

# Deriving Multi-level Protein Structures Through Data Mining

Hongyuan Li  
PhD Candidate

Srinivasan Parthasarathy  
Assistant Professor

Ohio State University

Primary Contact: [srini@cis.ohio-state.edu](mailto:srini@cis.ohio-state.edu)

## **Abstract:**

*Proteins are molecules with different levels of repeated structures or patterns. These patterns are of great interest to biochemists since their identification can enhance the study of protein function. Though a lot of patterns have been found using ad-hoc methods, there is no systematic approach to find all the important structures in proteins. Here we try to establish such a method to efficiently find the important structures within a protein. We discover these patterns building up from the atom level using data mining techniques. Initial results show that this method is quite adept at deriving important high-level structures ranging from the backbone peptide bond plane to the basic structures that make up a  $\alpha$ -helix structure.*

## **1. Introduction:**

Proteins can be thought of as a family of molecules that have different levels of repeated structures or patterns. The atoms that make up the protein can be thought of as basic first-level patterns. A protein usually contains at most five different kinds of atoms: C, N, O, H and S. Second-level patterns can be thought of as the chemical bonds connecting these atoms. These include carbon-carbon single and double bonds, carbon-nitrogen bonds, carbon-oxygen bonds, carbon-hydrogen bonds, nitrogen-hydrogen bonds and oxygen-hydrogen bonds etc. The third level pattern is the backbone peptide bond structure of the protein. A protein can be viewed as a sequence of peptide bonds linked together with side chains protruding from it. All the peptides together form the backbone of the protein. Higher-level patterns include the  $\alpha$ -helix structure, the  $\beta$ -sheet structure, the configurations of amino acids, and super-secondary structures etc. These patterns have been carefully examined by the biochemists since it is clear that there is a close relationship between protein structure and function. X-ray crystallographers and NMR specialists have studied over 15,000 of these protein structures. However, the process of deriving such structures has traditionally been extremely time consuming since the search space can very easily explode. In this paper we evaluate how such structures may be derived from protein databases such as PDB using *data mining* techniques.

Data mining is a relatively new area in computer science dealing with the problem of analyzing datasets, typically large datasets, and identifying interesting patterns in the data. While typically mining techniques such as association rules [AFS93, AS94], sequential patterns [SA96], classification [Q86], and clustering [JD88], have largely been applied to business domains, more recently there has been a thrust to identify new techniques or adapt existing techniques to scientific domains. Recently several approaches to identify structures in molecular biochemistry have been proposed. Perhaps the most related to the work presented herein is the work presented by Wang et-al [W97], who consider the problem of discovering common motifs across three dimensional molecule families and use this approach to classify unseen proteins. Our work is distinct from the above since we are interested in discovering frequently occurring sub-structures within a *single large molecule* (the largest molecule considered by Wang et al [W97] involved

62 atoms). Since the search space explodes with molecule size, scalability is an important consideration and we consider some novel pruning techniques to limit the search space. Substructure discovery and the utilization of background meta-data and inductive logic have also been considered by others in the context of biological and chemical compounds [DTK98, DCH95]. This work presumes background information (bonds, bond-type etc.) whereas in this work we assume only the atoms and their spatial coordinates.

The important contributions of our work can be highlighted as follows. First, we describe a method that will discover multi-level frequently occurring structures within a protein or protein families (or large molecules in general). The algorithm is robust to noise and uses some novel pruning strategies to reduce the search space. We describe the pruning strategies in Section 2, our representation of structures in Section 3 and other details of our algorithm in Section 4. Second, we analyzed these results and found that the structures our method finds represent important characteristics of the protein such as the chemical bonds, the backbone peptide bond structure and other higher order structures. In some sense since these structures have already been identified by biochemists, this result serves to validate our approach. These results are described in Section 5. Third, as part of ongoing work, described in Section 6, we are looking into the problem of automatically identifying a subset of the structures discovered, that are deemed more interesting than others. Our hope is that these interesting structures have more value, in terms of predicting protein functional behavior and in classifying proteins and distinguishing amongst protein families, than less interesting structures. Statistical methods for identifying interesting rules have been studied within the context of data mining, and we will leverage some of these for the task on hand paying particular attention to domain-specific issues.

## ***2. Key Pruning Strategies:***

A set of one or more atoms in a molecule forms an atomset. An atomset composed of  $k$  atoms is called a  $k$ -atomset. The three dimensional spatial relationships amongst atoms in an atomset induce a *structure*. Many atomsets within a single protein, involving the same atom types (C, N etc.) can have the same structure. The most important structures are either highly repeated within a molecule (such as chemical bonds, peptide bonds) or conserved in a family of molecules of similar function (active sites). An atomset is deemed similar to another atomset if it has the same structure. Representing structures is an important issue since this can influence the overall performance of our approach. We discuss this issue in further detail in Section 3.

An exhaustive search for interesting structures is infeasible even for a medium-sized protein molecule. For instance in hemoglobin, an average sized protein, there are 4384 atoms (ignoring Hydrogen atoms). There are 9,607,536 combinations involving two atoms (2-atomsets) and there are 14,033,407,584 combinations involving three atoms (3-atomsets). Also note that the important structures discovered by biochemists involve many more atoms. Comparing all these atomsets to extract structure information exhaustively is computationally infeasible. What is needed, is an effective way to prune the search space.

