| Algorithms for Molecular Biology | Fall Semester, 2000 |
|---|---|

# Lecture 13: January 22, 2001

*Lecturer: Haim Wolfson* — *Scribe: Dina Duhovny, Aviad Tsherniak*

This document contains 4 major topics:

- Introduction to Protein Structure
- 3D Protein Structural Alignment Algorithms
- Molecular Surface Representation
- Docking (an overview)

## 13.1 Protein Structure Introduction

### 13.1.1 Background

Proteins are long chains of Amino Acids (AA). There are 20 types of AA that serve as building blocks for proteins. Each AA has a specific chemical structure. The length of a protein chain can range from 50 to 1000-3000 AA (200 on the average). One of the interesting properties of proteins 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). Researches of 3D structure of proteins have shown that when a folded protein is artificially stretched to a chain, it folds back to it's original 3D structure. 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. All proteins whose structure is known are stored in the Protein DataBank (PDB) which contains about 10,000 proteins [6].

A protein has multiple levels of structure (see figure 13.1):

- Primary structure - Chain of Amino Acids (1 dimensional)
- Secondary structure - Chains of structural elements, most important of which are $\alpha$-helices and $\beta$-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:

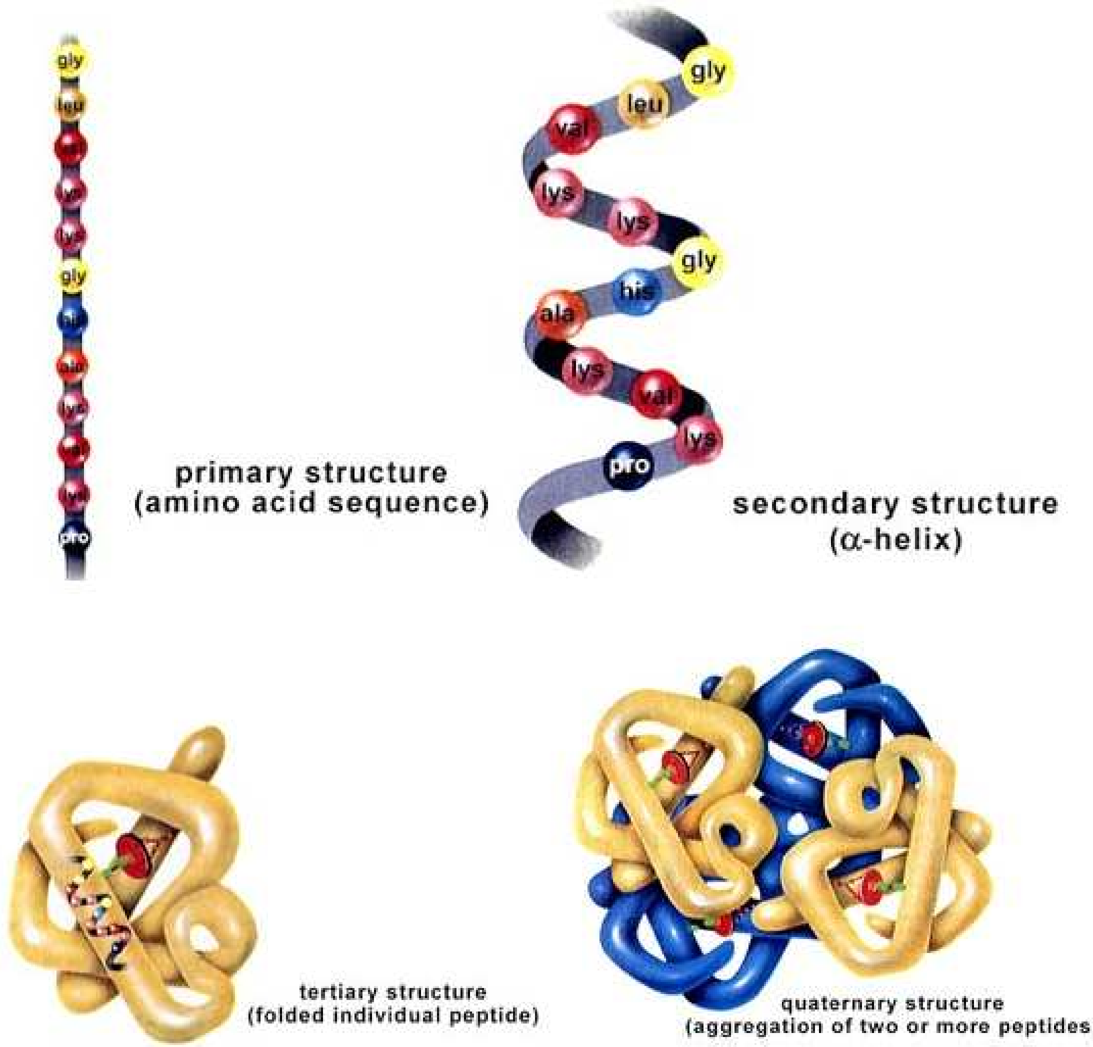


Figure 13.1: The four structuring levels of a protein. Taken from http://gened.emc.maricopa.edu/bio/bio181/BIOBK/BioBookCHEM2.html

- X-ray diffraction and neutron-diffraction
- 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.

### 13.1.2 Motivation for Protein 3D Structure Prediction

The structure of the protein is directly related to the protein's functionality, probably even determining it. 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 (instead of using, for example, AA sequence comparison algorithm) are:

- Protein 3D structure is more highly conserved than the primary structure.
- We can learn about similar function out of just partial surface similarity (e.g. active sites)

### 13.1.3 Protein 3D Structure

*The main hypothesis is that a protein folds to one unique structure, which depends only on the AA sequence.*

The common 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 can not overlap).

We will use Van der Waals radius balls as a 3D model of an atom. Each AA has a carbon atom called $C_\alpha$, connected to a carboxyl group and an amine group, a hydrogen atom and a part that depends on the specific AA - the *residue*. Amine group of one AA connects to the carboxyl group of the next adjacent AA (see figure 13.2). The $C_\alpha$-s form together a backbone wire, to which the rest of the atoms are attached.

