A Survey of Geometric Methods in Protein Docking Problem

Gayathri Priyalakshmi Ayyappan*

May 12, 2004

Abstract

This paper surveys the computational methods applied in the field of protein docking, which is related to the well-known protein folding problem. The focus is on geometric approaches to solving the problem.

1 Introduction

We recall that in the well known Protein Folding problem the main goal is to determine the 3-D structure of a protein given its amino acid sequence. This has been a most intensely researched problem since the 1950s, and has been an active area of application for computational geometry since the 1980s.

The Protein Docking problem, although is somewhat related to folding is often considered to be a distinct area, yet the tools developed for the former problem is useful for the latter one [8]. The interest in computational methods for molecular docking is because of its importance in studying the protein structures. With new advances in genetic engineering it is being realised that the complexity of human beings lie in the proteins produced by the human body. The biologist would often want to learn what is the function of a specific protein. Proteins do not act on their own, therefore we need to understand how they interact with other components of the cell in order to determine the function of a protein. These interactions of the protein come under the umbrella of molecular docking.

Here we specifically survey the area of protein docking in terms of the computational geometry applications therein. The area is of central significance to the field of pharmaceutica
dustry due to the fundamental assumption of drug design that drug activity is obtained and understood through the molecular binding of one molecule, the ligand, to another molecule, the receptor. When considering enzyme-inhibitor association, it is generally referred to as

*ID# 104667554. prigayat@cs.sunysb.edu
Figure 1: Molecular Docking: A trypsin-inhibitor molecular interface. The inhibitor (BPTI) is on top.

protein-protein docking, and for antibody-antigen association, it is receptor-ligand docking.

The Protein docking problem, or simply the docking problem, is to determine, given two molecules, whether they can associate with each other, i.e., is there a conformation in which they can bind together and form a complex and to predict the geometric structure of this combined complex [8]. One of the molecules is the receptor, it has a binding site into which the other molecule, the ligand, can dock to form the combined complex. The inputs are the atomic coordinates of the two molecules. In the general problem, theoretically, no additional data is provided.

In practice, however, additional information like the location of the primary binding site is sometimes provided. In this case, the docking problem reduces to predicting the structure of the complex. This problem is called the bound docking problem. When the bound structure of more than one molecule is known, usually a cocrystal of the ligand and the receptor, the structures of the molecules of interest can be extracted and then the structure of the combined complex can be determined [8]. This is often a reasonable assumption as the location of the binding site can be frequently inferred by comparsion with the protein cocrystallised with a different ligand or comparison with proteins of similar function [4].

The general problem with unbound molecules is more difficult and is called the unbound
docking problem. For this case, the inputs are either the native structure of the unbound molecules (i.e. the structure of the molecule when it is free, in its uncomplexed state), or a pseudo-native structure (i.e. the structure of a molecule when bound to another molecule different from the one used for the docking problem) or a modelled structure.

The interest in computational methods for solving the problem is due to the difficulty in experimental determination of the complex structures. As we noted earlier, the docking problem is very important for the computer aided drug design in the pharmaceutical industry [10]. A fundamental assumption of drug design is that drug activity is given by the molecular binding of the drug molecule to the protein. By binding, the drugs modulate signal pathways, for example, by altering sensitivity to hormonal action, or alter metabolism by interfering with catalytic activity of enzymes.

2 Aspects of the Problem

There are three key aspects in solving the docking problem –

a) representation of the molecules
b) conformational space search
c) ranking of potential solutions.

The three parts are inter-related since the choice of the representation determines the types of search algorithms that can be applied and this in turn determines the scoring function used to rank the potential solutions.

The molecular surface can be represented by mathematical models using shape descriptors or other geometric models like spheres for atoms of the molecule. Alternatively, the modelling of the protein frame can be static or dynamic i.e is the molecule considered to be rigid or flexible. The algorithms we consider use both kinds of representations.

Various search algorithms that have been used for docking are: Monte Carlo methods, genetic algorithms, fragment-based methods, point-complementary methods, distance geometry methods, tabu searches and systematic searches. Molecular dynamics methods find conformations that minimize the global energy of the docked complex. Standard optimization techniques such as least squares method, simplex are used. Monte Carlo methods for docking use random sampling techniques or a Markov chain. Programs that use these are AutoDock, ProDock, DockVision. Fragment based methods divide the ligand into separate fragments, dock the fragments and then finally link the fragments together. Tabu searches are based on stochastic processes in which new states are randomly generated from an initial state which are ranked. The best solution now serves as the start state for the next iteration. While both molecular dynamics and Monte Carlo methods are theoretically capable of
estimating the thermodynamic properties of a receptor-ligand interaction, they require very large amounts of computation time for ligands with many degrees of freedom. We shall focus our attention on geometric methods for molecular docking in this survey.