### ***2.1. Neighbor Pruning***

We use several techniques to do this pruning. The first, which we call neighbor-pruning limits the atomsets, that need to be evaluated to those where the distance between two adjacent atoms within the atomset is within a certain range ( $R$ ). The intuition as to why this will significantly prune the search space is easy to see from figure 1 that plots a graph between the distance ( $R$ ) and the number of average number of adjacent atoms within that range from a given atom. It is possible to lose accuracy if range is too small. To balance the calculation and accuracy,  $R=4.5\text{\AA}$  is used in most of the calculations in this study. This is almost 3 times the chemical

bond of C-C. At this R-value it is easy to see that on average each atom has around 31 adjacent atoms. Note, that for better quality of data, only crystal structures are used. Since hydrogen atoms are missing in crystal structures, they are ignored by our approach. However, hydrogen bonds are not necessarily missing since they involve two other (non-hydrogen) atoms. If these two atoms are within the prescribed range such structures will be found by our method. Therefore hydrogen bonds may be found indirectly by our method.

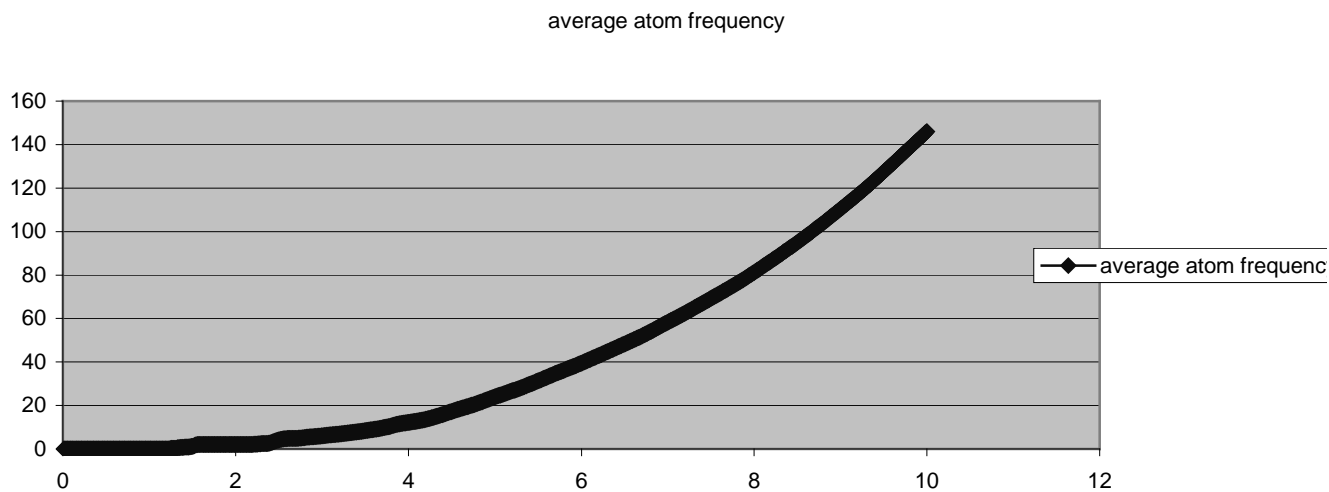


Figure 1: the average neighbor atoms vs. range. X-axis: the distance in angstroms. Y-axis: the average neighbor atom count. The calculation is based on the coordinates of hemoglobin (4384 atoms, PDB ID=1BZ0).

## 2.2. Using frequency of occurrence to identify important structures

The second pruning technique we use is to prune away those structures that occur infrequently. Below we offer some intuition as to why this simple technique works and can automatically discover important structures.

