International Journal of Computational Intelligence and Informatics, Vol. 3: No. 3, October - December 2013

Protein Sequence Motif Detection using Novel Rough Granular Computing Model

E Elayaraja
Department of Computer Science
Periyar University
Salem, Tamilnadu, India
elayarajaphd.e@gmail.com

M Chitralegha
Department of Computer Science
Periyar University
Salem, Tamilnadu, India
chitra_legha04@yahoo.co.in

K Thangavel
Department of Computer Science
Periyar University
Salem, Tamilnadu, India
drktvelu@yahoo.com

T Chandrasekhar
Department of Computer Science
Periyar University
Salem, Tamilnadu, India
ch_ansekh80@rediffmail.com

***Abstract*-Protein sequence motifs information is essential for the analysis of biologically significant regions. Discovering sequence motifs is a key task to realize the connection of sequences with their structures. Protein sequence motifs have the potential to determine the function and activities of the proteins. Many algorithms or techniques are used to determine motifs which require a predefined fixed window size. Our input dataset is extremely large as a result, an efficient technique is demanded. So we apply three different granular computing models to find protein motif information which transcend protein family boundaries. The constructed segments from 3000 protein sequences are divided into granules using Rough K-Means and then K-Means has been applied on each granule. The highly structured clusters are further considered to find motif patterns. This approach is compared with Adaptive Fuzzy Granular model. The proposed Rough Granular computing model generates more number of highly structured motif patterns.**

Keywords-Protein Sequence Motifs, DBI, HSSP-BLOSUM62, Granular Computing, K-Means, Adaptive Fuzzy C-Means, Rough K-Means.

I. INTRODUCTION

The relationship between protein structure and its sequence is one of the most vital roles of current bioinformatics research. The term biological sequence motifs obtained from functionally conserved sequence regions may be used to predict any subsequent reoccurrence of structural or functional areas on other proteins. These functional and structural areas may include enzyme-binding sites, DNA or RNA binding sites, prosthetic group attachment sites, or regions involved in binding other small molecules.

PROSITE [1], PRINTS [2], and BLOCKS [3] are three popular databases for sequence motifs. There are some commonly used softwares for protein sequence motif discover including MEME [16], Gibbs Sampling [15, 17], Block Maker [25] and some of the latest algorithms include MITRA [14], and Gemoda [26]. Several protein sequences are required to be input by the user while using these tools. Since the size of input dataset is limited and discovered motifs are based on these input sequences, the obtained information from above methods may carry little information about conserved sequence regions, which transcend protein families.

In this research, protein sequences are converted into segments using sliding window concepts and patterns are extracted from the selected segments. These sliding sequence segments are separated into different groups with granular computing models that utilized Fuzzy C-Means, Adaptive Fuzzy C-Means and Rough K-Means clustering algorithms to divide the whole data space into several smaller subsets and then apply K-Means and Rough K-Means algorithm to each subset to discover relevant information. Finally, we merge the information generated by all granules and obtain the final sequence motif information. Three evaluation methods are applied in this study such as structural similarity, DBI measure, and HSSP-BLOSUM62 evaluation method. The novelty of the study is applying Rough K-Means to have the granules which will include more segments.

The rest of the paper is organized as follows. Section 2 presents related work in this area of research. In section 3, the description of granular computing techniques and clustering algorithms are explained. Experimental

setup is explained in section 4. In section 5, experimental results are explained. Section 6 concludes the paper with directions for further enhancement.

II. Related Work

K-Means clustering algorithm with random initial centroids is utilized by Han et al. [7] to find recurring protein sequence motifs across the boundaries of a protein family. To overcome the inherent problem of K-Means clustering algorithm, Wei et al. proposed an improved K-Means clustering algorithm to obtain initial centroid locations more wisely [6,12] and the results published by Wei et al. have been improved in their experiment. Fast computation is always one of the advantages for K-Means, other clustering methods with higher time and space costs may not be suitable for this task.