Scoring functions identify the best solution generated by a search algorithm. Structural parameters of the molecules are used to estimate the strength of the interaction and free energy is used to provide an estimate of the binding affinity [4]. Current docking methods use the scoring function in two ways. The first approach is to use the scoring function to rank a protein-ligand conformation, the system is then modified by the search algorithm and the same scoring function is applied to the new structure. In the other approach a general scoring function is used to first filter out possible solutions and a more specific scoring function is applied to this set to obtain the final solution. The scoring functions that are used are heuristic in nature. A good scoring function to rank the solutions is an open problem.[3]

The performance of the docking algorithm is evaluated using benchmarks such as CASP2 (Critical Assessment of techniques for protein Structure Prediction). The benchmarks assess the ability of docking algorithms and techniques to predict the correct interactions between two proteins. At the beginning of such docking trials the crystal structure of the complex in question is publicly unknown but have been solved. The assessments are submitted to the host with no prior knowledge of how accurate the prediction is. The basis of comparison of the docked results in the root mean square deviation (RMSD) from the experimentally determined structure [4]. It was however, found that there is trade-off between accuracy and speed of the algorithms. Therefore, RMSD should not be the sole criterion for evaluation.

3 Difficulty of the problem: Degrees of freedom

A molecule is characterized by a pair (A,B) in which A represents a collection of atoms and B represents a collection of bonds between pairs of atoms. Three pieces of information is associated with a bond: bond length (the distance between bond centers), ii) bond angle (the angle between consecutive bonds), and iii) whether the bond is rotatable or not. Since bond lengths and bond angles do not change significantly, they are considered to be fixed. Thus the degrees of freedom come from the rotatable bonds. The 3-D embedding of a molecule when values are assigned to its rotatable bonds is called the conformation of the molecule. Ligands usually have 3-15 rotatable bonds and receptors have 1000-2000 bonds. The dimension of the search space makes the problem computationally intractable [27], [18].
4 Geometrical Approaches to Docking

We now describe the various geometric approaches that have been used to solve the docking problem. Initial approaches to the problem assumed the molecules were rigid bodies in order to overcome the tractability with the many degrees of freedom. Newer algorithms can handle the case where molecules are flexible.

Most geometrical methods for docking problem can be classified as algorithms for rigid protein-rigid ligand, rigid protein-flexible ligand, and flexible protein-flexible ligand. The principal computational geometrical methods that have been applied to docking are from the areas of surface matching, object recognition and motion planning.

4.1 Docking as a Surface Matching problem

We assume that the atomic coordinates for the molecules are available and the receptor and the ligand are rigid bodies. The fundamental assumption for the geometric approach to the docking problem is the necessity for surface complementarity of the two interacting molecules. Since two rigid molecules interact along their surfaces, we can view the docking problem as a surface matching problem [9]. Two surfaces are considered to match each other if there is a correspondence between points on the surface, i.e., we can apply a rigid body transformation to one molecule we can get a point-to-point contact between two surfaces. The goal is to find maximum match between the two surfaces.

A typical computational geometry approach is to construct a Minkowski sum of the two rigid objects [9]. Once the Minkowski sum of the two objects is constructed, we know the number of contacts in the features (points, edges, planes) of the two molecules. Each feature is associated with a rigid body transformation. If we construct the Minkowski sum, the contact between the two molecules corresponds to a 5-D hypersurface inside a 6-D conformational space. The computational complexity is $\omega(n^3m^3)$ where $n, m$ are the number of atoms in each molecule. Since the number of atoms is huge, the computation is impractical [9]. In order to make the problem feasible, the surface is discretized.

In another algorithm, Soft Docking [25], the molecular surfaces are represented by cube sets. This method does a form of exhaustive search in the 6-D conformational space. The rotations are discretized and for each discrete rotation a set of translations which cause the cube sets representing the molecules to come into contact is computed. The relative configurations which are output by the algorithm are those that maximize the number of cubes matched. This method implicitly allows for conformational changes caused by complex formation due to the cube representation.