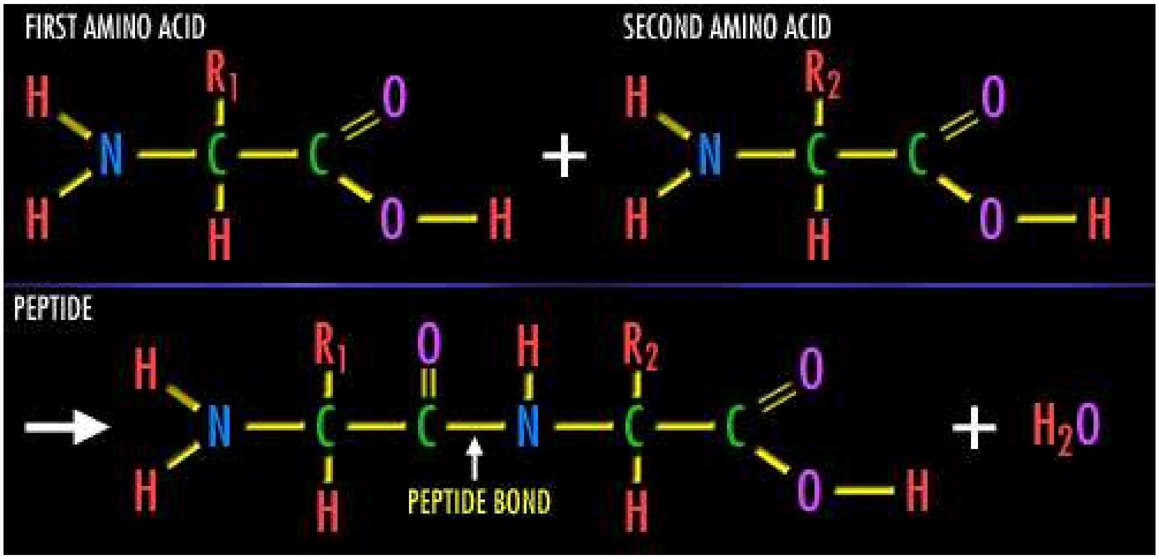


Figure 13.2: Formation of a peptide bond between two amino acids by the condensation (dehydration) of the amino end of one amino acid and the acid end of the other amino acid. The above image is from http://zebu.uoregon.edu/internet/images/peptide.gif.

### 13.1.4 Methods for Proteins Folding

Similar protein sequences have similar 3D structure. 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 assume, that the amino acids are divided to two kinds: hydrophobic (water hater) and hydrophilic (water lover). Using this model we can try to build the 3D structure minimizing rejections and maximizing attractions between nearby amino acids. Even solving this model was proven to be NPC, although there are heuristics for it (the best known has $\frac{2}{3}$ approximation ratio).

### 13.1.5 Structural Genomics Project

The main idea is to find 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. We also know that the number of possible protein folds is relatively small (there are about 600-700 different folds among the 10,000 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, than 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.

## 13.2 Protein Structural Alignment - The Rigid Case

### 13.2.1 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 $C_\alpha$ atomic centers (see figure [13.2])
- $C_\alpha$–$C_\beta$ vectors
- Secondary Structure Elements ($\alpha$-helices, $\beta$-sheets)
- Molecular Surface Representation (see section [13.3])

### 13.2.2 Problem Definition

The input to protein pairwise structural comparison is a set of 3D atomic coordinates of two different molecules (see figure [13.4]).

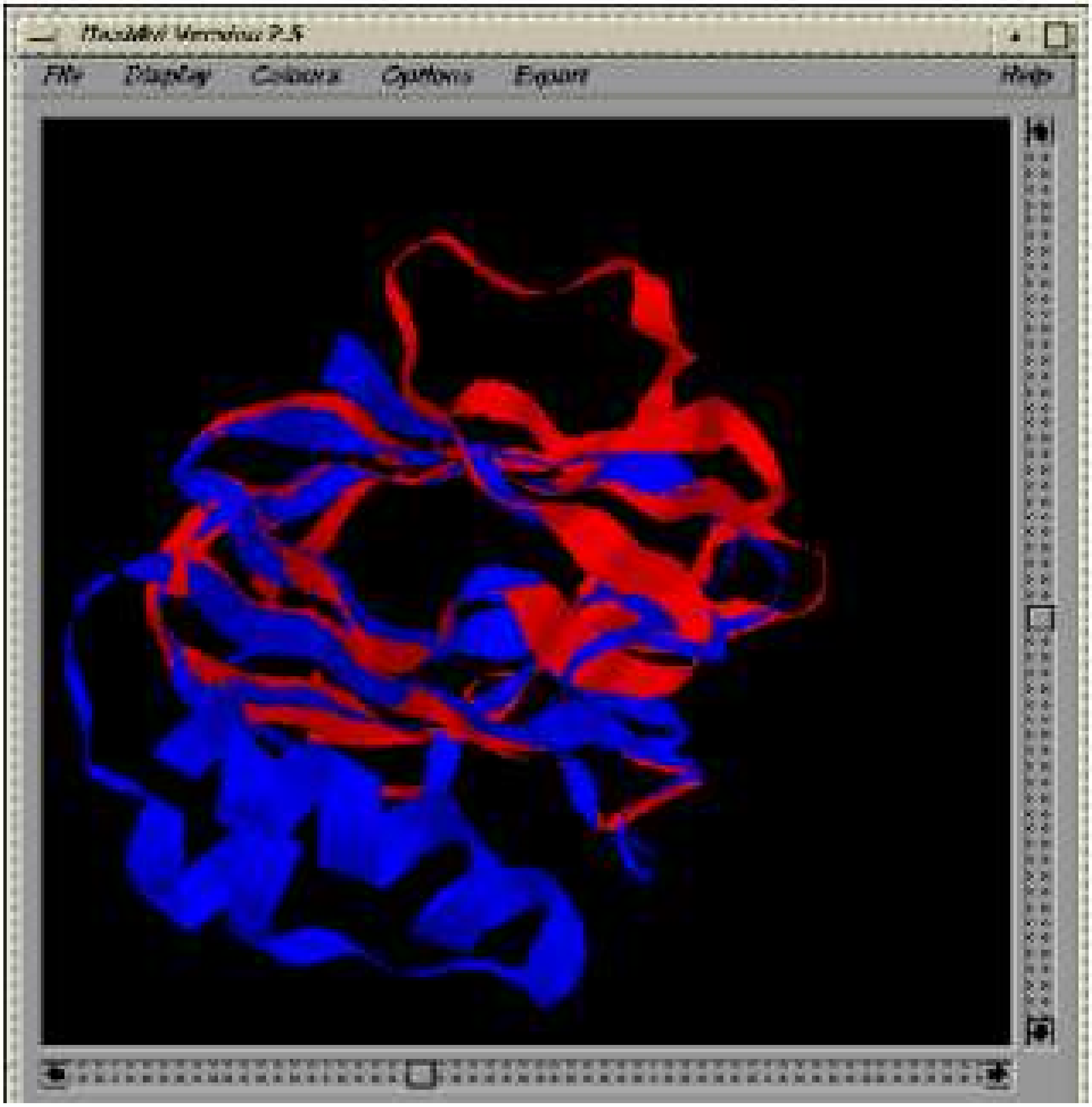


Figure 13.3: Output of the $C_\alpha$ matching program [4]