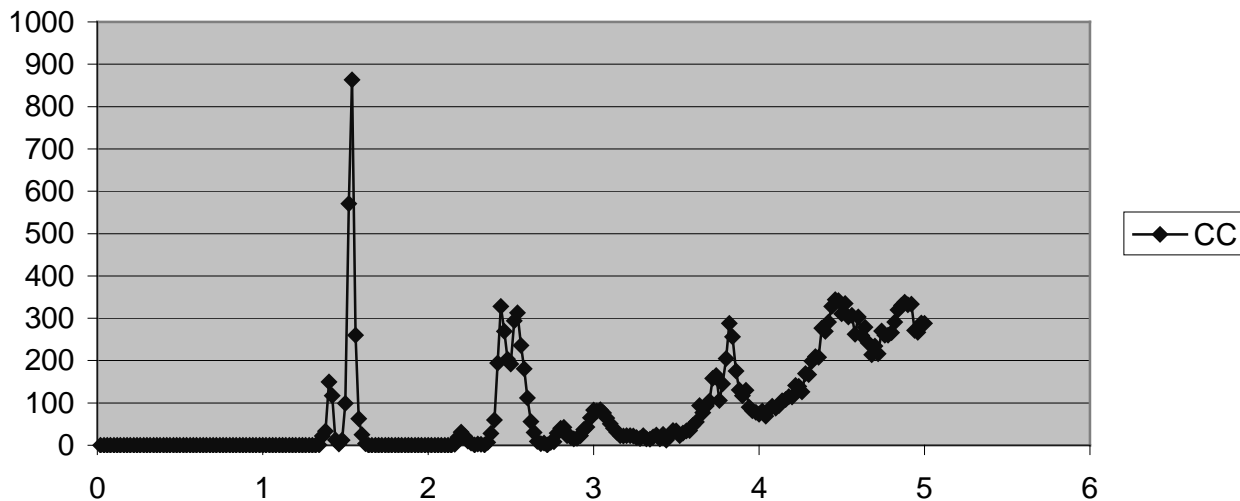


Figure 2. The distribution of the C-C relationship in hemoglobin. X-axis: distance in Å. Y-axis: the number of atom pairs within the range of  $[x, x+0.02)$ .

In this study, we completely ignore any previous chemical knowledge such as chemical bonds. The only information we assume is the type of each atom and its coordinates. In figure 2 we plot the distribution of C-C distances within the hemoglobin protein (limiting distances to within the range  $R=5 \text{ \AA}$ ). In figure 2, the first two peaks centered at  $1.4 \text{ \AA}$  and  $1.54 \text{ \AA}$ , are well separated, the remaining peaks are not so clear-cut. The first two peaks represent the carbon-carbon double bond and single bond respectively. The peaks after these peaks cannot be separated very well though they very well may represent interesting patterns if we consider the curve at that region as the result of overlapping peaks. Though there are no direct chemical bonds between the atoms, however, these patterns may reflect some indirect relationship between the carbon atoms. Other two-atom relationships in hemoglobin have also been plotted (data not shown here) and similar results were obtained i.e. if there are direct chemical bonds between the two atoms, the relationship can be clearly seen as peaks and if there are no direct chemical bonds, there are still some patterns seen as overlapping peaks. The above evidence suggests that frequency can be used to identify and prune the structural patterns. Please note that the overlapping peaks are not pruned. We believe those overlapping peaks are found when other atoms are involved and represent an indirect connection (such as those induced by hydrogen bonds).

The first two pruning strategies can significantly prune the search space. For example for lysozyme the total number of 3-atomsets is 172,227,220. The total number of evaluated 3-structures is 25475 (a result of the first pruning technique and the fact that several atomsets have the same structure) and the number of important 3-structures (as determined by a frequency parameter of 100) is 3008. We will next show that a further optimization is to use these frequent k-atomsets to build and evaluate higher order structures further reducing the total number of evaluated structures.

### ***2.3. Building potentially frequent k-atom structures from frequent k-1-atom structures***

As mentioned earlier each k-atomset induces a structure, we refer to this structure as a k-atom structure.

*Definition 1A:* In a given k-atom structure we refer to extremal atoms as the two atoms that are furthest away from each other within the structure. We define extremal substructures as follows:

*Definition 1B:* For a given k-atom structure we define extremal k-1-atom substructures as those substructures that contain exactly one of the extremal atoms.

By the above definition each structure has exactly two extremal substructures. The above two definitions now enable us to state the lemma that will help prune the number of potentially frequent patterns (candidates) that will need to be evaluated.

*Lemma 1:* Any frequently occurring k-atom structure has at least two k-1-atom substructures that are frequent and that satisfy the input range criteria ( $R$ )

*Corollary 1:* For any frequently occurring k-atom structure both of its extremal substructures are frequent and satisfy the input range criteria ( $R$ ).

#### **Proof Sketch:**

It is trivial to show that all substructures of a given frequent substructure must be frequent. However not all of these substructures will satisfy the range criteria. A simple example that

shows why this is so involves a line of points each separated by a distance corresponding to the range ( $R$ ). If you take out any of the points except the two end points the resulting substructure will not satisfy the range criteria, because eliminating such a point breaks the linkage. The rest of the proof is based on the fact that eliminating either of the extremal points cannot possibly break the linkage. Now assume we are given the set  $S$  of frequently occurring  $k-1$ -atom structures. By the above lemma and its corollary the set  $C$  of potentially frequent  $k$ -atom structures is limited to those candidates whose extremal sub-structures are in  $S$ . Basically, if the extremal sub-structures were not in  $S$  then the potential candidate can never be frequent.

### 3. Representing structures

While there are several ways to represent structural information using geometrical descriptors (such as local moments) we rely on a simple mechanism based on the spatial relationship between each pair of atoms in the atomset. The relationship between one atom and another atom is represented using a short integer, which we refer to as mining bonds. The mining bond contains two parts. The first part is used to represent atom-pair information. The second part is used to represent the distance information between the two atoms. Currently we use 8 bits for each component, which allows us to represent only 256 distance relations. From the experimental results in Section 5, we will see that this is enough for proteins. Note that one can easily limit the atom-pair information to 6 bits allowing us to represent up to 1024 distance relations. The above representation is more than enough to represent the atom relationship in protein if reasonable range ( $R$ ) and resolution ( $res$ ) are chosen. The resolution ( $res$ ) is another important parameter in this study. The distance relation between two atoms is represented by dividing the distance by the resolution ( $res$ ) and then taking the integral part. In doing so, the data is essentially quantized into bins of size  $res$ .