An alternative approach is to represent the binding site with a number of sparse points and match these points against corresponding points of the ligand. Hence in this case the
docking problem can be viewed as a point matching problem. More recent methods adopt the above approach: they try to match features (points) of the binding pocket to features (points) of the ligand. The definition of matching features in the receptor and ligand vary with the method used.

In [13], the features of the ligand are the atoms on its surface considered as point masses while the receptor’s features are the centers of the pockets on the molecular surface of the receptor into which the ligand atoms may be placed. They define the docking problem as finding a relative configuration which maximizes the number of pairwise contacts made where a ligand atom and a receptor site point are considered to be in contact in a given configuration if the distance between them is less than some small fixed tolerance. Their algorithm uses a combinatorial graph, the docking graph. The nodes of the graph are ordered pairs of binding site of the receptor and molecule points of the ligand. There is an edge between two nodes if the distance between the site points is within the upper and lower bounds of the distance between the molecular points. A maximal clique in this graph will produce a maximal matching between receptor and ligand. It is well-known that the problem is NP-hard [6], their approach takes $O(p^{2.8})$ where $p$ is the number of nodes in the graph.

The CLIX algorithm [16] uses points of minima in interaction energy maps as centers to which it fits ligands. It tries all possible pairs of points in the ligand and receptor. At each step a match is sought between two interaction points and corresponding features in the ligand. This initial orientation is then optimized and those with good geometric and chemical complementarity are retained.

Other approaches such as [17] use a molecular representation which augments point information with surface normals for better discrimination

In [9], the molecular surface is first discretized by uniformly sampling points on the molecular surface. Then they place normals at the points local properties are captured by the normals and the points. The measure used for surface complementarity is the number of point to point contacts between the molecular surfaces. Since the surfaces are discretized by points, this match can be computed as surface normal to surface normal matching. Each normal is considered as a local coordinate system and a rigid transformation can be applied to match it with another normal. This gives one point-to-point contact. The rigid transformations are stored in hash table with the count of the number of points of contacts each results in. The rigid transformation with the largest count of matches is chosen.

The DOCK algorithm is one of the earliest docking algorithms. DOCK has been widely used in this field. In [14], the features used in DOCK are spheres. DOCK generates spheres inside the binding site in a way that they touch the surface of the pocket in two points and have their centers along the surface normal at one of these points. The centers of these typically overlapping spheres are the receptor’s matching points. Spheres are created in a
similar way inside the ligand and their centers are the matching points of the ligand. A ligand sphere can be paired with a receptor sphere if each sphere belongs to a set of spheres with the following property: the internal distances of all the spheres in the ligand set match all the internal distances within the receptor set. The suggested pairings are then ranked based on the degree of overlap.

After the essential points in the ligand and receptor have been identified, the docking problem reduces to a matching problem. Now, the algorithm systematically pairs each ligand sphere with each receptor sphere. For each such pairing a second pair is chosen which maximizes the number of further matches which could still be made without violating a simple intramolecular distance constraint. Then a third set of spheres is picked which maximizes the remaining possible matches and continue until no further matches can be made. Whenever at least 4 matches are made the orientation is uniquely defined and the match is retained to pass through other filters (such as “handedness test” and energy evaluation). Several successful predictions have been made using this algorithm [24]. These tests were performed on pairs of molecules for which high resolution X-ray crystallography data is available for the docked complexes. In all reported cases, the algorithms found the correct conformations to a high degree of accuracy within 24 hours of computing time. However, the algorithm also found many false positives when using unbound molecules as the starting point [14].

In [5], Connolly used a shape description of the molecules. The complementary pattern of knobs and holes of molecular surfaces. He used a shape function to approximate the molecular surface and found critical points on the surface (local maxima and minima of the shape function). For each “knob” on one molecule the “holes” on the other molecule that have a complementary shape are determined. Next, quadruples of critical points (knobs and holes) on one surface are matched with those on the other surface for similarity. For each similar pair, a transformation is computed and it is scored based on Connolly’s scoring function.

However, a molecular shape is qualitative and hard to quantify. The shape is arbitrary and highly dependent on the molecules under consideration. The advantage of using a surface description as opposed to a shape description is that a good approximation of the molecular surface can be obtained which still maintains some distinct features of the real surface and does not require ‘features’ as in shape descriptions [9].

