Algorithms for Molecular Biology **Fall Semester, 2001**

Lecture 12: January 24, 2002

Lecturer: Haim Wolfson *Scribe: Gilad Wainreb, Emir Haleva*[1]

12.1 Protein Structure Introduction

12.1.1 Background

Proteins are long chains of Amino Acids (AA). There are 20 different AAs that serve as building blocks for proteins. Each AA has a specific chemical structure which contains a carbon backbone similar to all amino acids and a residue which varies between the AAs. The length of a protein chain can range from 50 to 1000-3000 AA (200 on the average). Proteins are known to have many important functions in the cell, such as enzymatic activity, storage and transport of material, signal transduction, antibodies and more. An important property of a protein is the length and composition of the AA chain. The series can be obtained automatically from the gene that encodes for the protein. Another interesting property is the unique folding. The AA composition of a protein will usually uniquely determine (on specific environment conditions) the 3D structure of the protein (e.g., two proteins with the same AA sequence will have the same 3D structure in natural conditions). An experiment conducted by Anfinsen [1] showed that a denaturated protein (unfolded by special chemicals), folded back to its original structure after the removal of the denaturating chemicals. All proteins whose structure is known are stored in the Protein DataBank (PDB) which contains about 20,000 proteins [7].

A protein has multiple levels of structure (see Figure 12.1):

- Primary structure - Chain of Amino Acids (1 dimensional).
- Secondary structure - Chains of structural regular elements, most important of which are α-helices and β-sheets.
- Tertiary and Quaternary structure - 3D structure, of a single AA chain or several chains, respectively.

The common methods for finding protein 3D structure are:

- Cristalography - Performed by X-ray diffraction and neutron-diffraction.

[1] based on a scribe from January 22, 2001 written by Dina Duhovny, Aviad Tsherniak.

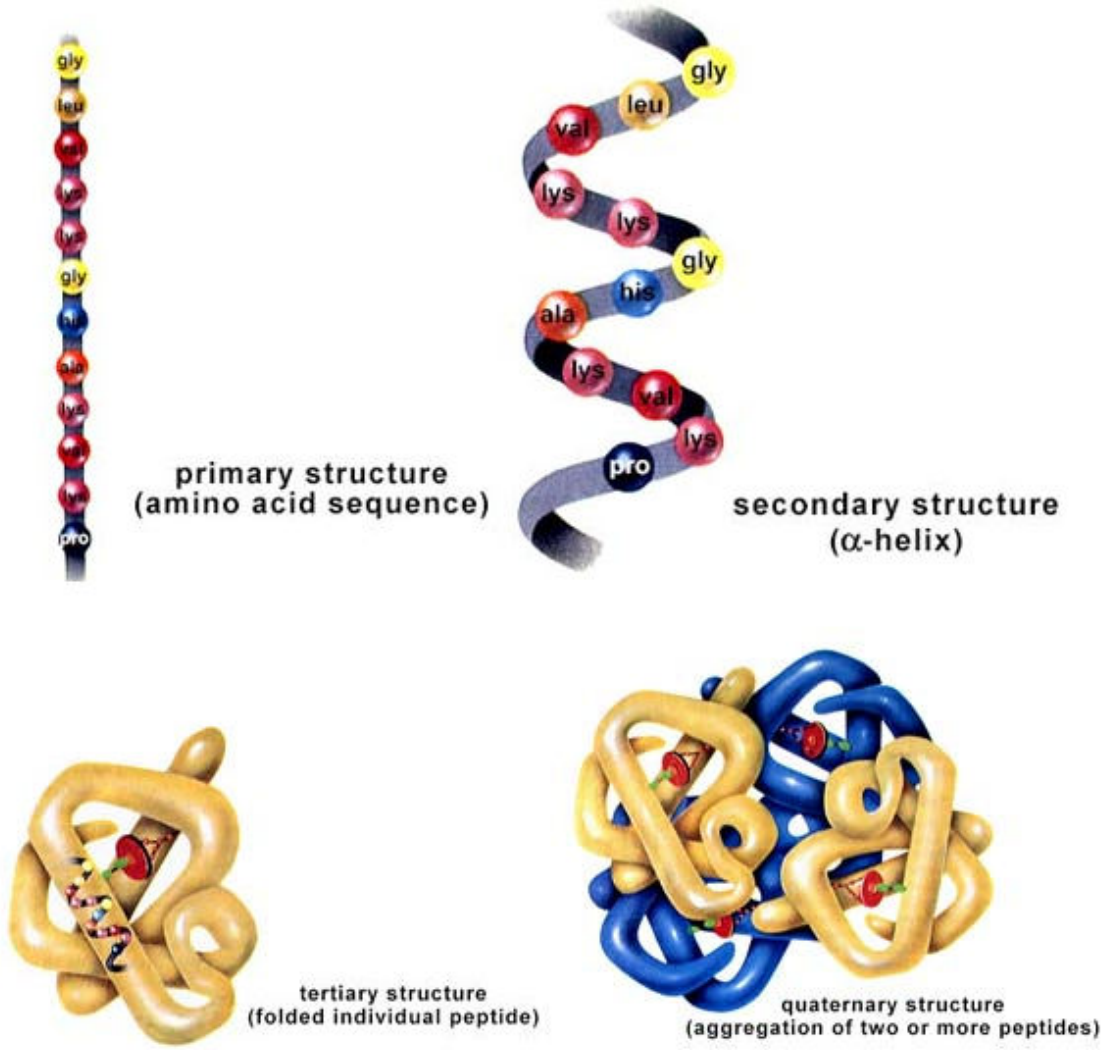

Figure 12.1: Source: [15]. The four structuring levels of a protein.

- Nuclear Magnetic Resonance.

These methods are slow and costly (taking up to several months of lab work), much slower than DNA sequencing. This creates interest in algorithms for protein structure prediction.

12.1.2 Motivation for Protein 3D Structure Prediction

The structure of the protein is directly related to the protein's functionality. The reasons for research of 3D
structure are:

- Medicine - Understanding biological functions. Binding and unbinding of proteins constitute much of the cellular activity of living organisms.
- Finding "targets" for docking drugs.
- Agriculture - *Genetic engineering* of better and richer crops.
- Industry - Synthesis of enzymes (e.g. detergents).

The main reasons for using 3D comparison algorithms (rather than just the AA sequence) are:

- Protein 3D structure is more highly conserved through evolution than the primary structure.
- The 3D structure encapsulate more information than just the AA sequence (e.g. active sites)

12.1.3 Protein 3D Structure

The main hypothesis is that a protein folds to one unique structure, which depends only on the AAs sequence.
The physical explanation for this phenomenon is that proteins fold in order to reach the minimal level of energy. Different AA have different chemical, electrical, and size properties, and therefore two different folds of a protein usually have two different levels of energy.

Definition The *Van der Waals radius* of an atom is defined as the minimum radius of the nucleus into which other atoms can not "penetrate" (two balls with Van der Waals radius cannot overlap).