#### 3.1 Feature array to represent structure

Each atomset has certain structural information. The structure information can be easily expressed as the set of all the mining bonds involving each pair of atoms within the atomset. The set of mining bonds (an integer array) is called a feature array in this study. We sort the mining bonds in a specific manner to ensure that a given atomset has only one unique feature array. The feature array ensures that the structures of different atomsets can be compared easily. However, there is a possibility that two atomsets with different structures have the same or similar feature array. It will happen when the structure is highly symmetric. But the chance is very little in protein structure. And as the range increases, each atom will have more and more adjacent atoms to identify itself, and the chance of mismatching will become less and less.

#### 3.2 Global Structures from Local Structure Linkage

An obvious limitation of using range ( $R$ ) is that one may not be able to construct more complicated structures within a protein. However, our approach has a mechanism in place that ensures that two atoms out of the range can be linked together by intermediate atoms in between them. One can also make the distinction between weak and strong global structures. In strong global structures, to make sure 2 atoms out of range are correctly linked together by intermediate atoms; we enforce each atom to have at least 3 adjacent atoms if the atomset has more than 4 atoms. This restriction is not imposed in the case of weak global structures. In our case, let us suppose atom  $A$  is adjacent to  $B$ ,  $C$ ,  $D$  and another atom  $E$  which is adjacent to  $B$ ,  $C$ ,  $D$  but not  $A$ . The resulting structure composed of  $A$ ,  $B$ ,  $C$ ,  $D$ , and  $E$  would be a *strong* global structure linked together by local structures that satisfy the range criteria. The rest of this study focuses on

the discovery of local structures and *strong* global structures. We plan to evaluate the utility of weak global structures (linked by less than 3 atoms) as part of future work.

## 4. Algorithmic Details

Before we detail the algorithm we first highlight how one can compare structures and determine whether two structures are similar or not.

### 4.1 Comparing Structures

We use geometric hashing to compare structures (atomsets) efficiently. The essential idea is to hash each atomset, based on the pair-wise distance relationships of atoms within the atomset, in such a way such that all the atomsets that conform to the same structure are hashed to the same location in the hash table. Such techniques have been used in the context of discovering motifs in molecules [W97]. Here, we extend the idea presented in [W97] and resort to recursive hashing to reduce memory utilization. During recursive hashing, the mining bonds in the feature array are hashed one by one. At each hashing step, the atomsets that are hashed to the same position are hashed again to next level until all mining bonds in the feature array are finished. Associated with each hashing step, there is a pruning step, during which infrequent structures (i.e. similarly structured atomsets which do not occur frequently enough) are removed. Recursive hashing ensures an efficient comparison of atomset structures and makes full use of pruning process described in Section 2.2. The hashing mechanism proposed may be improved with the utilization of more complex structure descriptors (local moments). This is an area of future investigation.

### 4.2 Fuzzy matching to deal with noisy data

As described earlier, the distances between atoms are discretized into bins of size *res*. It is highly possible that two distances varying very little are binned differently. To deal with this problem we implemented a fuzzy matching algorithm. In the fuzzy matching, each mining bond is hashed into its own position and its neighbor positions. Though this significantly increases computation time, empirical evidence suggests that fuzzy hashing is essential to deal with noisy data such as protein structure coordinates. Basically, so long as the resolution is carefully selected to cover the maximal range of error, no structural patterns will be lost.

#### *Algorithm for mining structures:*

1. All the atoms in the protein are considered with their adjacent atoms to create candidate 2-atomsets. Each atomset is represented by a **feature array**, which is composed by the mining bonds of each atom.
2. The structures of k-candidate sets (starting from 2-candidate sets) are compared using their feature array. Here, **recursive fuzzy hashing**, is employed. After hashing, structures (and the atomsets they represent) with low occurrence are pruned. Low occurrence is determined by a user-defined notion of **minimal support** (minSup). The minimal support value may be varied for different tasks.
3. After the pruning the infrequent atomsets, the remaining k-candidate sets are now the **k-atomsets**. Those of these that hash to the same final location are considered to represent the same basic structure.
4. (k+1)-candidatesets can be generated by combining k-atomsets that satisfy the criteria described in section 2.3. Currently we have not completely implemented this step and use a weaker approach. Essentially we compute k+1 candidates by combining k-atomsets with 2-atomsets.
5. Repeat step 2-4 until there are no more candidates.

## 5. Experimental Results:

We use lysozyme (PDB ID=193L) as the model molecule to demonstrate our results. Lysozyme is a protein with 129 amino acids. The PDB file for lysozyme (193L) has 1012 atoms. It is known to contain  $\alpha$ -helix,  $\beta$ -sheet and also turns and random structures (see Figure 3:  $\alpha$ -helix structure is represented with red tubes,  $\beta$ -sheet structure is represented with blue arrow).

