \pm x, \boldsymbol{v} \pm \boldsymbol{u}) \end{aligned}$$

Scalar Multiplication. Multiplying by a real number results in a component wise scaling:

$$\begin{aligned} q &= (a, b, c, d) = a + bi + cj + dk \\ xq = qx &= (xa, xb, xc, xd) = xa + (xb)i + (xc)j + (xd)k \\ &= (xa, x\boldsymbol{v}) \end{aligned}$$

Conjugation. The the conjugate of a quaternion $z = a + bi + cj + dk$ is defined as

$$\begin{aligned} \bar{z} &= a - bi - cj - dk \\ &= (a, -\boldsymbol{v}) \end{aligned}$$

Absolute Value. The absolute value is similar to that of a complex number:

$$|q| = \sqrt{a^2 + b^2 + c^2 + d^2} = \sqrt{q\bar{q}}$$

Multiplication. The operation of multiplication is best expressed with the scalar and vector form as follow

$$\begin{aligned} q &= (a, \boldsymbol{v}) \\ p &= (x, \boldsymbol{u}) \\ qp &= (ax - \boldsymbol{v}.\boldsymbol{u}, a\boldsymbol{u} + x\boldsymbol{v} + \boldsymbol{v} \times \boldsymbol{u}) \\ &= (ax - by - cz - dw, ay + bx + cw - dz, az - bw + cx + dy, aw + bz - cy + dx). \end{aligned}$$

where $\bullet$ denotes the inner product and $\times$ is the outer product. From the above, we can easily see that quaternion multiplication is not commutative since outer products are not commutative.

Inverse. From previous discussion on conjugates, we have

$$q\bar{q} = \bar{q}q = |q|^2$$

which on dividing gives

$$\frac{q\bar{q}}{|q|^2} = \frac{\bar{q}q}{|q|^2} = 1$$

So we have the inverse of q as $q^{-1} = \bar{q}/|q|^2$. Division can thus be defined by the use of the inverse.

4 Rotations

In geometric modeling of proteins, we frequently need to compute the rotation of an arbitrary vector $\boldsymbol{P}$ through an angle θ about an arbitrary axis whose direction is represented by a *unit* vector $\boldsymbol{A}$. It is possible to decompose $\boldsymbol{P}$ into components that are parallel to $\boldsymbol{A}$ and perpendicular to $\boldsymbol{A}$. Since the rotation only acts on the perpendicular components, the problem is reduced to that of rotating the perpendicular components about the vector $\boldsymbol{A}$.

The parallel component is found by projecting the vector $\boldsymbol{P}$ on to $\boldsymbol{A}$ as follow

$$Proj_a(\boldsymbol{P}) = (\boldsymbol{A}.\boldsymbol{P})\boldsymbol{A}$$

The perpendicular component is then given by

$$Perp_a(\boldsymbol{P}) = \boldsymbol{P} - (\boldsymbol{A}.\boldsymbol{P})\boldsymbol{A}$$

After rotating the perpendicular component, we just need add back the parallel component to get the final answer.

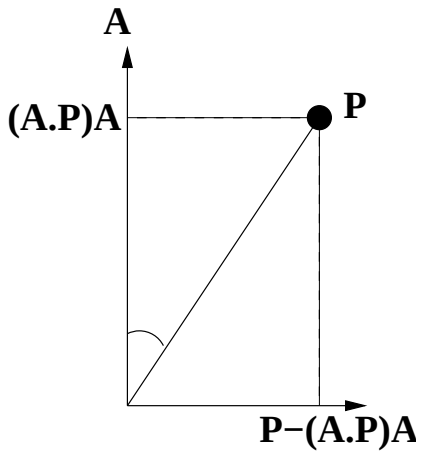

Figure 18: Rotation about an arbitrary axis.

The rotation of the perpendicular component is about a plane perpendicular to the vector $\boldsymbol{A}$. The final rotated version of $Perp_a(\boldsymbol{P})$ about $\boldsymbol{A}$ can be expressed as a linear combination of $Perp_a(\boldsymbol{P})$ and a vector that results from the 90-degree counterclockwise rotation of $Perp_a(\boldsymbol{P})$ about $\boldsymbol{A}$. Let α be the angle $\boldsymbol{P}$ makes with $\boldsymbol{A}$. We note that the length of $Perp_a(\boldsymbol{P})$ is equal to $|P| \sin \alpha$. A vector of the same length and in the direction we need is given by $\boldsymbol{A} \times \boldsymbol{P}$.