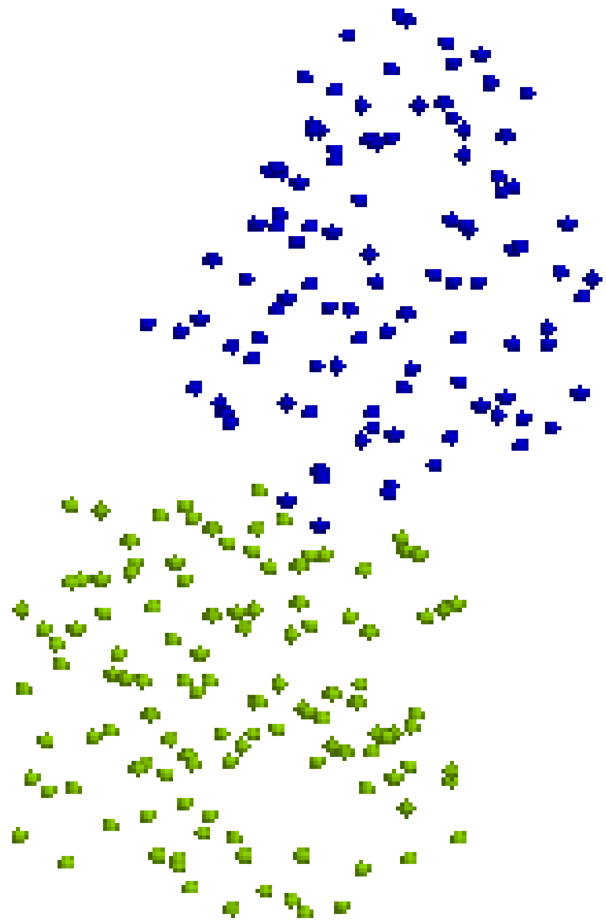

Figure 13.4: $C_\alpha$ coordinates of input molecules

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 a polypeptyde chain, thus reducing a problem to 3D curve matching, that is essentially a 1D task.

### Advantages of the Sequence Independent Alignment

Since the techniques don't 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.

**Sequence order independent** - align features in 3D space, a real 3D 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

Figure 13.5: Input Molecules: 1aaj and 1pmy

order of chain is exploited.

### 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 [13.6]) all the instances of models. The models appearing in the image are usually transformed by an *a-priori* unknown transformation, they can also be partially occluded. The solution is shown in figure [13.7].

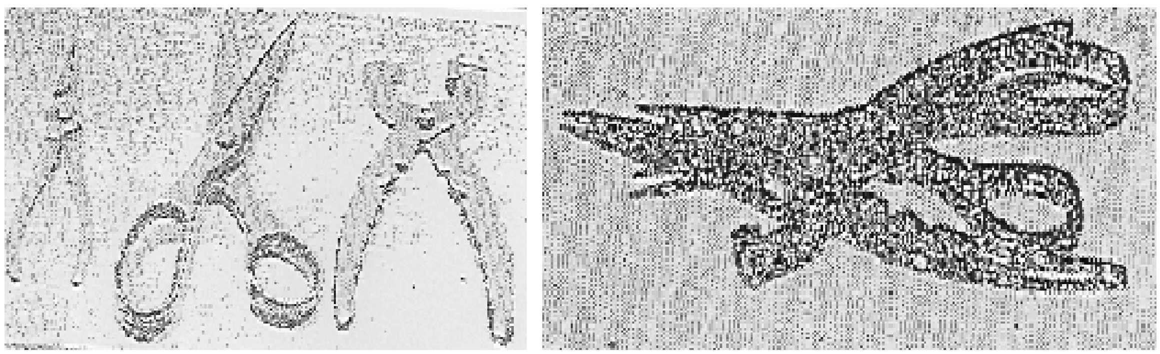

Figure 13.6: model and target images from [8]

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 transformation of one structure into another with minimal RMSD (Root Mean Square Deviation)

### Superposition - best least squares (RMSD) rigid alignment

Given two set of 3D points: $P = \{p_i\}, Q = \{q_i\}, i = 1...n$
find a 3D rotation $R_0$ and translation $a_0$ such that:
$min_{R,a} \sum_i |Rp_i + a - q_i|^2 = \sum_i |R_0 p_i + a_0 - q_i|^2$
A closed form solution exists for this task. It can be computed in *O(n)* time (Umeyama, 1991 [12]).
The problem is related to the well known Procrustees problems in statistics and involves eigenvalue analysis of a correlation matrix of the points.

**Solving the Correspondence (Matching) Problem**

We exploit the fact that our objects are rigid.
In this case the correspondense of a pair of ordered triplets of points ( *"fat enough"* triangles), uniquely defines a 3D rigid transformation (see figure [13.8]).

This leads to the following algorithm:

### 13.2.3 Straightforward 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 order of n points in both molecules - $O(n^7)$.