### *Rediscovering the Protein Backbone Structure*

We first tried to mine the frequent atomsets, using our algorithm with minSup=100, range=4.5. Basically, this is to find the backbone peptide structures since it has 129 amino acid residues and it should have 128 peptide bonds. Considering the possibility of presence of cis-peptide bond, minSup=100 is a safe number to capture this information. We tried different resolutions: 0.04Å, 0.05Å and 0.06Å. The atomsets with maximal number of atoms have 5 atoms. At res=0.04, 125 frequent 5-atomsets are found. At res=0.05, 127 frequent 5-atomsets are found. At res=0.06, all 128 peptide bonds are found. The missing peptide bond at res=0.05 is a trans peptide bond. The missing atomsets at smaller resolution indicate that the coordinates are noisy and that the resolution did not capture the maximal error (noise) in the data (see Section 4.1). We consider the results obtained with res=0.06 for further discussion. Under this setup the frequent atomsets discovered are: 3415 3-atomsets, 873 4-atomsets and 128 5-atomsets. Interestingly enough all the 5-atomsets found have one structure. A typical 5-atomset is shown below

Atom	Id	type	Amino Acid		x	y	z
ATOM	4	O	LYS	1	2.390	12.719	9.904
ATOM	13	N	VAL	2	2.467	12.329	7.688
ATOM	14	CA	VAL	2	2.417	13.741	7.336
ATOM	2	CA	LYS	1	2.429	10.424	9.199
ATOM	3	C	LYS	1	2.415	11.932	8.954

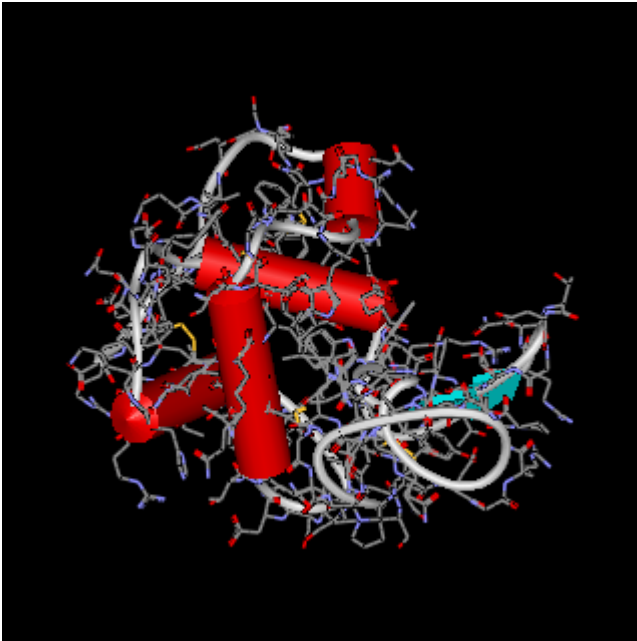
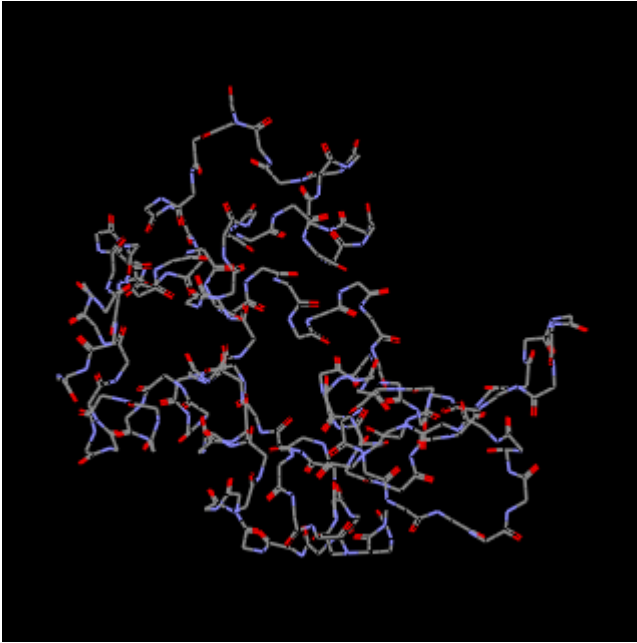


Figure 3. The structure of lysozyme(193L).  $\alpha$ -helix is represented with red tubes.  $\beta$ -sheet structure is represented with blue arrow.

It is composed of the backbone oxygen and carbon,  $\alpha$ -carbon of residue  $i$  and backbone nitrogen and  $\alpha$ -carbon of residue  $i+1$  (as shown in figure 4). On displaying all the 5-atomsets we can easily see that it forms the complete backbone of the protein (shown in figure 5). This result serves to validate the fact that the algorithm presented can be used to completely specify the backbone structure of a protein.