We will use Van der Waals radius balls as a 3D model of an atom. Each AA has a carbon atom called C_α, connected to a carboxyl group and an amine group, a hydrogen atom and a part that depends on the specific AA - the *residue*. In the protein chain Amine group of one AA connects to the carboxyl group of the next adjacent AA (see Figure12.2). The C_α-s form together a backbone wire, to which the rest of the atoms are attached.

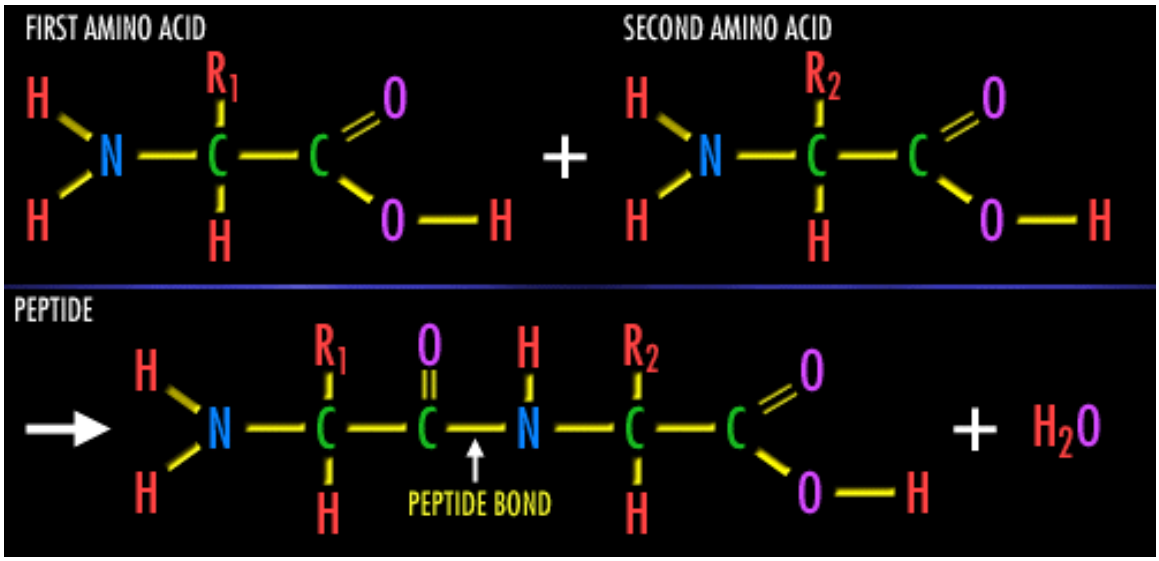

Figure 12.2: Source: [16]. Formation of a peptide bond between two amino acids by the condensation (dehydration) of the amino end of one amino acid and the acid end of the other amino acid.

12.1.4 Methods for Protein Folding

We will regard similar protein sequences as having similar 3D structure. However, recent studies on prions (certain proteins that have folded differently than their family) have shown that is not always true. Therefore the method of choice for folding a given protein depends on existence of a similar protein whose structure is already known, and on the extent of such similarity.

- When one can find a known-structure protein with good sequence similarity (over 30% amino-acid identity) to the protein we wish to fold, the two proteins will have the same structure. This method is called *homology modeling*.
- When only less conclusive similarity is available to a known structure, we can use *threading* as follows. Align our protein to a remotely similar protein whose structure is known. Use the new forced structure as a starting point for finer folding operations.

- When no homology is available one is forced to fold the proteins *ab-initio* (from scratch). This is hard even in simplistic models. For example we can simplify the problem by dividing the AAs into two kinds: hydrophobic (water hater) and hydrophilic (water lover). Using this simple model we can try to build the 3D structure minimizing rejections and maximizing attractions between nearby amino acids. Even solving this simple model was proven to be NPC [4], although there are heuristics for it (the best known has $\frac{2}{3}$ approximation ratio).

12.1.5 Structural Genomics Project

This project aims at finding all protein structures using Homology Modeling approach. The number of protein sequences known so far is much higher than the number of proteins with known 3D structure. The number of known protein folds is relatively small (there are about 600-700 different folds among the 20,000 PDB structures). Proteins within the same family usually have the same fold. Under these assumptions the Structural Genomics Project works as follows. The space of protein sequences is divided to clusters according to sequence similarity. If there is a protein with known 3D structure in the cluster, then the other structures are determined using homology modeling. If all the structures are unknown, the most "appropriate" protein for crystallization is selected from the cluster and its structure is determined by X-ray crystallography.

12.2 Protein Structural Alignment - The Rigid Case

12.2.1 Protein Shape Representation

When we want to align proteins, we have to think of a discrete 3D shape representation of the molecules. Possible "critical features" may be:

- Backbone C_α atomic centers (see Figure [12.3]).
- C_α–C_β vectors.
- Secondary Structure Elements (α-helices, β-sheets).
- Molecular Surface Representation (Figure 12.10).

12.2.2 Problem Definition

The input to protein pairwise structural comparison is a set of 3D atomic coordinates of two different molecules(see Figure [12.3]).

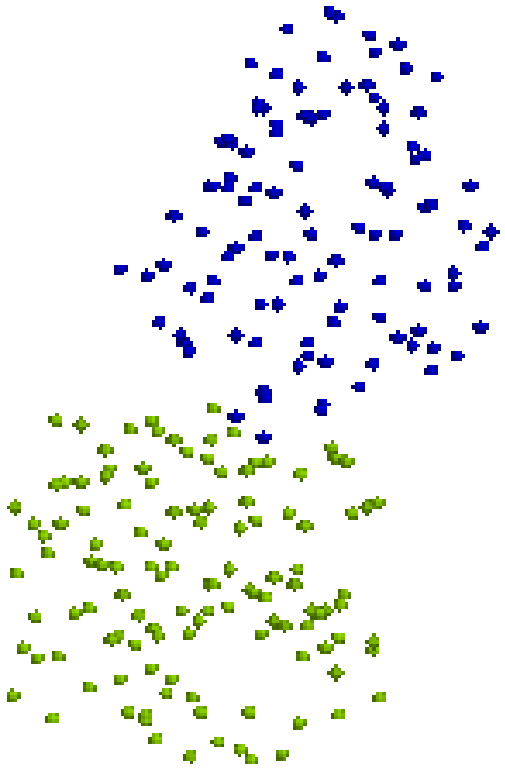

Figure 12.3: C_α coordinates of input molecules.

The goal is to find a rigid transformation (rotation and translation) in space that matches a "sufficient" number of atoms of one molecule to those of the other molecule. The algorithms that tackle the problem can be divided as follows:

Sequence order dependent - use the order of atoms on the protein chain, thus reducing a problem to 3D curve matching, that is essentially a one dimensional task.