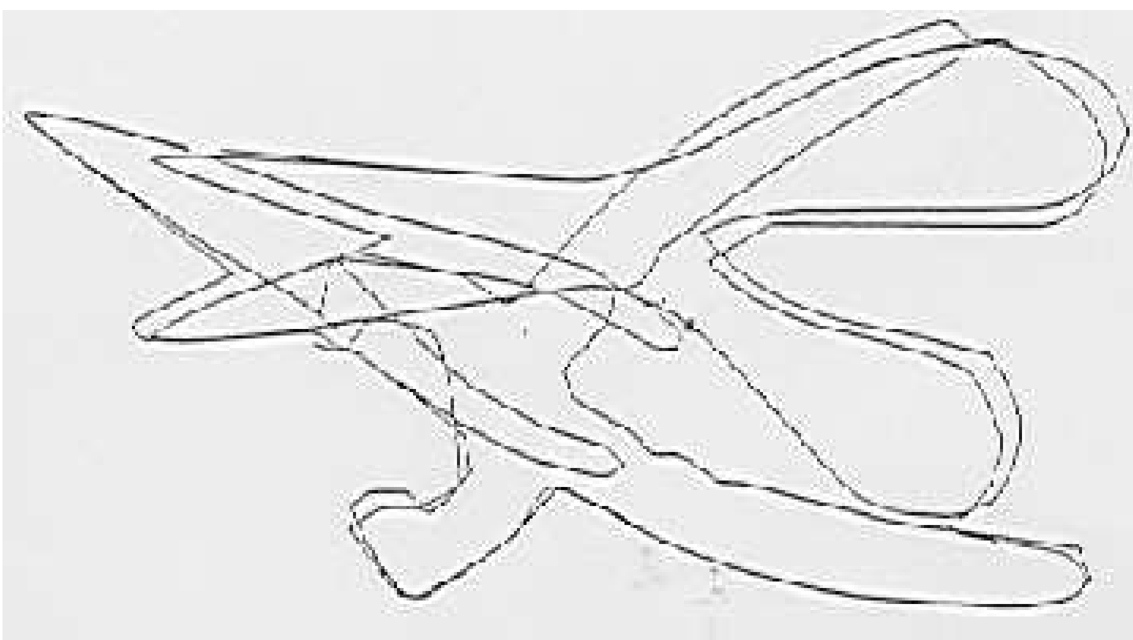

Figure 13.7: The solution

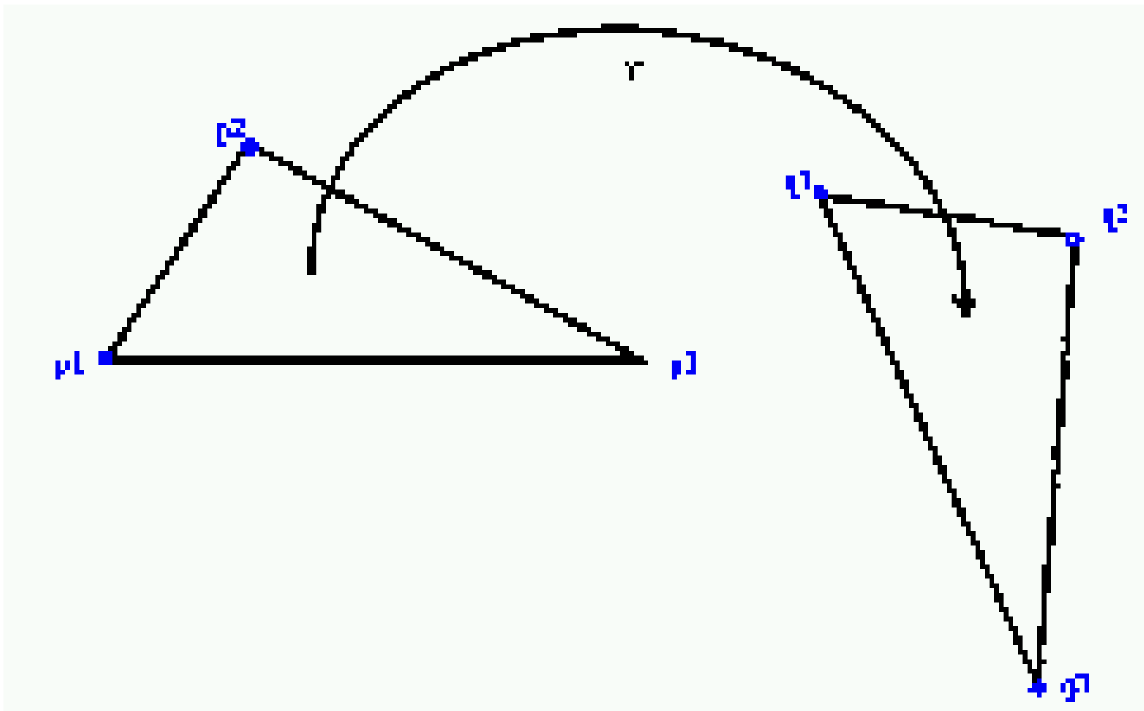

Figure 13.8: Transformation of one triangle into another

### 13.2.4 Geometric Hashing techinque based Alignment Algorithm

This approach was originally developed for object recognition in Computer Vision (Lamdan, Schwartz, Wolfson, 1988-rigid [7], Wolfson, 1991-flexible [8]).