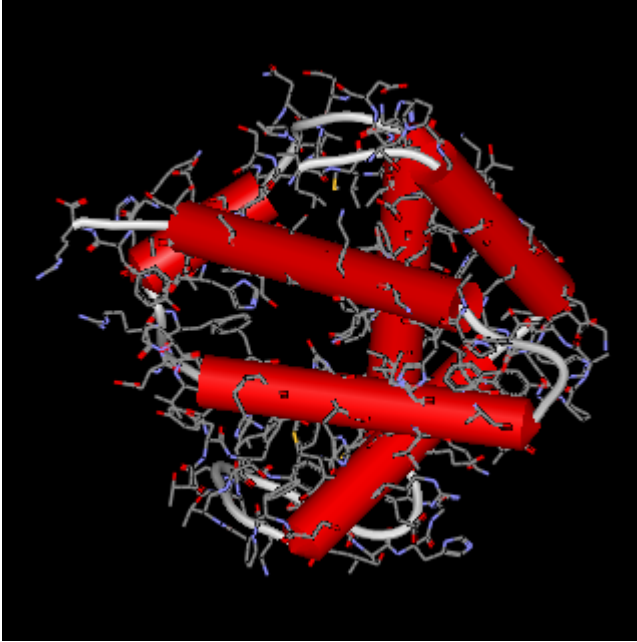
**Figure 4: Peptide bond structure found by the algorithm**



**Figure 5: The frequent structure forms the backbone.**

### *Discovering Secondary Structures*

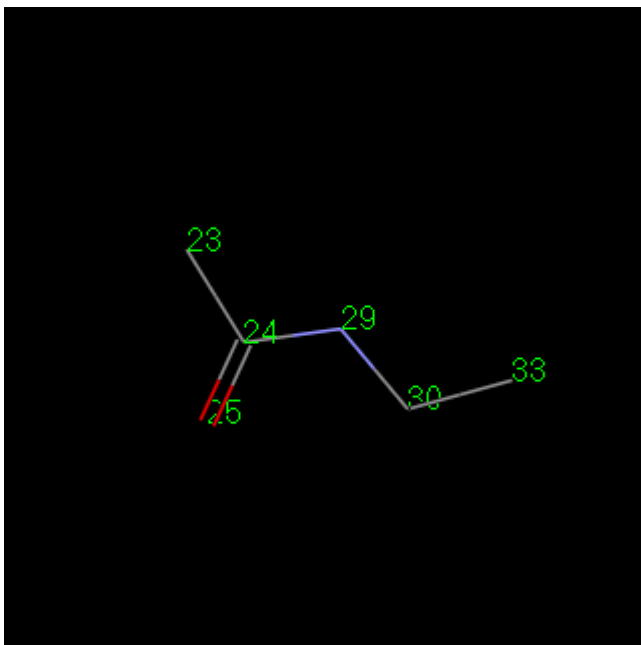
We further evaluated the proposed algorithm on the first subunit (141 amino acids and 1069 atoms) of a hemoglobin protein (PDB ID=1ZB0). Hemoglobin is rich in  $\alpha$ -helix as shown in figure 6 and we expected our algorithm to discover the important structures that make up a  $\alpha$ -helix within this protein.



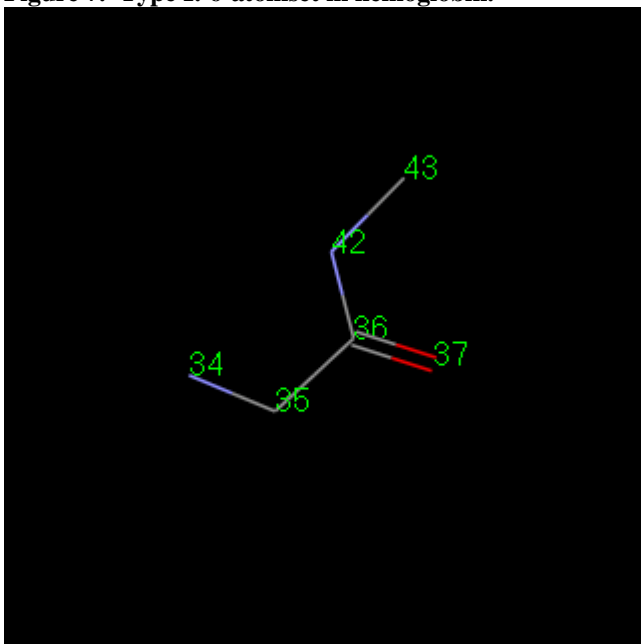
**Figure 6: the structure of hemoglobin first sub unit**

We set minSup=80, range=4.5, resolution=0.06 to test what kind of structures can be found. We found 7147 3-atomsets, 3332 4-atomsets, 1841 5-atomsets and 212 6-atomsets. The atomsets with six atoms were essentially found to represent two unique structures and the following are examples of each:

Atom Id	type	Amino Acid	x	y	z
ATOM 25	O	PRO 4	30.987	31.959	42.393
ATOM 29	N	ALA 5	30.207	31.668	44.503
ATOM 33	CB	ALA 5	31.024	30.092	46.215
ATOM 30	CA	ALA 5	31.077	30.521	44.760
ATOM 23	CA	PRO 4	29.213	33.432	43.229
ATOM 24	C	PRO 4	30.224	32.297	43.317
ATOM 37	O	ASP 6	30.062	27.547	40.612
ATOM 34	N	ASP 6	29.352	29.193	43.621
ATOM 42	N	LYS 7	29.422	29.712	40.844
ATOM 43	CA	LYS 7	29.870	30.083	39.484
ATOM 35	CA	ASP 6	28.961	28.160	42.651
ATOM 36	C	ASP 6	29.573	28.470	41.280