Advantages of the Sequence Dependent Approach

Sometimes biologists are more interested in motifs preserving sequence order, since mutations might happen in specific area. In addition, the computational task becomes easier when an order of chain is exploited. these algorithms are:

- Taylor and Orengo: following the sequence of amino acid try to build the structure of the protein. For each residue define a local, rotation and translation invariant structural environment. For each pair of residues compute their similarity/distance based on their structural environments. Use the above computed distances as entries of a dynamic programming matrix. Find optimal path in the matrix.
- Vriend and Sander: cut the sequence into chunks of 10-20AA and try to find a matching pattern for each chunk.
- Shetsky et al.(will be discussed late).

Sequence order independent - align features in 3D space (e.g., aligning the two proteins from Figure 12.3), a real 3D task.

Advantages of the Sequence Independent Alignment

Since the techniques do not exploit chain order, they can detect non-sequential motifs in proteins, such as molecular surface motifs, especially binding sites. In addition, it allows us to search structural databases with only partial structural information. The same algorithms can be applied to other molecular structures, such as drugs. It also provides robustness to insertions and deletions.

Analogy with Object Recognition in Computer Vision

The problems raised in the research of the proteins 3D structure have a surprising similarity to problems in computer vision.
The task of model-based recognition in computer vision is given a model database to identify

and locate in the target image (see Figure [12.4]) all the instances of models. The models appearing in the image are usually transformed by an *a-priori* unknown transformation, and can also be partially occluded. The solution is shown in Figure [12.5].

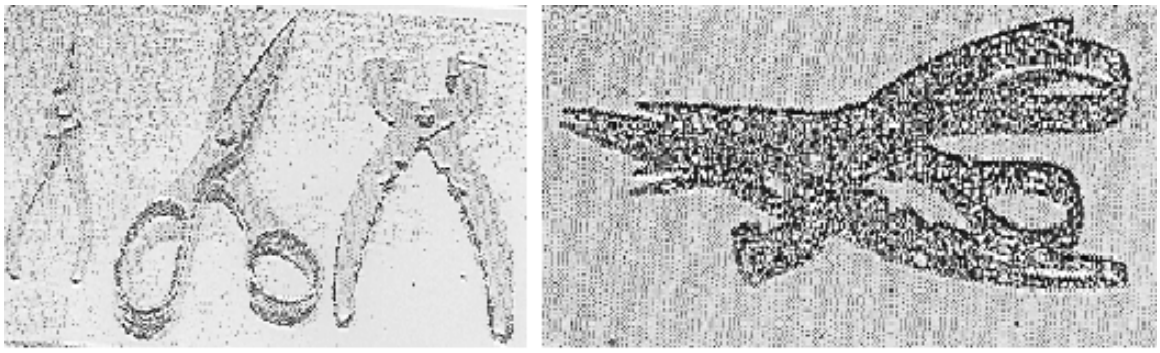

Figure 12.4: model and target images from [9].

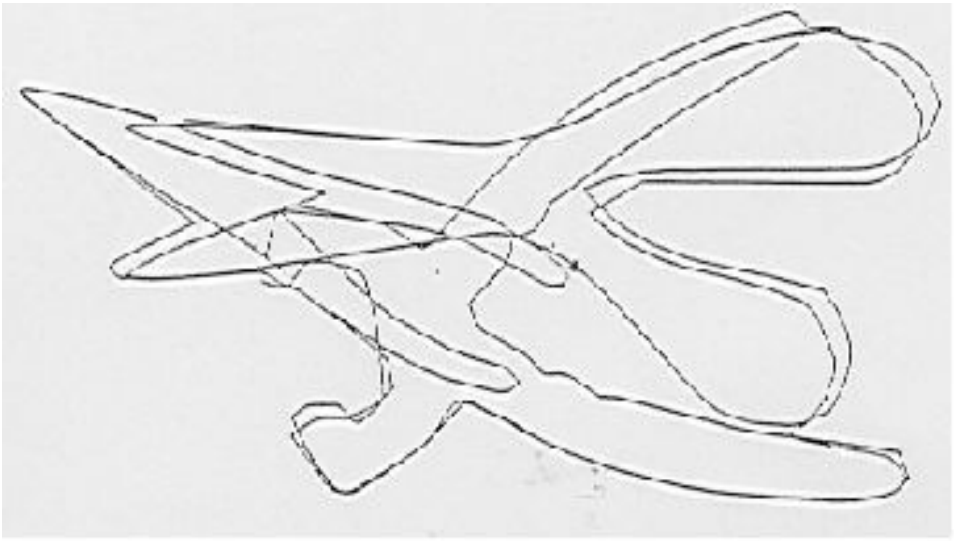

Figure 12.5: The solution to the object recognition problem in Figure [12.4].

One can divide the problem of structural comparison into two major tasks:

- The ***correspondence*** task - detecting matching features (difficult).
- The ***best superposition*** of given matching features - finding a transformation of one structure into another with minimal RMSD (Root Mean Square Deviation).

Superposition - best least squares (RMSD) rigid alignment

The input: two sets of 3D points: $P = \{p_i\}, Q = \{q_i\}, i = 1...n$.
The goal: find a 3D rotation R_0 (i.e., turns a figure about a point a given R_0 degrees) and translation a_0 (i.e., each of the points of the geometric figure moves a_0 distance in the same direction) such that: $min_{R,a} \sum_i |Rp_i + a - q_i|^2 = \sum_i |R_0 p_i + a_0 - q_i|^2$
The solution: A closed form solution exists for this task. It can be computed in $O(n)$ time ([13]).
The problem is related to the well known Procrustees problems in statistics and involves eigenvalue analysis of a correlation matrix of the points.

Solution of the Correspondence (Matching) Problem

We exploit the fact that our objects are rigid. In this case the correspondence of a pair of ordered triplets of points (*"fat enough"* triangles), uniquely defines a 3D rigid transformation (see Figure [12.6]).

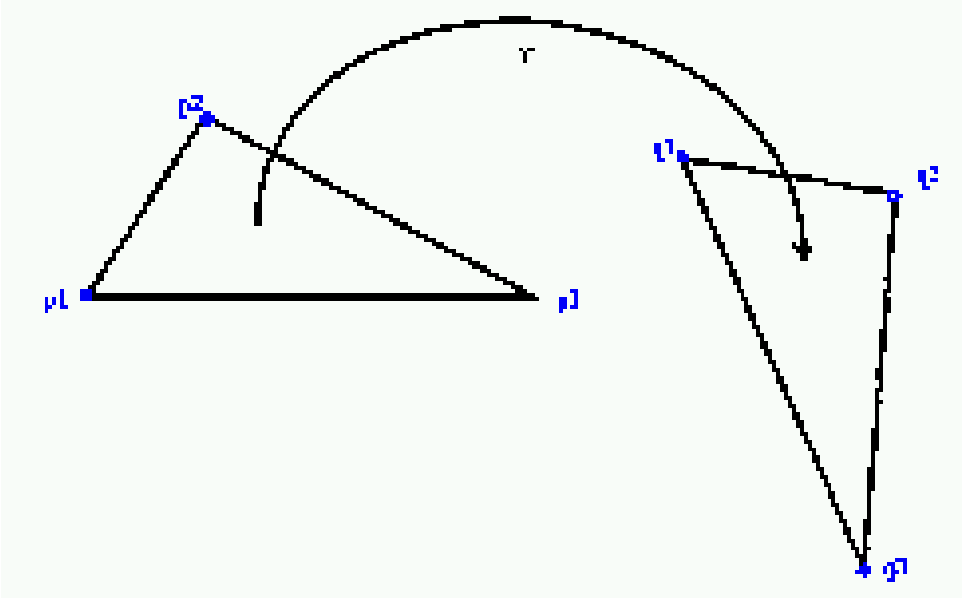

Figure 12.6: Transformation of one triangle into another.