**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.

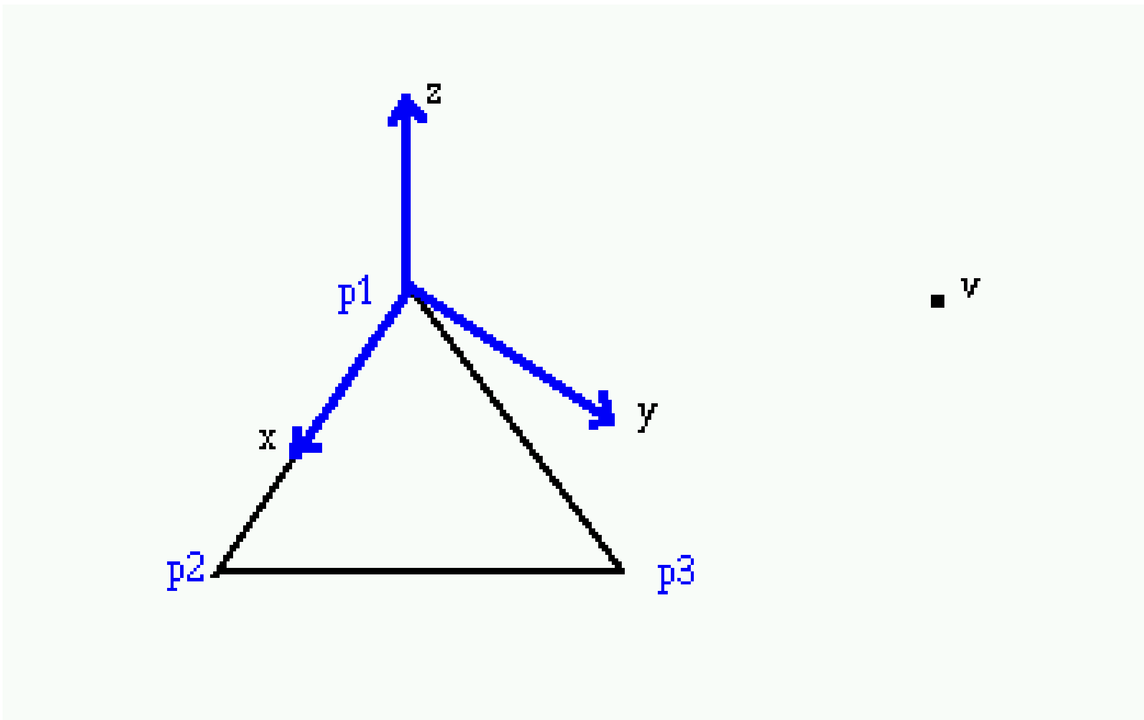


Figure 13.9: Example of possible Reference Frame choice

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 a unit length vectors (see figure [13.9]).
Suppose $e_x, e_y, e_z$ are the relevant unit vectors, than each point $v$ in the 3D space can be represented in 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.

### Geometric Hashing Technique

Geometric Hashing is a technique for matching a target molecule (or a collection of such) against a set of one or more models(e.g. database). The models are assumed to be known in advance, and thus allow preprocessing, as follows:

For each *model* object (first molecule in our case or each molecule in database) do:

1. Pick a reference frame.
2. Compute the 3D orthonormal basis associated with this reference frame and its shape signature (e.g. 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 above steps for each model reference frame (noncollinear triplet of model points).

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:
2. Compute the 3D orthonormal basis and the shape signatureassociated with it.
3. Compute the coordinates of all other points in the current reference frame.
4. Use each coordinate to access the hash-table and retrieve all the records [protein id, ref. frame, shape sign.,point].
5. For records with matching shape signature "vote" for the pair [protein, ref. frame].
6. 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:

**Alignment Algorithm**

We first define local neighbours 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 practise, 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:** $O(R * n * s)$

### 13.2.5 The Flexible case

**Motivation**
Proteins and drugs are flexible entities. There are two kinds of motion within the protein: hinge motion and shear motion [3].
The geometric hashing approach can be extended to take care of flexible case as well as rigid [13].
There are other successfull approaches to solve the problem, see [11].

## 13.3 Molecular Surface Representation

Representing a protein by its molecular surface representation helps us in the study of protein folding [9], in prediction of biomolecular recognition, detection of drug binding 'cavities' and Molecular Graphics.

A common representation is due to Connolly [2]. 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 + normals to the surface. These are sampled according some required sampling density (e.g. 10 pts/Å$^2$).

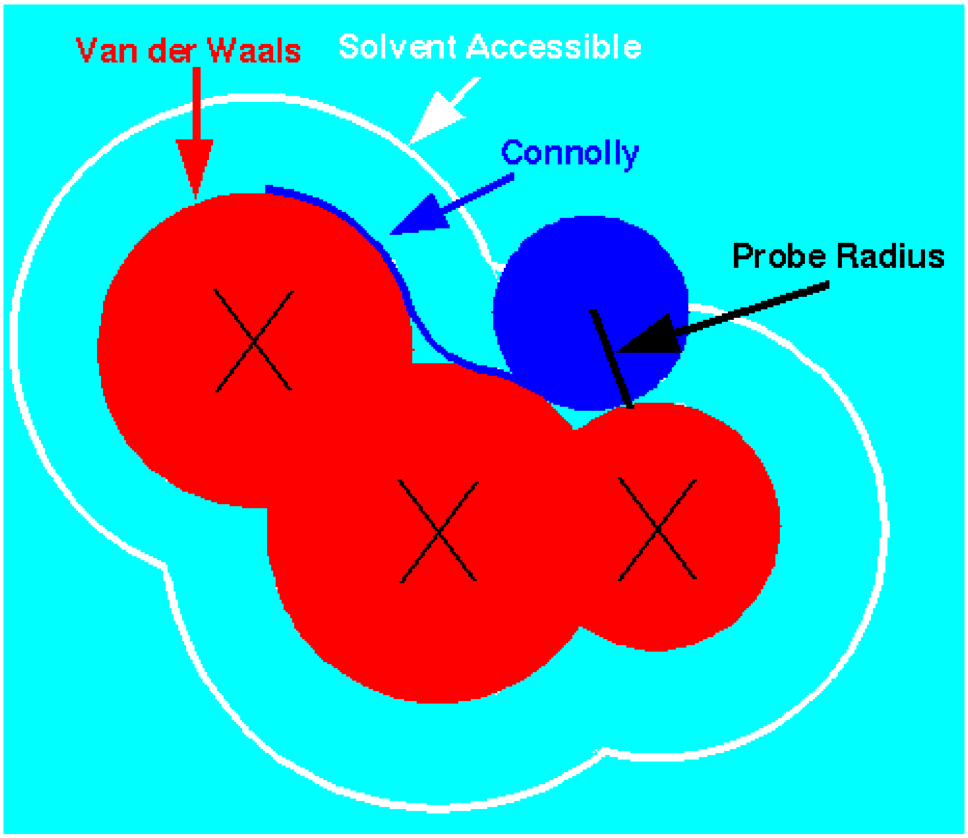


Figure 13.10: Connolly's Solvent Accessible Surface. Image taken from http://www.chem.leeds.ac.uk/ICAMS/eccc/cangaroo.html.

There are different ways to represent shape complementarity. The points + normals representation [2] is a dense one. The points representation in [10] is sparser, and so is the Solid Angle local extrema [1]. Another representation is SPHGEN [5] - 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 13.12 and 13.13.