The rotation of $Perp_a(\boldsymbol{P})$ about the axis $\boldsymbol{A}$ through an angle of θ can be written as

$$[\boldsymbol{P} - (\boldsymbol{A}.\boldsymbol{P})\boldsymbol{A}] \cos\theta + (\boldsymbol{A} \times \boldsymbol{P}) \sin\theta$$

on adding the parallel component, we have the final rotated version of $\boldsymbol{P}$ about the axis $\boldsymbol{A}$ as

$$\begin{aligned} \boldsymbol{P'} &= (\boldsymbol{A}.\boldsymbol{P})\boldsymbol{A} + [\boldsymbol{P} - (\boldsymbol{A}.\boldsymbol{P})\boldsymbol{A}] \cos\theta + (\boldsymbol{A} \times \boldsymbol{P}) \sin\theta \\ &= \boldsymbol{P} \cos\theta + (\boldsymbol{A} \times \boldsymbol{P}) \sin\theta + (1 - \cos\theta)(\boldsymbol{A}.\boldsymbol{P})\boldsymbol{A} \end{aligned} \tag{1}$$

4.1 Rotations using Quaternions

The main problem with encoding a rotation in a 3×3 matrix is the inevitable numerical drift encountered by using finite precision arithmetic. A 3×3 rotation matrix is orthonormal, and its 9 components encode the 3 degrees of freedom of the rotation with orthonormality providing the 6 remaining constraints. As numerical computations on the matrix build up the components naturally drift, violating the orthonormality constraints. This means that the matrix is no longer a valid rotation and abnormalities in the transform will start to appear.

One solution to this problem is to regularly check that the matrix is orthonormal, correcting it if necessary. Although quite simple to achieve this is computationally

expensive, and the act of correcting the matrix may alter the encoded rotation in unexpected ways. Due to the long chain structure of proteins, even minute errors in the tranformation matrix can cause one to end up with a 3D structure that is totally off its actual shape.

It turns out that rotations can be encoded using quaternions. Since quaternions have 4 degrees of freedom there is only 1 redundant constraint, which is far easier to enforce than matrix orthogonality. A wise choice of encoding also allows us to use quaternion algebra as described above to easily manipulate our rotations in quaternion form.

4.2 Deriving Quaternion Rotation

Rotations in three dimensions can be thought of as a function φ that maps $\mathbb{R}^3$ onto itself with the properties of length, angle and handedness preservation.

Length Preservation. A function φ is length preserving iff

$$|\varphi(\boldsymbol{P})| = |\boldsymbol{P}| \tag{2}$$

Angle Preservation. The preservation of angles can be expressed using the concept of dot products. If $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$ are two points in $\mathbb{R}^3$, an angle preserving function φ must satisfy

$$\varphi(\boldsymbol{P}_1).\varphi(\boldsymbol{P}_2) = \boldsymbol{P}_1.\boldsymbol{P}_2 \tag{3}$$

Handedness Preserving. This means that orientations are preserved under the action of the function φ. This is mathematically expressed via the notion of cross products

$$\varphi(\boldsymbol{P}_1) \times \varphi(\boldsymbol{P}_2) = \varphi(\boldsymbol{P}_1 \times \boldsymbol{P}_2) \tag{4}$$

The next step is to generalize the above concepts to $\mathbb{H}$. If we impose the condition $\varphi(a + \boldsymbol{v}) = a + \varphi(\boldsymbol{v})$ and treat the points $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$ as quaternions with zero scalar part, then (3) can be written as

$$\varphi(\boldsymbol{P}_1).\varphi(\boldsymbol{P}_2) = \varphi(\boldsymbol{P}_1.\boldsymbol{P}_2) \tag{5}$$

Since $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$ have zero scalar part, multiplying them in quaternion fashion gives

$$\boldsymbol{P}_1\boldsymbol{P}_2 = -\boldsymbol{P}_1.\boldsymbol{P}_2 + \boldsymbol{P}_1 \times \boldsymbol{P}_2. \tag{6}$$

With this we can combine the angle and handedness preservation properties into a single equation as follow

$$\varphi(\boldsymbol{P}_1)\varphi(\boldsymbol{P}_2) = \varphi(\boldsymbol{P}_1\boldsymbol{P}_2). \tag{7}$$