This leads to the following algorithm:

- For each pair of triplets, one from each molecule that define 'almost' congruent triangles, compute the transformation that superimposes them.
- Count the number of point pairs, which are 'almost' superimposed and score the hypotheses by this number.

- Pick the highest ranking hypotheses and improve the transformation by replacing it with the best RMSD transformation for all matching pairs.

Complexity: assuming $O(n)$ points in both molecules - $O(n^7)$.

12.2.3 Geometric Hashing technique based Alignment Algorithm

This approach was originally developed for object recognition in Computer Vision [8, 9]. The algorithm has two phases: preprocessing, and recognition. In the first phase, each of the models in the database is processed. In this phase, for each model, its geometric information is encoded according to a hash table. This can be done offline. In the second phase, given an object in a scene, its features are extracted. These features are used to map the object to multiple entries in the hash table.

Definition: A 3D ***reference frame*** is a triplet of orthogonal vectors emanating from a common arbitrary origin. It can be uniquely defined by the ordered vertices of a non-degenerate triangle.

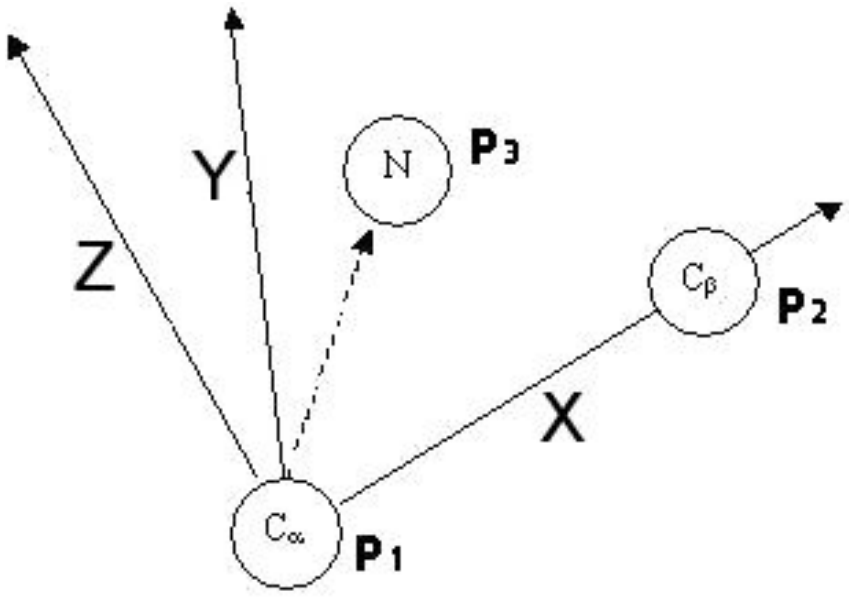

Figure 12.7: Example of possible Reference Frame choice

There are many ways to define it. For example we can select the origin of the reference frame to be p_1. The x-axis will be in the direction of a vector $p_2 - p_1$. The y-axis is on

the triangle plane orthogonal to x-axis in the counterclockwise direction and the z-axis is orthogonal to the triangle plane, its direction defined by right hand rule. Since we are dealing with rigid objects, we can pick unit length vectors (see Figure [12.7]).
Suppose e_x, e_y, e_z are the relevant unit vectors, than each point v in the 3D space can be represented using the above reference frame as $v = \alpha e_x + \beta e_y + \gamma e_z + p_1$. The lengths of the triangle sides are translation/rotation *invariant*, and therefore define a valid ***shape signature*** of the reference frame.

The algorithm

The preprocess stage encodes the geometric information to a hash table. For each *model* object (first molecule in our case or each molecule in a database) do:

1. Pick a reference frame.
2. Compute the 3D orthonormal basis associated with this reference frame and its shape signature (triangle sides length).
3. Compute the coordinates of all the other points (in a pre-specified neighborhood) in this reference frame.
4. Use each coordinate as an address to the hash (look-up) table. Store the entry [protein id, ref. frame, shape sign., point] at the hash table address.
5. Repeat the above steps for each model reference frame (noncollinear triplet of model points).

The recognition stage of the algorithm uses the hash table, prepared in the preprocessing step. The matching of a *target* object is accomplished as follows:

1. For each reference frame of the target:
   (a) Compute the 3D orthonormal basis and the shape signature associated with it.
   (b) Compute the coordinates of all other points in the current reference frame.
   (c) Use each coordinate to access the hash-table and retrieve all the records [protein id, ref. frame, shape sign.,point].
2. For records with matching shape signature "vote" for the pair [protein, ref. frame].
3. Compute the transformations of the "high scoring" hypotheses. For each hypothesis one can also register the pairs of matching points. This *match list* along with the transformation comprise a *seed match*.

Now we can construct an alignment algorithm that uses geometric hashing as follows: We first define local neighbors of residues. Note that if we use all points for all possible triplets, each atom will have a redundant representation. It will appear in the hash table in all possible reference frames. (In practice, since we are not interested in very closed nor very distant atoms, we pick atoms in an annulus defined by *min* and *max* radii). The geometric hashing technique is applied next using only neighboring points to detect *seed matches* defined by a transformation and a match-list. Many of the matches obtained before represent the same transformation, i.e., different match lists may share the same transformation. We cluster seed matches and merge match-lists that were found. The last step of the algorithm is the *extending* step. The seed matches are extended to contain additional matching pairs and best RMSD transformation is detected. For this purpose a heuristic iterative matching algorithm which minimizes the sum of the distances between the newly matched pairs is applied.

Complexity
N - number of protein molecules in database.
O(n) - number of "features" in each protein structure.
R - number of reference frames, typically $R = n, n^2, n^3$.
s - size of hash-table entry. s can be kept low by not processing "fat" entries.

Preprocessing: $O(N * R * n)$.

Recognition: $O(R * n * s)$.

12.2.4 The Flexible case

Motivation : Proteins and drugs are flexible entities. There are two kinds of motion within the protein: hinge motion and shear motion [5]. The geometric hashing approach can be extended to take care of flexible case as well as rigid. However this method requires a preliminary knowledge of hinge locations [14]. A successful approach to enable hinges with no prior knowledge was published by Shatsky et al.[12].