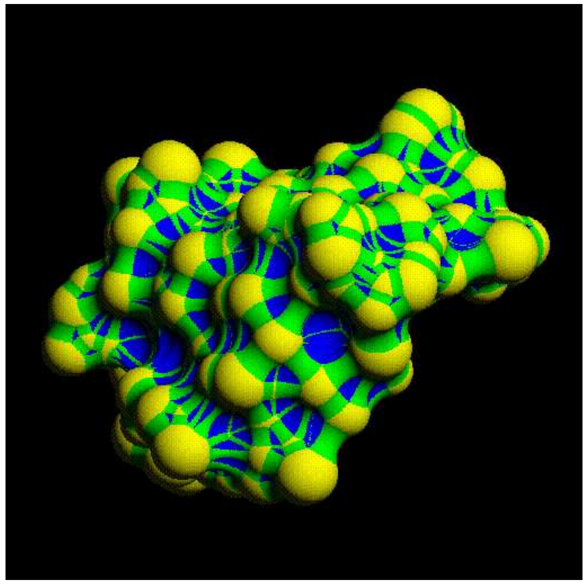

Figure 13.11: The molecular surface of Crambin is shown above, with the convex spherical patches colored yellow, the saddle-shaped pieces of tori colored green, and the concave reentrant surface colored blue. Image taken from http://www.netsci.org/Science/Compchem/feature14f.html.

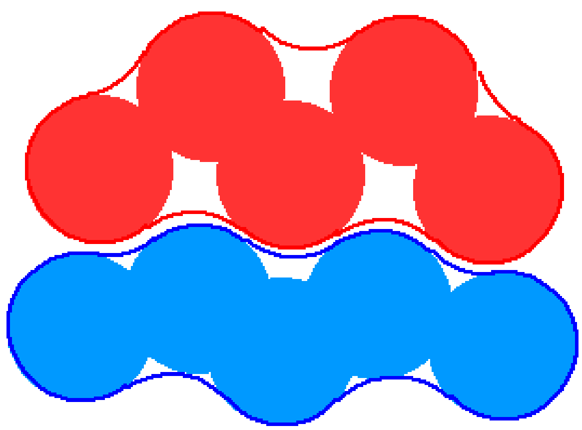

Figure 13.12: Shape complementarity at interfaces. Taken from http://www.netsci.org/Science/Compchem/feature14e.html

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 13.14. 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 13.15.

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 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

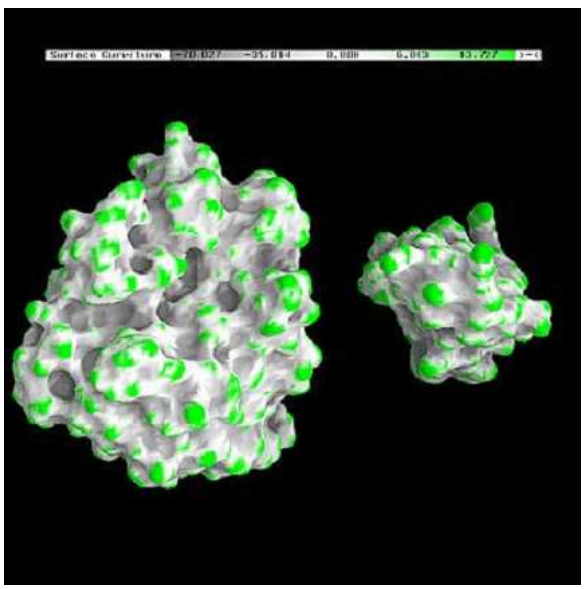


Figure 13.13: Trypsin/Trypsin inhibitor. Figure from B. Honig's Labs web-site at Columbia University.

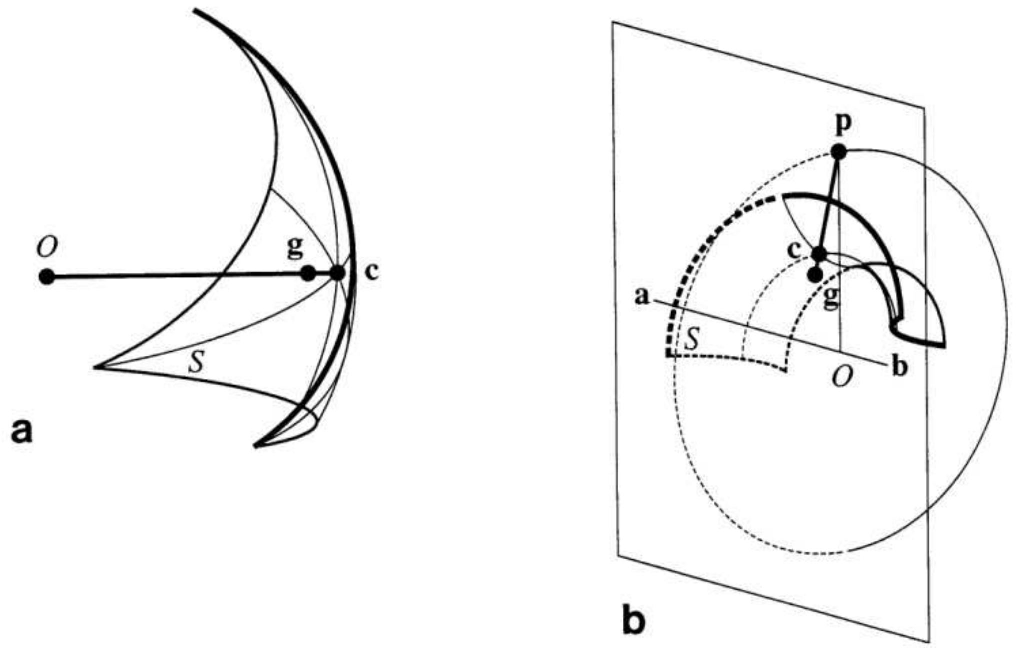


Figure 13.14: An illustration of the generation of (a) *caps* and *pits*, and (b) *belts*, respectively. Symbols are as follows: $S$ is a Connolly face, either convex (for caps) or concave (for pits) (a), and saddle-shaped (for belts) (b); $O$ in (a) is either the atomic center (for caps) or the probe 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 throught **c** and **g** intersects the plain normal to **a**b, that passes through $O$.

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.