We call any function satisfying the above a *homomorphism*. In order to represent rotations, we need to find a class of functions that satisfy (7) and (2). It can be shown that the class of functions defined by

$$\varphi_q(\boldsymbol{P}) = \boldsymbol{q}\boldsymbol{P}\boldsymbol{q}^{-1} \qquad \boldsymbol{q} \in \mathbb{H}, \boldsymbol{q} \neq \bar{0} \tag{8}$$

are the required functions that we are looking for to represent rotations using quaternions. The length preservation property can be verified as follow

$$|\varphi_q(\boldsymbol{P})| = |\boldsymbol{q}\boldsymbol{P}\boldsymbol{q}^{-1}| = |\boldsymbol{q}||\boldsymbol{P}||\boldsymbol{q}^{-1}| = |\boldsymbol{P}|\frac{|\boldsymbol{q}||\bar{\boldsymbol{q}}|}{|\boldsymbol{q}^2|} = |\boldsymbol{P}|.$$

Similarly, homomorphism is demonstrated as below

$$\varphi_q(\boldsymbol{P}_1)\varphi_q(\boldsymbol{P}_2) = \boldsymbol{q}\boldsymbol{P}_1\boldsymbol{q}^{-1}\boldsymbol{q}\boldsymbol{P}_2\boldsymbol{q}^{-1} = \boldsymbol{q}\boldsymbol{P}_1\boldsymbol{P}_2\boldsymbol{q}^{-1} = \varphi_q(\boldsymbol{P}_1\boldsymbol{P}_2)$$

To take stock of things, what we have at this point is a means of capturing rotations using quaternions. What we need to figure out now is how to represent the familiar operation of rotation about an arbitrary axis using such a representation. Specifically, this means figuring out what are the values for the components of $\boldsymbol{q}$ such that it achieves a rotation of θ about some axis $\boldsymbol{A}$. It turns out that we only need to concern ourselves with *unit* quaternions when representing rotations, since length scaling do not affect the outcome.

Let $\boldsymbol{q} = s + \boldsymbol{v}$ be a unit quaternion, we examine equation (8),

$$\begin{aligned}
\boldsymbol{q}\boldsymbol{P}\boldsymbol{q}^{-1} &= (s+\boldsymbol{v})\boldsymbol{P}(s-\boldsymbol{v}) \\
&= (-\boldsymbol{v}.\boldsymbol{P} + s\boldsymbol{P} + \boldsymbol{v}\times\boldsymbol{P})(s-\boldsymbol{v}) \\
&= -s\boldsymbol{v}.\boldsymbol{P} + s^2\boldsymbol{P} + s\boldsymbol{v}\times\boldsymbol{P} + (\boldsymbol{v}.\boldsymbol{P})\boldsymbol{v} - s\boldsymbol{P}\boldsymbol{v} - (\boldsymbol{v}\times\boldsymbol{P})\boldsymbol{v} \\
&= s^2\boldsymbol{P} + 2s\boldsymbol{v}\times\boldsymbol{P} + (\boldsymbol{v}.\boldsymbol{P})\boldsymbol{v} - \boldsymbol{v}\times\boldsymbol{P}\times\boldsymbol{v}
\end{aligned}$$

From the properties of cross products, we have

$$\boldsymbol{P}\times\boldsymbol{Q}\times\boldsymbol{P} = P^2\boldsymbol{Q} - (\boldsymbol{P}.\boldsymbol{Q})\boldsymbol{P}$$

Substituting into the previous equation we obtain

$$\begin{aligned}
\boldsymbol{q}\boldsymbol{P}\boldsymbol{q}^{-1} &= s^2\boldsymbol{P} + 2s\boldsymbol{v}\times\boldsymbol{P} + (\boldsymbol{v}.\boldsymbol{P})\boldsymbol{v} - (\boldsymbol{v}^2\boldsymbol{P} - (\boldsymbol{v}.\boldsymbol{P})\boldsymbol{v}) \\
&= (s^2 - \boldsymbol{v}^2)\boldsymbol{P} + 2s\boldsymbol{v}\times\boldsymbol{P} + 2(\boldsymbol{v}.\boldsymbol{P})\boldsymbol{v}.
\end{aligned} \tag{9}$$

If we set $\boldsymbol{v} = t\boldsymbol{A}$ where $\boldsymbol{A}$ is a unit vector then (10) becomes

$$\boldsymbol{q}\boldsymbol{P}\boldsymbol{q}^{-1} = (s^2 - t^2)\boldsymbol{P} + 2st\boldsymbol{A}\times\boldsymbol{P} + 2t^2(\boldsymbol{A}.\boldsymbol{P})\boldsymbol{A}. \tag{10}$$