The flexprot algorithm[12]

The algorithm is sequence dependent and combines 3-D matching graph theoretic technique and is not sensitive to deletions or insertions. The input to the algorithm are two molecules $A = v_1...v_n$ and $B = w_1...w_m$ which are represented by its Cα coordinates. The goal of the algorithm is to decompose the two molecules into a minimal number of disjoint fragments of maximal size, so that each fragment's number of Cα will be as closely matched as possible to

the number of Cα in the matched fragment and there is a 3-D rotational translation which superimposes the corresponding atoms with a small RMSD.

1. Detect as "big enough" congruent rigid fragments. Assume that we have a single matching atom pair (one from each molecule). Then, iteratively we try to extend this initial match-list by adding one more atom pair to the left and to the right (following the backbone direction) till we obtain the longest pair of consecutive congruent fragments which includes our initial matching atom pair. To calculate all the match-list of continues pairs of matched atoms performed this stage for each pair of atoms V_i, W_j.

2. Find a sequence of disjoint fragments that will follow the sequence of Cα of A and of B. The method for this step is similar to the one used in the FastaA algorithm:

   - The match lists are represented as vertices of a graph.
   - Join two vertices by a directed edge, if the fragment pairs that they represent might be consecutive in the alignment. The fragment pairs are allowed to overlap, the maximal overlap being 2Δ. The result is an acyclic directed graph.
   - Each edge is assigned a weight penalty W(e). Rewarding long matching fragments and penalizing big gaps. Define M1 and M2 as a match list of atoms $V_i...V_j$ with atoms $V_k...V_t$ and $V_b...V_f$ with atoms $V_p...V_r$ respectably, where $0 < i < j < f < n$, $0 < k < t < r < m$, $b < f, p < r$. If there is no overlap then Δ=0. l is defined as the length of the M1. $Gap_1 = j - b$, $Gap_2 = t - p$.
     $W(e) = -((l-1) - \lceil\Delta\rceil)^2 + max(|gap_1|, |gap_2| + ||gap_1| - |gap_2||$.
     (see Figure 12.8).
   - A virtual vertex called a start vertex and edges from it to all other vertex with a zero penalty are added to the graph.
   - The single source shortest path algorithm is preformed, starting from the start vertex. This calculates the shortest path from each vertex.
   - All possible paths are collected and divided according to their number of vertices into groups.
   - The RMSD of the paths is calculated and each group is sorted according to it.
   - Out of each group the ten best results will give a number of the possible solutions each with a different number of hinges.

   The overall complexity is bounded by $O(K)^4$ where $K = max(|A|, |B|)$.

12.3 Molecular Surface Representation

Representing a protein by its molecular surface representation helps us in the study of protein folding [10], in prediction of biomolecular recognition and detection of drug binding 'cavities'.

Because the complexity of these algorithms depends on the number of points representing the molecule surface, a major issue in this representation is the sparseness of the "interest points" that represent the molecule.
A common representation is due to Connolly [3]. It virtually rolls a 'water' probe ball (1.4-1.8 Å diameter) over the Van der Waals surface, smoothing the surface and bridging narrow crevices, which are inaccessible to the solvent. This partitions the surface into convex, concave and saddle patches according to the number of contact points between the surface atoms and the probe ball. As Output, the representation consists of points and normals to the surface. These are sampled according some required sampling density (e.g., 10 pts/Å^2).

There are different ways to represent shape complementarity. The points and normals representation [3] is a dense one. The points representation in [11] is sparser, and so is the Solid Angle local extrema [2]. Another representation is SPHGEN [6] - surface cavity modeling by pseudo-atom centers.

One of the advantages of the molecular surface is its ability to visualize the shape complementarity at interfaces, as show on Figures 12.11 and 12.12.

In the critical points representation based on Connolly's representation we define a single point and a normal for each patch, as illustrated in Figure 12.13. Convex patches are called *caps*, concave ones - *pits* and saddles are called *belts*.

The Solid Angle local extrema (knobs - holes) representation is based upon centering a sphere at the protein surface and measuring the fraction of the sphere inside the solvent-excluded volume of the protein. If more than half of the sphere is inside the protein, the region is concave, if less than half of the sphere is inside the protein, the shape is convex. Either the solid sphere or the sphere surface may be used. A two-dimensional example is shown in Figure 12.14.

The SPHGEN representation generates sets of overlapping spheres to describe the shape of a molecule or molecular surface. For receptors, a negative image of the surface invaginations is created; for a ligand, the program creates a positive image of the entire molecule. Each sphere touches the molecular surface at two points and has its radius along the surface

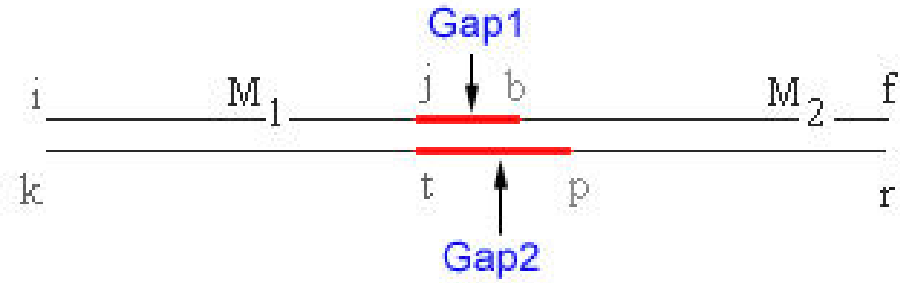

Figure 12.8: An illustration of the indices between two match lists.