In [19], the surface complementarity as a criterion for docking of 26 bound and 19 unbound proteins is examined. Their study reinforces the importance of surface complementarity as a major determinant for successful docking.
4.2 Docking as a Model-based Recognition problem

Algorithms from the robotics domain of Model based Object Recognition have been successfully used in protein docking [20]. The problem is: given one or more object models and a scene possibly containing images of the objects, determine if the object occurs in the scene (object recognition) and if so, where does it occur (object localization). The object model may be thought of as the ligand and the scene corresponds to points on the receptor. The goal is to compute a rigid body transformation to align as much of the data as possible to one of the object models known to the program.

4.2.1 Interpretation Tree

In [20], the authors suggest that the techniques from model based object recognition can be easily mapped to solve protein docking. They discuss the interpretation tree introduced in [7]. The interpretation tree is a way of organizing the set of possible interpretations of the data. An interpretation of an image with respect to one of the object models is a set of matchings of the form \((s_i; f_j)\) where \(s_i\) is one of the sensed points and \(f_j\) is a feature of the object model. (Note: a feature could be an atom of the ligand). An interpretation could be a path from the root to leaf in the interpretation tree. Each edge along the path corresponds to one datum-to-feature matching of the interpretation. The usefulness of this data-structure is that it facilitates eliminating large sections of the search space by the application of geometric constraints. Useful constraints on which pairs of matchings are simultaneously feasible derive from the rigid body constraint of the object. For example, the partial interpretation \((s_i, f_j), (s_k, f_l)\), where \(f_i, f_l\) are model vertices, is feasible only if \(d(s_i, s_k) - d(f_i, f_l)\) < \(\epsilon\), where \(d\) is the Euclidean distance. The constant term \(\epsilon\) is a way of accounting for error in sensing data. (Similar constraints can be derived for other feature types; angle constraints involving triples of matchings).

The algorithm constructs an interpretation tree in a systematic way by matching each image data point in turn to a feature of the model. The \(i^{th}\) level of the tree contains attempted matchings of the \(i^{th}\) data point. From a given node \(T\), the subtree corresponding to a new matching \((s_i, f_j)\) is only generated if \((s_i, f_j)\) is compatible with all of the matchings already made on the path from the root of the tree to \(T\). Since all the image points lie on a single object, and three independent matchings completely define a rigid body transformation, three levels of the tree are built. At the leaf of the three level tree, the other \(m - 3\) points are checked by computing and applying the two possible transformations.

4.2.2 Geometric Hashing

Another technique developed for model based recognition, geometric hashing is now being applied to protein docking [1]. The idea is to invert the natural "imaging" mapping from the set of possible interpretations of the data to the set of sensor readings. The inversion is
made possible by a hash table data structure. Conceptually, the sensor data is used as an index to the table and interpretations are stored at each entry. To make such table lookups possible, the method uses transformation-invariant representations of objects. If one chooses fixed, minimum-sized tuple of object features which can uniquely determine a basis for a coordinate system for the object, all other feature can be described with respect to the basis. Three non-collinear points are necessary and sufficient to uniquely define such a basis. In geometric hashing, the hash table is filled with one entry for every feature coordinate value arising from every possible basis tuple. In the entry for a coordinate, the basis tuple (which gave rise to the coordinate value) and the model, in case there are more than one, are stored.

The recognition step then given an image picks a tuple of basis image points, computes the coordinates of all other image points with respect to the basis and looks up the table for each and tallies votes for the favourite basis among those retrieved from the table. If no clear favorite emerges from one image basis tuple, others are used and cumulative votes are tallied. Given the image basis points and elected overall favorite object basis, a transformation is uniquely determined.

One advantage of this approach is that the hash table for a set of possible models can be computed offline and then reused on many different images. Note that an interpretation tree cannot be precomputed as its branching topology depends on the model and the particular image. During the recognition step, all possible images may have to be tried in the worst-case, in which case, the precomputation does not save any work. But if many of the points in the image arise from just one of the modelled objects, the first few tuples tried are very likely to yield a good match right away and recognition will be efficient.

During the process of molecular association either of the participating molecules may undergo conformational changes, enabling their docking. The induced conformational transitions involve flexible movements of molecular parts, in the form of rotational movements of relatively rigid subparts about hinges.