**Figure 7. Type I: 6-atomset in hemoglobin.**

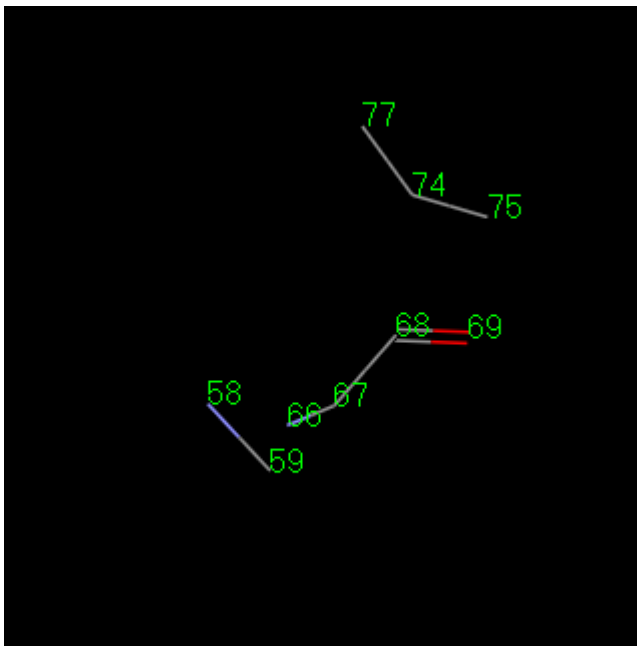


**Figure 8. Type II: 6-atomset in hemoglobin.**

The first structure (see figure 7) was found to be more frequent than the second (see figure 8). This result would seem to indicate that in a  $\alpha$ -helix rich protein such as hemoglobin, the  $\beta$ -carbon of residue  $i+1$  is more likely to form certain conserved structure. It is the most important structure after backbone peptide. The other frequently repeated structure includes the backbone nitrogen of residue  $i$  as compared with the conserved 5 atoms structure of peptide bond.

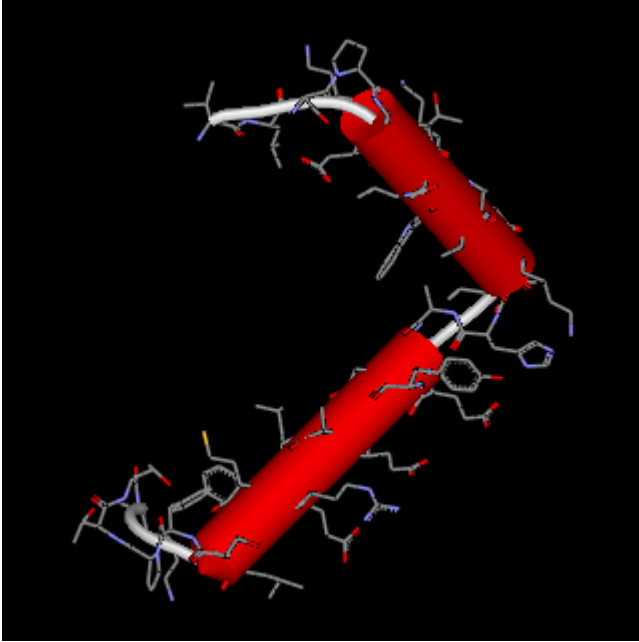
We also try the algorithm on the first 2  $\alpha$ -helices of the first subunit of hemoglobin (299 atoms, 40 amino acid residues) with a lower minSup. The parameters are minSup=15, range=4.5, resolution=0.07. The frequent atomsets with maximal number of atoms contains 9 atoms and has one unique structure. The following is one example.

Atom	Id	type	Amino Acid	x	y	z
ATOM	69	O	VAL A 10	33.089	25.103	35.737
ATOM	66	N	VAL A 10	32.020	26.018	38.984
ATOM	58	N	ASN A 9	33.155	27.625	40.993
ATOM	77	CB	LYS A 11	34.214	29.208	35.850
ATOM	75	C	LYS A 11	35.320	26.968	35.777
ATOM	74	CA	LYS A 11	34.019	27.736	35.693
ATOM	67	CA	VAL A 10	31.635	25.529	37.630
ATOM	68	C	VAL A 10	32.693	25.908	36.591
ATOM	59	CA	ASN A 9	33.383	26.158	40.955



**Figure 9: the 9-atomset found in the first helices of first unit of hemoglobin.**

This structure involves atoms from 3 different residues. And the distances between some atom pairs are greater than the range. It indicates that local structure linkage scheme works. The atoms that form the 9-atomset are mostly backbone atoms and  $\beta$ -carbon atoms in the helical region. Based on a comparison between figures 10 and 11, it is reasonable to conclude that it is one of the basic structural units that make up a  $\alpha$ -helix.



**Figure 10.** The structure of the first two helices of the first subunit of hemoglobin.  $\alpha$ -helices are represented by red tubes.