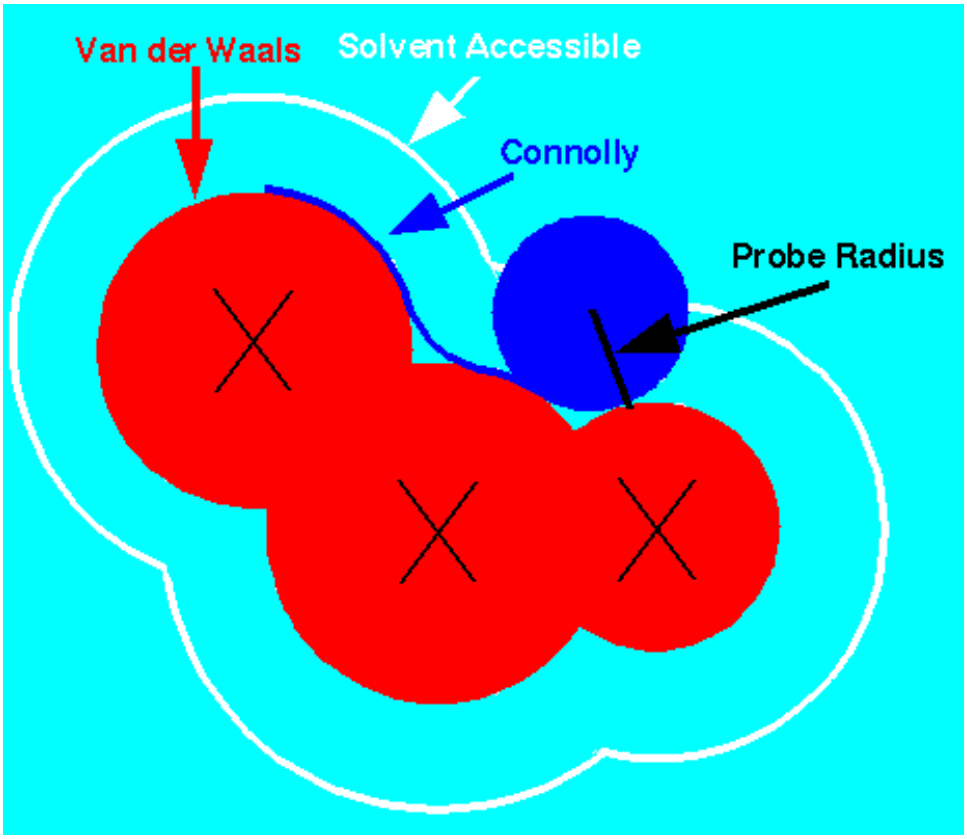

Figure 12.9: Connolly's Solvent Accessible Surface. Image taken from http://www.chem.leeds.ac.uk/ICAMS/eccc/cangaroo.html.

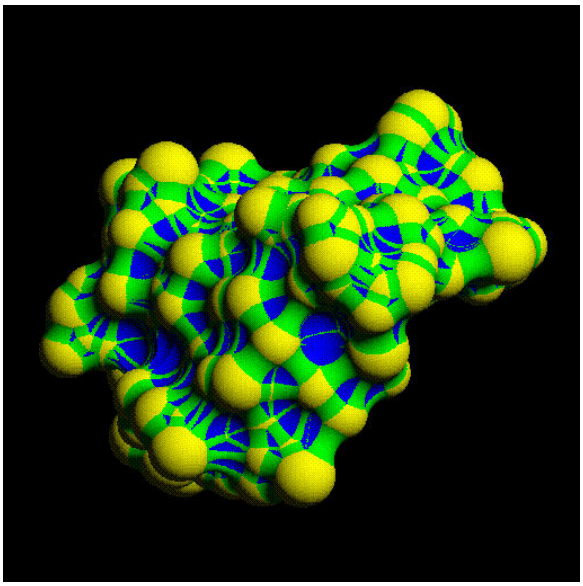

Figure 12.10: The molecular surface of Crambin is shown above, with the convex spherical patches colored yellow, the saddle-shaped pieces of tori colored green, and the concave reentrant surface colored blue. Image taken from http://www.netsci.org/Science/Compchem/feature14f.html.

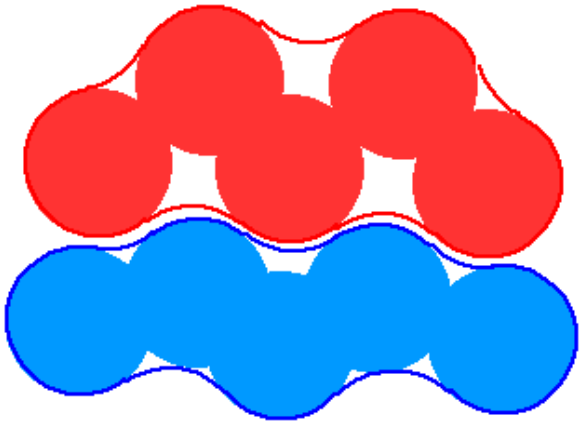

Figure 12.11: Shape complementarity at interfaces. Taken from http://www.netsci.org/Science/Compchem/feature14e.html

normal of one of the points. For the receptor, each sphere center is "outside" the surface, and lies in the direction of a surface normal vector. For a ligand, each sphere center is "inside" the surface, and lies in the direction of a reversed surface normal vector. Spheres are calculated over the entire surface, producing approximately one sphere per surface point. This very dense representation is then filtered to keep only the largest sphere associated with each receptor surface atom. The filtered set is then clustered on the basis of radial overlap between the spheres using a single linkage algorithm. This creates a negative image of the receptor surface, where each invagination is characterized by a set of overlapping spheres. These sets, or "clusters", are sorted according to numbers of constituent spheres, and written out in order of descending size. The largest cluster is typically the ligand binding site of the receptor molecule.

12.4 Docking

Problem 12.1 Docking problem
Input: A receptor organic molecule R and a drug molecule (ligand) L.
Output: A matching between the receptor surface and the ligand surface maximizing the contact area between the surfaces.

There are several reasons for our interest in docking problems:

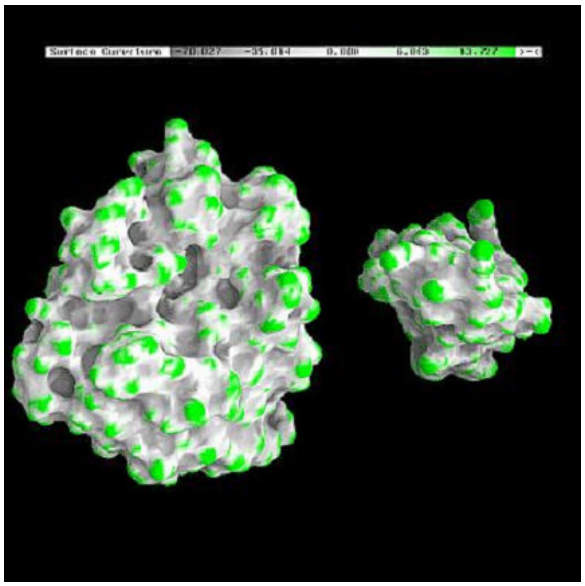

Figure 12.12: Trypsin/Trypsin inhibitor. Figure from B. Honig's Labs web-site at Columbia University.