In [22], the geometric hashing technique is used to solve the docking problem in which the ligand may have upto 3 internal rotational degrees of freedom, centered on a designated hinge atom. In [21], the technique is adapted to solve the more general docking problem which allows conformational flexibility in both the ligand and the receptor. The close analogy between rigid docking and object recognition extends to the flexible case as well and is known as articulated object recognition. Articulated objects are objects consisting of rigid parts which are connected either by rotary or sliding joints. The analogue of a hinge in the molecule is the rotary joint in the articulated object. The conformational transitions we allows rotational movements of the molecular substructures about the hinges in either ligand or the receptor.

Earlier the recognition task involved finding ligands (models) which had substantial sur-
face match with the receptor, now the problem has an additional constraint that we find a plausible conformation such that the parts of the ligand do not self-collide. The geometric hash is augmented with the hinge location for every stored ligand.

First, in the preprocessing phase: Every ligand is represented by a set of points. The hinge positions are picked as origins of 3-D cartesian coordinate frames called as ligand frames. The orientation of the frames is set arbitrarily. For every non-collinear triplet, in each ligand part, they define three unique triplet based Cartesian frames. When considering a triplet of points, an internal cyclic order is defined, so that the frames are positioned as follows: The origin of each frame is defined at the respective triplet point, the $x$-axis as the line from the point to the neighbouring point, the $z$-axis as the normal to the triangle plane and the $y$-axis is the cross-product of the $x$ and $z$ axes. The unique triplet of triangle-side lengths is the key in the hash-table. And the hash entry represents the hinge by the transformations between the triplet frames and the ligand frame. The angle spanned by the bonds connected to the hinge is computed. It will be used in the verification stage. As before, this stage is independent of the receptor.

In the recognition phase, the receptor structure is introduced into the system. The receptor is also represented by a set of points. All non-collinear triplets of points are considered for matching. The triplet based Cartesian frame is calculated as above. The length of the triangle sides are computed. This calculation is invariant under translation and rotation. Therefore, congruent ligand triangles should have similar values. The hash table is looked up with this triplet and candidate ligand frames are computed by applying the recorded transformations in the hash entry. The origins of the candidate ligand frames are the hinge positions. Votes are cast for the identity of the ligand molecule together with the location of the candidate ligand frames.

Ligand surfaces having a partial fit are recovered by choosing high scoring (ligand, location, hinge) triplets are chosen and as before, the one with the highest vote is picked This is the matching stage.

The hinge location describes the 3-D translation that the ligand has to undergo in this docking. The location resulting from the translation is verified to ensure that it does not cause ligand subparts to collide with one another (self-collision check) or with the receptor (collision check). The recognition phase requires $O(n^3R)$ complexity where $R$ is the hash-table look-up cost and $n$ is the number of points on the receptor. The roles of the ligand and the receptor can be interchanged due to the symmetry of the problem.

### 4.3 Docking as a Motion Planning problem

The traditional framework of robot motion planning is based on manipulating a robot through a workspace while avoiding collisions with the obstacles in this space Robot motion
planning is applicable to the study of receptor-ligand interactions due to the fact that a flexible ligand can be naturally modeled as an ‘articulated robot’ (as described in earlier section). This approach was first described in [23]. Given a starting and ending configuration of a ligand, motion planning algorithms attempt to determine a collision-free path between these two configurations. Paths are usually computed in the configuration space (or C-space) of the robot, which contains one dimension for each degree of freedom of the robot. The method of choice for docking is probabilistic road maps.

Here they sample from the space of all possible paths that a ligand may take as it binds to the receptor protein and effectively guess several possible intermediate configurations of the ligand and obtain a distribution of energetically favorable paths to the binding site via these intermediate configurations. For each path, they generate a “difficulty weight” that represents the energy barriers that the ligand encounters along the path. By randomly generating several starting and ending configurations they estimate the average difficulty of entering or leaving different sites on the receptor protein.

The ligand is modelled as an articulated robot with a free base. Each atomic bond of the ligand molecule maps to a joint of the robot with torsional freedom of motion. Bond angles and bond lengths are kept constant. The root atom, which represents the free base of the robot, is an arbitrarily chosen terminal atom from the ligand. It is given 5 degrees of freedom: 3 to specify its coordinates and 2 to specify the orientation of only one bond. Each additional non-terminal atom requires only a single torsional angle to define the orientation of the rest of the molecule. Bonds involved in a ring are modeled as being completely rigid (i.e., no torsional freedom), which is generally true of most organic molecules.