## 13.4 Docking

**Problem 13.1** 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:

- **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 occuring 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

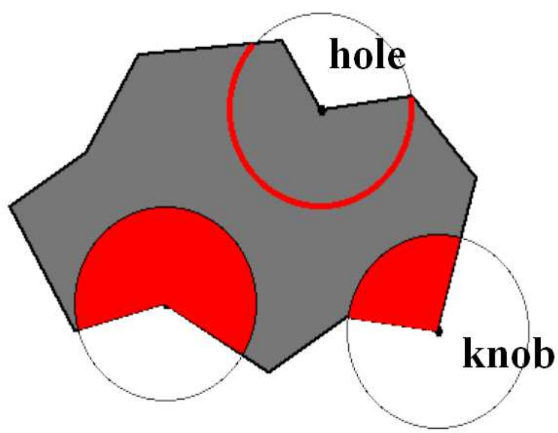


Figure 13.15: Solid Angle local extrema. Taken from http://www.netsci.org/Science/Compchem/feature14h.html

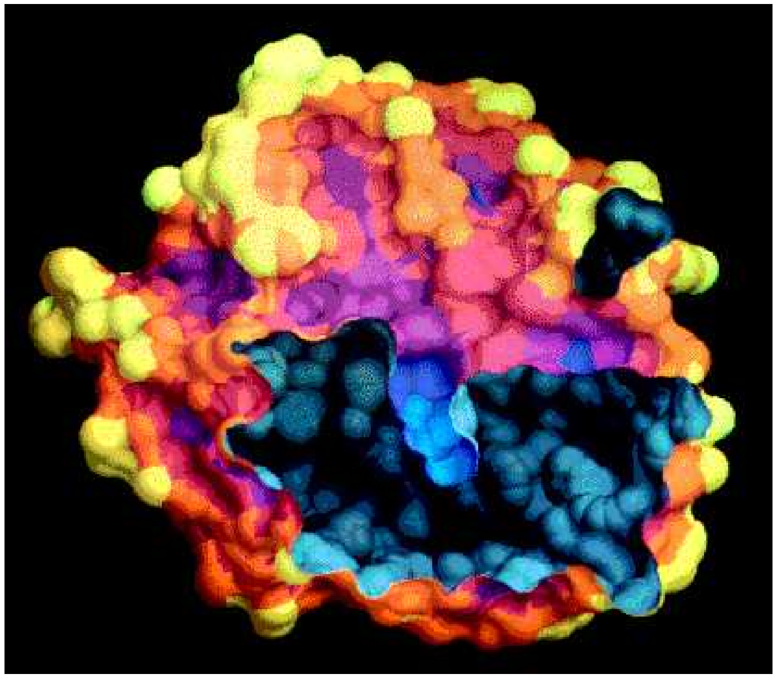

Figure 13.16: The chymotrypsin surface above has been colored according to convexity or concavity. A sphere of radius 6 Å was centered at several points on each surface face, and the solid angle of the sphere lying inside the protein's molecular surface was computed and averaged over the face. Each face was colored according to where its average solid angle fell in the range between zero and four $\pi$ 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.

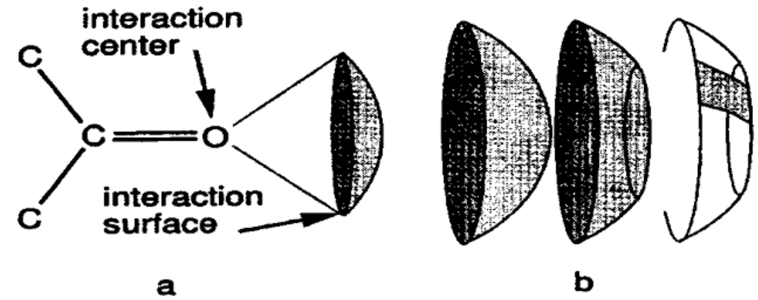


**Figure 1.** Interaction geometries. a, Interaction center and surface pertained to a carbonyl group. b, Three of the four different types of interaction surfaces: cones, capped cones and spherical rectangles.

Figure 13.17: Hydrogen bond complementarity modelling.

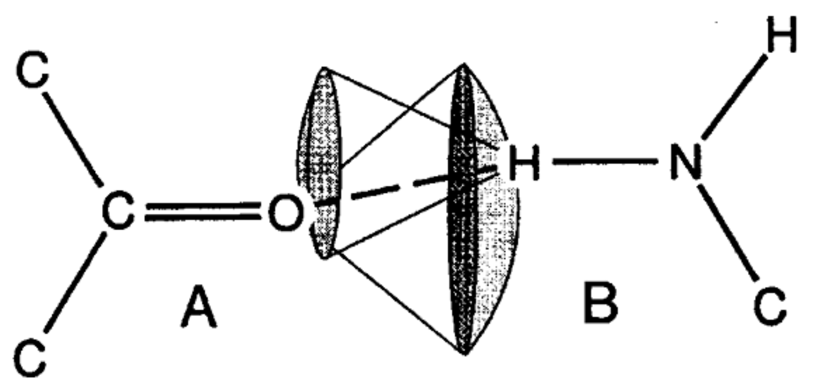


**Figure 2.** Condition for the formation of interactions: a hydrogen bond between the carbonyl oxygen and the nitrogen. The interaction centers are the oxygen and the hydrogen atom forming the hydrogen bond. They have to fall mutually on the surrounding interaction surfaces.

Figure 13.18: Hydrogen bond complementarity modelling.

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 13.19).