The above equation is similar in form to that in (1). If we compare the equations and equate the coefficients of the respective vector terms, we obtain the following

$$\begin{aligned} s^2 - t^2 &= \cos\theta \\ 2st &= \sin\theta \\ 2t^2 &= 1 - \cos\theta \end{aligned}$$

From the third eqaution we have

$$t = \sqrt{\frac{1-\cos\theta}{2}} = \sin\frac{\theta}{2}.$$

The first and third equation implies that $s^2 + t^2 = 1$, therefore s has to be $\cos(\theta/2)$. It is trivial to check that the second equation is also satisfied by these choice of s and t.

We have now fully determined the exact component values required to encode a rotation through the angle θ about the axis $\mathbf{A}$. It is represented by the quaternion

$$\mathbf{q} = \cos\frac{\theta}{2} + \mathbf{A}\sin\frac{\theta}{2} \tag{11}$$

It should be noted that any scalar multiple of the quaternion $\mathbf{q}$(in particular $-\mathbf{q}$) also represents the same rotation. This is also the reason why we are only concerned with unit quaternions. The verification of this property is show below

$$(a\mathbf{q})\mathbf{P}(a\mathbf{q}^{-1}) = a\mathbf{q}\mathbf{P}\frac{\mathbf{q}^{-1}}{a} = \mathbf{q}\mathbf{P}\mathbf{q}^{-1} \tag{12}$$

Concatenation of Rotations. Given two quaternions $\mathbf{q}_1$ and $\mathbf{q}_2$, the product $\mathbf{q}_1\mathbf{q}_2$ represents the rotation resulting from first applying $\mathbf{q}_2$ followed by $\mathbf{q}_1$. This can be verified as below

$$\mathbf{q}_1(\mathbf{q}_2\mathbf{P}\mathbf{q}_2^{-1})\mathbf{q}_1^{-1} = (\mathbf{q}_1\mathbf{q}_2)\mathbf{P}(\mathbf{q}_1\mathbf{q}_2)^{-1} \tag{13}$$

Thus, it is possible to concatenate a whole strings of rotation operations into a single quaternion representing the entire series of rotations.

Representing rotations using quaternions instead of the usual matrix approach provides both better efficiency in computation as well as making it more resilient to error accumulation during numerical computation. The number of multiplication and add operations required to multiply two quaternions is 16 whereas for matrix operations the figure is 27. Since quaternions have fewer parameters, it is more robust to errors when concatenating long chains of rotations. Hence quaternions is clearly better in computing rigid transformation of proteins.

References

[1] Furio Ercolessi, *A Molecular Dynamics Primer*, Spring College in Computational Physics, ICTP, Trieste June 1997

[2] Dan Halperin, Lydia Kavraki, Jean-Claude Latombe, Rajeev Matwani, Christian Shelton and Suresh Venkatasubramanian, *"Geometric Manipulation of Flexible Ligands"*, Proc. of 1996 ACM Workshop on Applied Computational Geometry, May 1996

[3] Dan Halperin and M.H. Overmars, *"Spheres, Molecules and Hidden Surface Removal"*, Proc 10*th* ACM Symposium on Computational Geometry, Stony Brook, 1994, Pg 113-122

[4] Dan Halperin and Christian R. Shelton, *"A Perturbation Scheme for Spherical Arrangements with Application to Molecular Modeling"*, Annu. Symp. on Comp. Geom. 1997, Pg 183-192

[5] T C Lim, *"The Relationship between Lennard-Jones(12-6) and Morse Potential Functions"*, Zeitschrift fur Naturforschung A, Vol.58, no.11, pp.615-617 (2003)

[6] Michael Levitt, Mark Gerstein, Enoch Huang, S. Subbiah and Jerry Tsai, *"Protein Folding: The Endgame"*, Annu. Rev. Biochem., 66, Pg 549-579, 1997

[7] B. Lee and F. M. Richards, *"The Interpretation of Protein Structures: Estimation of Static Accessibility"*, J. Mol. Biol. 55, Pg 379-400

[8] B. Maher, *"Researchers reveal a new twist in torsion dystonia"*, The Scientist, 17(8), Pg 32-33, April 21, 2003

[9] P.G.Mezey, Molecular Surfaces, Reviews in Computational Chemistry, Vol I, K.B.Lipkowitz and D.B.Boyd, Eds., VCH Publishers, 1990, Pg 265-294