**Figure 11.** The frequent atoms in 9-atomset are mainly found in the helices. There is one exception the very end (lower left corner in the figures). It might be the structure is too small to be considered as part of a  $\alpha$ -helix.

### ***6. Discussion and Conclusions:***

The results, when put into the perspective that this is clearly work in progress, are very encouraging. First, the approach presented seems robust to noise since the important structures

(e.g. backbone peptide bonds, some secondary structures) are found even though the protein structure coordinates are noisy. The fuzzy matching mechanism presented, while inevitably expensive, is able to make the algorithm that much more robust. Another factor that plays a role is resolution (res), which reduces the influence of noise. Second, the completeness of the algorithm enables it to identify all 128 peptide bonds. Third the approach is flexible enough to find even larger structures (involving 9 and 10 atoms) that form the basis of some of the more important secondary structures (such as  $\alpha$ -helices). The approach is able to discover such large structures even if the maximum distance between any two atoms within the structure exceeds the parameterized range value(R). This is testament to the fact that the local structure linkage method presented works quite well.

The critical factors that influence our results are the range and resolution. The range determines the scope of atoms that should be considered when constructing local structure. A small range will not help discover bigger structures unless the bigger structure can be constructed by smaller structures linked together (e.g. the 9-atomset structures that likely form one of the basic structural units of an  $\alpha$ -helix structure). Too short a range will certainly damage the construction of local structures and their linkability. A long range will result in too many redundant or trivial structures being found.

The resolution parameter plays three roles. First it reduces the memory requirements of the algorithm, which is crucial given the number of structures that have to be evaluated even for an average sized protein. Second, it also minimizes the effect of noise present in protein structure coordinates. Of course the higher the physical resolution (smaller in value) the more defined the discovered structure will be. So long as the resolution value can cover the error range of the noisy data, the completeness is guaranteed. If the resolution value is too large, the discovered structures may not be well defined and there is the danger that not so similar structures can be considered to be similar by fuzzy comparison. Third, as we have mentioned in Section 4.1, the resolution parameter should be chosen so as to cover the maximal range of error (noise). This is to ensure that the fuzzy matching scheme will not lose any structural patterns.

Another parameter that has an important role to play is minSup. Proteins as we have noted have different levels of repeated structures. The number of peptide bond structures is just a little less than the number of amino acid residues in the molecule. The number of secondary structures ( $\alpha$ -helix,  $\beta$ -sheet etc) may vary from protein to protein. Empirical evidence shows that by varying the minSup, different levels of important patterns can be found. Clearly by varying the above three parameters (range, resolution and minSup), the resulting patterns may be quite different. The process of discovery [HK01] is likely to involve several such interactions on the part of the domain expert (biochemist). Since this is the case, a question that can be asked is whether we can re-use some of the information extracted to expedite future interactions [PZOD99].

Though the algorithm presented is designed for protein molecules, it can probably be generalized to mine structures in other domains (such as defect structures within a bulk lattice in molecular dynamics simulations) as well. It can be easily extended to mine common structures among different kinds of molecules. The discovered common structures can be used to classify molecules [W97]. Another approach would be mine the common structure within a family of molecules and get the feature array set. When new molecules are synthesized, the feature array of the new molecule can be compared to the feature array set of the family. Prediction of function can be made if the new molecule has the feature array of certain families of molecules.

In order to make this method capable of predicting structure from primary structure, it is also very important to develop algorithms that can do the classification efficiently.

Though the pattern finding part of this algorithm has been established, it still lacks of an effective way of identifying which structures are important and which are not. As part of future work we plan to turn to interestingness analysis [MP96], an area that has received a lot of interest from the data mining community recently, to see how one can distinguish between the structural patterns found. Most of the current techniques use statistical methods to identify and differentiate amongst such structures. In our context, in addition to statistical information, scientific (domain-specific) knowledge may be needed to classify which structures are more important than others.

## References

- [AS94] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. VLDB'94.
- [DTK98] L. Dehaspe, H.Toivonen, R. King, Finding frequent substructures in chemical compounds, In KDD98.
- [DCH95] S. Djoko, D. Cook, and L. Holder, Analyzing the benefits of domain knowledge in substructure discovery. In KDD95.
- [HK01] J. Han, M. Kamber, Data Mining Concepts and techniques, Academic Press, 2001.
- [JD88] A.K. Jain and R.C. Dubes. In Algorithms for Clustering Data. Prentice Hall, 1988.
- [M96] Matheus et al, " Selecting and Reporting What is Interesting", In Fayyad et al. editors, Advances in Knowledge Discovery and Data Mining, AAAI Press, 1996.
- [P99] S.Parthasarathy, M.Zaki, M. Ogihara, S. Dwarkadas, Incremental and Interactive Sequence Mining, In ACM CIKM 1999.
- [Q96] J. R. Quinlan. Induction of decision trees. Machine Learning, 5(1):71-100, 1996.
- [SA96] R. Srikant and R. Agrawal. Mining sequential patterns: Generalizations and performance improvements. In 5th EDBT, 1996.
- [W97] X. Wang, et al. Automated discovery of active motifs in three dimensional molecules. In KDD97.