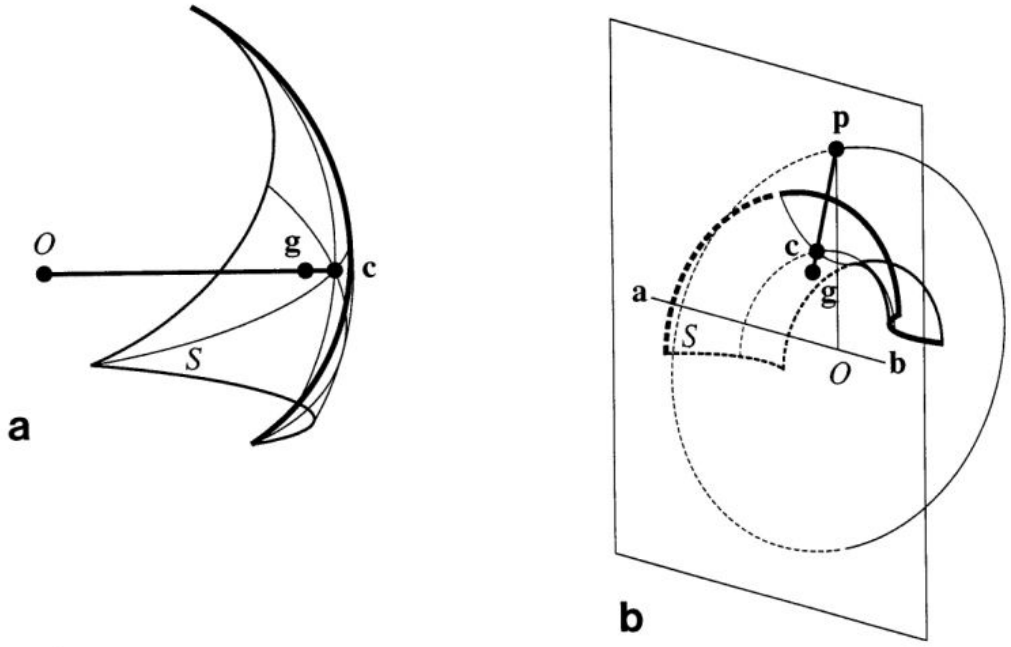

Figure 12.13: An illustration of the generation of (a) *caps* and *pits*, and (b) *belts*, respectively. Symbols are as follows: S is a Connolly face, either convex (for caps) or concave (for pits) (a), and saddle-shaped (for belts) (b); O in (a) is either the atomic center (for caps) or the probe center (for pits). In (b), O is the center of the torus central circle; in (a) and (b), **g** is the gravity center of a face and **c** is its projection on the surface. Also in (b): **a** and **b** are the two atoms along which the torus axis lies, and **p** is where the straight line through **c** and **g** intersects the plain normal to **a**b, that passes through O.

- **Rational drug design** - When we develop a drug that is supposed to be docked on a specific known receptor, we have to adjust it to the receptor. The efficiency of drugs is often a function of the contact area between the ligand (drug molecule) and the receptor.

- **Biomolecular structure recognition** - The action of docking happens naturally when enzymes dock on proteins and react with them. Understanding this process is a part of understanding the reaction processes occurring in organisms.

The main idea of docking is the "key in lock". The ligand is a key - small and sometimes flexible. The receptor is the lock, big and usually with a low level of flexibility. The better these two molecules fit - the better the influence of the drug and the interaction between them, will be. Researches have shown that there are molecules that are not completely rigid, but have partial flexibility. Usually the flexibility is in some spots, called *hinges*, between two parts of the molecule. In the hinges there is usually a determined range of angles where the rigid parts can rotate (see Figure 12.16).

The class of docking problems has two major subclasses.

- The rigid docking problem (two rigid molecules - the simpler problem).

- The flexible docking problem - one (or both) of the molecules has some degrees of freedom. This problem is harder to solve.

When evaluating docking methods, one should examine the following issues:

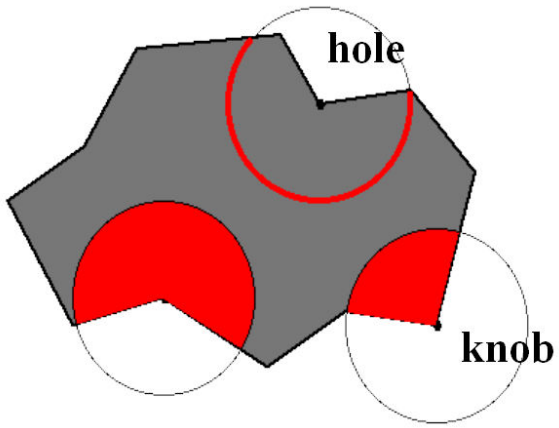

Figure 12.14: Solid Angle local extrema. Taken from http://www.netsci.org/Science/Compchem/feature14h.html

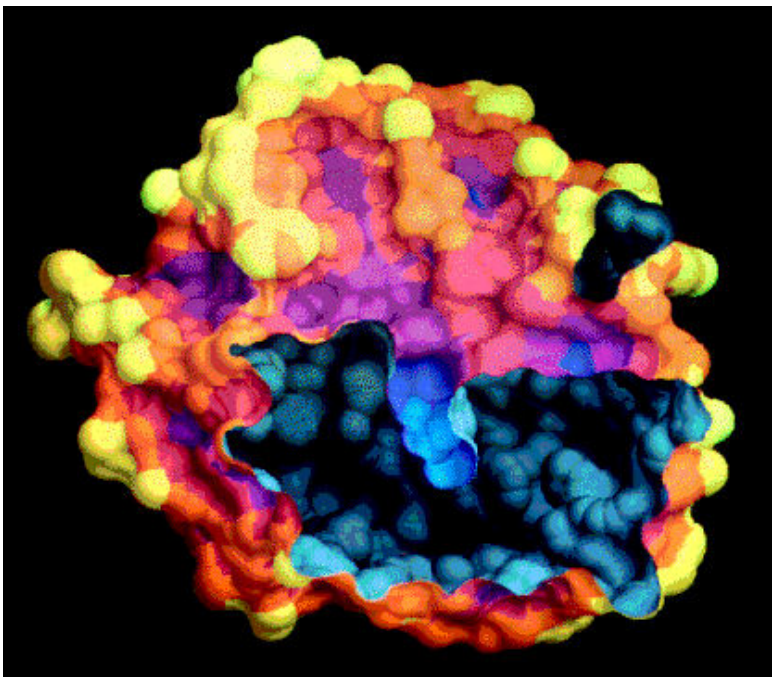

Figure 12.15: The chymotrypsin surface above has been colored according to convexity or concavity. A sphere of radius 6 Å was centered at several points on each surface face, and the solid angle of the sphere lying inside the protein's molecular surface was computed and averaged over the face. Each face was colored according to where its average solid angle fell in the range between zero and four π steradians. Convex regions are yellow, concave regions are blue, and regions of intermediate curvature are orange, red, and purple. The surface has been clipped, and the inner side of the molecular surface has been colored grey. The substrate-binding pocket is at the center of the image and can be seen to be blue. Taken from http://www.netsci.org/Science/Compchem/feature14h.html.