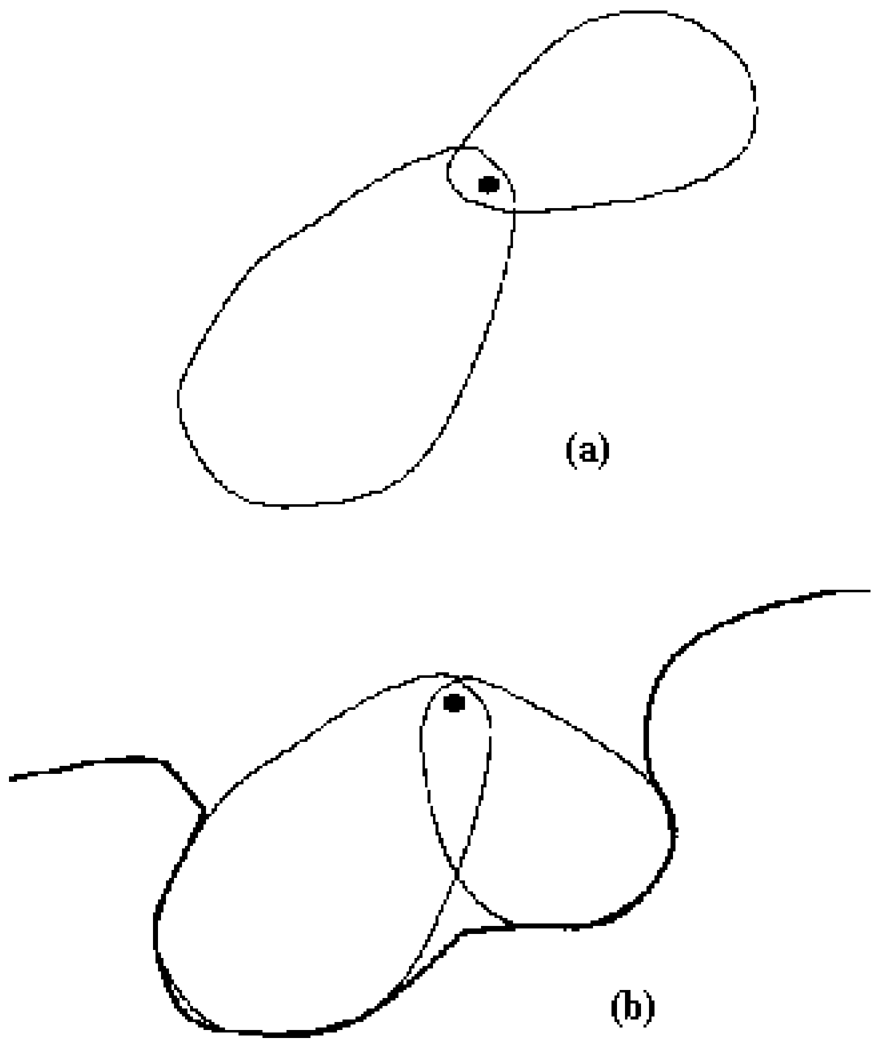


Figure 13.19: A molecule (two rigid parts and one hinge) and a receptor

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:

- 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.

On figure 13.20 you can see a sketch of the different stages a docking method has.

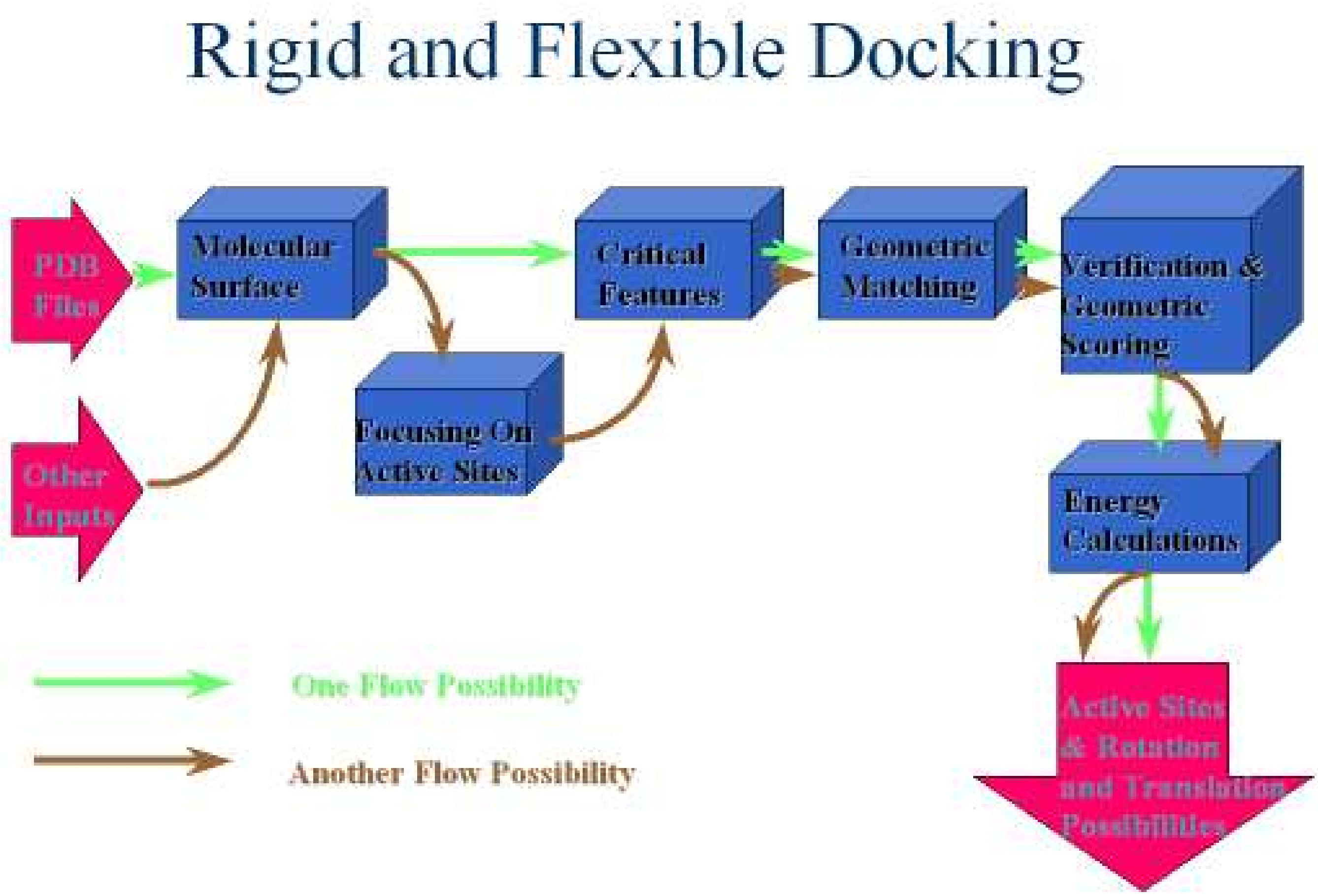


Figure 13.20: Rigid and Flexible Docking.

# Bibliography


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

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

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

[4] Nussinov-Wolfson group of Computational Structural Biology. The C-Alpha Matching program. *http://pc − gamba.math.tau.ac.il/software.html*.

[5] 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.

[6] Brookhaven National Laboratory. PDB - Protein Data Bank. *http://www.rcsb.org/pdb/index.html*.

[7] 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.

[8] 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.

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

[10] 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.

[11] M. Shatsky, Z.Y. Fligelman, H.J. Wolfson, and R. Nussinov. Alignment of flexible protein structures. In *Proc. 8th International Conference on Intelligent Systems for Molecular Biology (ISMB '00)*, pages 329–343, 2000.

[12] 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.

[13] G. Verbitsky, R. Nussinov, and H.J. Wolfson. Structural Comparison Allowing Hinge Bending, Swiveling Motions. *PROTEINS: Structure, Function and Genetics*, 1998. *in press.*