Probabilistic Roadmap Planners (PRMs) in [11] are particularly suited for molecular docking since they can efficiently handle robots with many degrees of freedom and are easily extended to represent energetic constraints in the configuration space. A roadmap, as the name implies, is a set of milestones or nodes, each of which is connected to several close neighbors by physically realizable paths. It is represented computationally as an undirected graph and is used to capture the connectivity of the configuration space in which the robot is maneuvering. The roadmap is constructed by selecting a set of milestones from the configuration space of the robot and then connecting each milestone to several of its nearest neighbors by a local path-planning algorithm. Complete representation of obstacles (in our case the receptor protein) in a high-dimensional configuration space is computationally impractical. Hence, milestones in a high-dimensional configuration space are not selected deterministically and are instead generated by sampling randomly from this space. The local path planner determines whether a path exists between any two nodes by connecting them with a straight line in configuration space and testing this path for collisions.

Once the roadmap is constructed, the robot can be maneuvered between two arbitrary initial and final configurations by simply finding, for these configurations, the two nearest
milestones on the roadmap to which it can be connected by the local planner. A path between these two milestones can then be computed by performing a graph search on the roadmap.

The fundamental difference between their application of PRMs and the prior applications of this technique is the use of energy values in the configuration space to estimate the naturally induced motion of the ligand. They not only need to find whether a path exists but also whether the path is “energetically favorable”. Thus, while traditional PRM planners require only a binary test to determine whether a particular point in the configuration space corresponds to a collision between the robot and the obstacle, here they need to compute a continuous energy value for all of these points. Since any given point in the configuration space corresponds to a particular configuration of the ligand, the total energy at this point is computed by summing the individual energy contributions of all ligand atoms in the corresponding configuration.

The two fundamental steps of the energy-based motion planning algorithm are – 1) generating milestones, and 2) constructing the roadmap using the local path planner.
Generating milestones

The model for the ligand assigns each degree of freedom of the molecule to a separate dimension in the configuration space. Hence, the ligand configurations are generated by simply assigning random values to each coordinate in this high-dimensional space. For each sample that is generated, its total energy is computed and used to determine whether or not the sample will be accepted as a milestone. The energy computation involves the van-der Waals energies and so implicitly models collisions between the ligand and the protein. Also the sampling process is biased towards generating milestone regions of low energy.

Constructing the roadmap

The probabilistic roadmap is constructed by first obtaining a set of $S$ randomly generated milestones and then connecting pairs of milestones using the local path planner. The following algorithm is used to construct the roadmap from the set of randomly generated milestones (i.e. nodes): For each node $i (0 \leq i < S)$ - 1) Sort all remaining nodes (i.e., with index $> i$) based on their distance from $i$, and 2) While the number of edges at node $i < N$, use the local path planner to connect $i$ to its first un-tested nearest neighbor (i.e., a node to which an edge has not yet been attempted with the local path planner). The above algorithm results in an undirected graph with approximately $N$ edges at each node.

The local path planner is a modification of the one described in [11]. The planner con-
nects two given milestones by a straight line in configuration space and determines whether this straight-line path is energetically feasible (i.e., collision free). The path is tested by discretizing the line segment into a series of consecutive configurations, each separated by a maximum distance. The path is accepted only if all the configurations along the path have energy less than a maximum energy threshold (usually 5-10 kcal/mol).

This approach has also been applied to protein folding [26].

5 Validation of the Geometric Approach

As we saw in the introduction, there are various aspects to the docking problem, at least two of which are most significant: one based on geometry (computing relative configuration of ligand and the receptor), and the other on energetics (free energy of a tentative match). While it is accepted that for correctly positioning a ligand onto a receptor, the question is which approach is more beneficial to begin with. The geometric methods have some distinct advantages: one does not need to know apriori the positioning and approximate orientation of the ligand within the receptor’s binding site. Also, the ligand could possibly be allowed to bind at another feasible site. The geometry based methods are based on the rationale that for two molecules to be able to bind together their molecular surfaces need to show geometric compatibility. Also, the study by Norel[19] reinforces the importance of geometry-based method like surface complementarity as a major determinant for successful docking.

Acknowledgements The figures in the survey are due to the following site –
http://www.netsci.org/Science/Compchem/

References


14


[9] D. Ierardi, S. Park, Docking Molecules by Geometric Surface Complementarity at Multiple Resolutions

[10] L. Kavraki, P. W. Finn Computational Approaches to Drug Design


[16] Lawrence, P. Davis. CLIX: A search algorithm for finding novel ligands capable of binding proteins of known three-dimensional structure. Proteins, 12:31-41