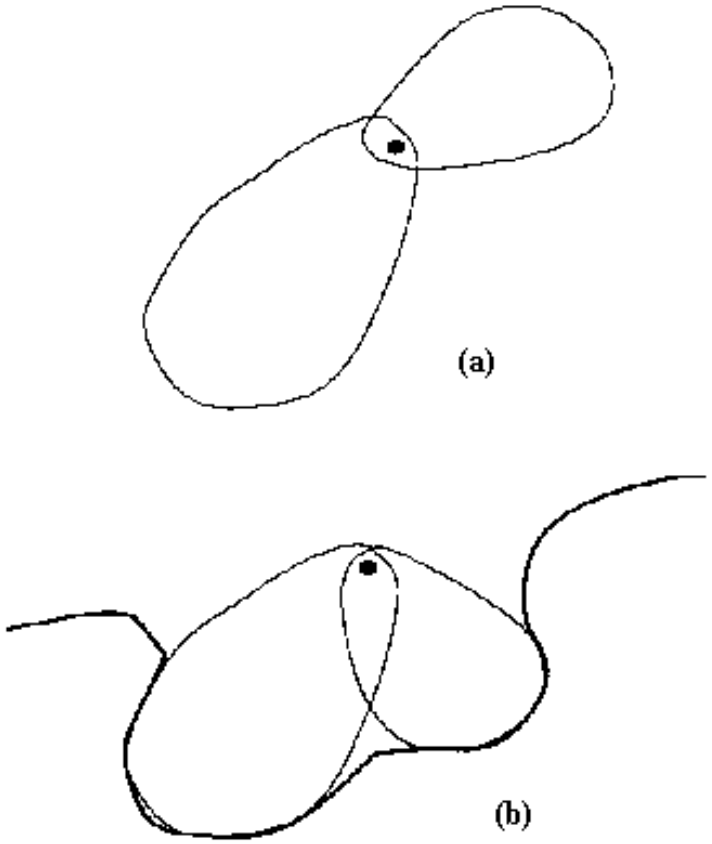

Figure 12.16: A molecule (two rigid parts and one hinge) and a receptor.

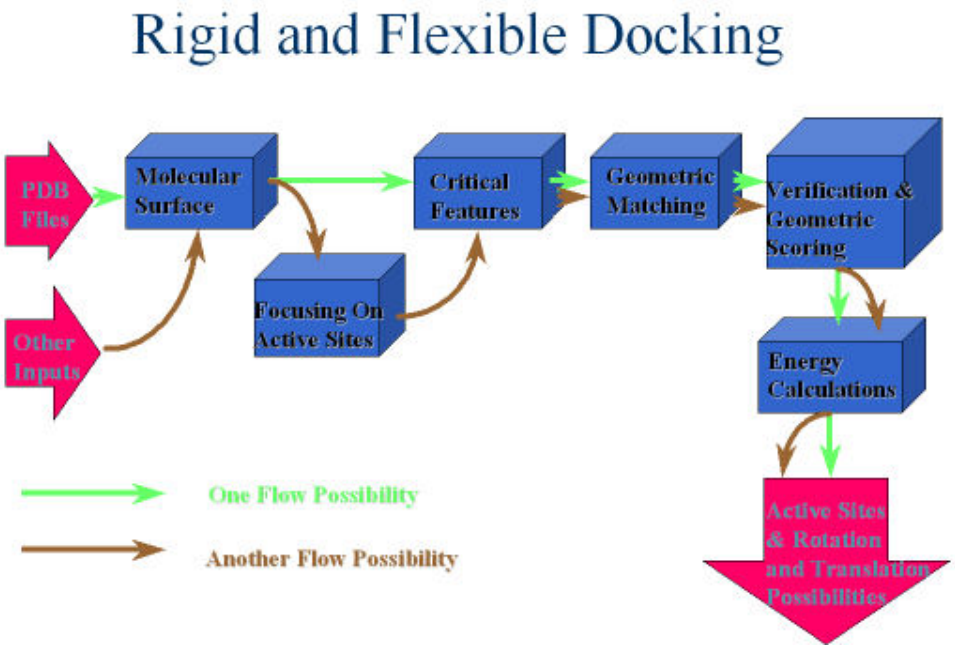

Figure 12.17: Rigid and Flexible Docking.

- Does the method deal with rigid docking or flexible docking?.
  If the method allows flexibility:
  - Is flexibility allowed for ligand only, receptor only or both?.
  - What is the number of flexible bonds allowed and the cost of adding additional flexibility.
- Does the method require prior knowledge of the active site?.
- Speed - ability to explore large libraries.

Figure 12.17 shows a sketch of the different stages a docking method has.

Bibliography

[1] Anfinsen CB. Principles that govern the folding of protein chains. *Science*, 181:223–230, 1973.

[2] M. Connolly. Measurement of protein surface shape by solid angles. *J. Mol. Graph.*, 4:3–6, 1986.

[3] M.L. Connolly. Solvent-accessible surfaces of proteins and nucleic acids. *Science*, 221:709–713, 1983.

[4] D. Papadimitriou C. Piccolboni A. Crescenzi, P. Goldman and M. Yannakakis. On the complexity of protein folding. *J.Computational Biology.*, 5:523–466, 1998.

[5] M. Gerstein, A.M. Lesk, and C. Chothia. Structural Mechanisms for Domain Movements in Proteins. *Biochemistry*, 33(22):6739–6749, 1994.

[6] I. Kuntz, J. Blaney, S. Oatley, R. Langridge, and T. Ferrin. A geometric approach to macromolecule-ligand interactions. *J. Mol. Biol.*, 161:269–288, 1982.

[7] Brookhaven National Laboratory. PDB - protein data bank. *http://www.rcsb.org/pdb/index.html*.

[8] Y. Lamdan, J. T. Schwartz, and H. J. Wolfson. Object Recognition by Affine Invariant Matching. In *Proceedings of the IEEE Computer Vision and Pattern Recognition Conf.*, pages 335–344, Ann Arbor, Michigan, June 1988.

[9] Y. Lamdan, J. T. Schwartz, and H. J. Wolfson. On Recognition of 3-D Objects from 2-D Images. In *Proceedings of IEEE Int. Conf. on Robotics and Automation*, pages 1407–1413, Philadelphia, Pa., April 1988.

[10] B. Lee. The interpretation of protein structures: estimation of static accessibility. *J. Mol. Biol.*, 55:379–400, 1971.

[11] S. L. Lin, R. Nussinov, D. Fischer, and H.J. Wolfson. Molecular Surface Representation by Sparse Critical Points. *PROTEINS: Structure, Function and Genetics*, 18:94–101, 1994.

[12] M. Shatsky, Z.Y. Fligelman, R. Nussinov, and H. Wolfson. Flexprot: an algorithm for alignment of flexible protein structures. *J. Proc. 8th International Conference onIntelligent Systems for Molecular Biology (ISMB '00).*, pages 329–343, 2000.

[13] S. Umeyama. Least-Squares Estimation of Transformation Parameters Between Two Point Patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-13(4):376–386, April 1991. Comparing proteins 3D structure algorithm.

[14] R.; Verbitsky, G.; Nussinov and H. Wolfson. Structural comparison allowing hinge bending, swiveling motions. *PROTEINS: Structure, Function and Genetics*, 34:232–254, 1999.

[15] `http://gened.emc.maricopa.edu/bio/bio181/BIOBK/BioBookCHEM2.html`.

[16] `http://zebu.uoregon.edu/internet/images/peptide.gif`.