In order to overcome the high computational cost caused by a huge input dataset, Bernard Chen et al. proposed a granular computing model work called FIK model [11, 12] which utilizes a Fuzzy C-Means clustering algorithm to divide the whole data space into several smaller subsets and then applies a standard improved K-Means algorithm to each subset to discover relevant information. In FGK model [11, 12] Bernard Chen et al. develop a new greedy K-Means algorithm to further improve secondary structural similarity sequence motifs. In this paper, our goal is to produce more clusters with good structural similarity.

III. Granular Computing Techniques

A. Fuzzy Granular Model

This model works by using Fuzzy C-Means (FCM) for building a set of information granules and then applying K-Means and Rough K-Means clustering algorithms to obtain the final information. The FGM process is given in Fig. 1 and Fig. 2.

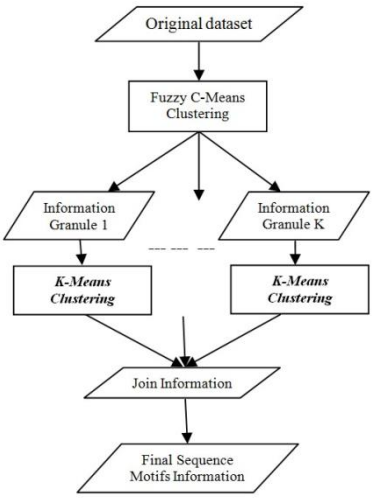


Figure 1. Sketch of FGM using K-Means Computing Model

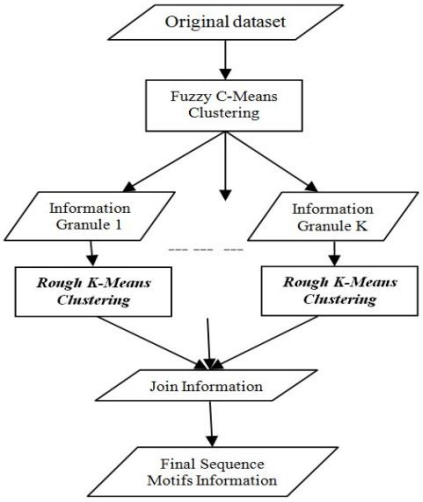


Figure 2. Sketch of FGM using Rough K-Means Computing Model

1) Fuzzy C-Means

Fuzzy C-Means (FCM) is a clustering algorithm which allows one segment of data is belongs to one or more clusters. This algorithm is to minimize the following objective function [12].

$$J_m = \sum_{i=1}^{N}\sum_{j=1}^{C} u_{ij}^{m} \left\|x_i - c_j\right\|^2, 1 \le m < \infty \quad (1)$$

where m, the fuzzification factor, is any real number greater than 1, u_{ij} is the degree of membership of x_i in the cluster j, x is the i^{th} of d-dimensional measured data, c is the d dimension center of the cluster, and ‖*‖ is any norm expressing the similarity between any measured data and the center. Fuzzy partitioning is carried out through an iterative optimization of the objective function shown above, with the update of membership u_{ij} and the cluster centers c_j by:

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^{m} \cdot x_i}{\sum_{i=1}^{N} u_{ij}^{m}} \quad (2)$$

Where

$$u_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}} \quad (3)$$

This iteration will stop when $\max_{ij} \{ |U^{(k+1)} - U^{(k)}| \} < \delta$ where δ is a termination criterion between 0 and 1, whereas k is the iteration step. This procedure converges to a local minimum or a saddle point of J_m.

The Fuzzy C-Means Clustering algorithm is described as following:

---

1. Initialize membership function matrix U = [u_{ij}], and U (0).
2. at k step: Calculate the centroid point by the equation (2)
3. Update $U^{(k)}$ and $U^{(k+1)}$ by using equation (3).
4. if $|U^{(k+1)} - U^{k}| < \mathcal{E}$ then stop; otherwise return to step 2.

B. Adaptive Fuzzy Granular Model

A set of information granules is built using the Adaptive Fuzzy Granular Model (AFGM) and then applying K-Means and Rough K-Means Clustering algorithms to obtain the final information. The AFGM process is given below in Fig. 3 and Fig. 4 [23].

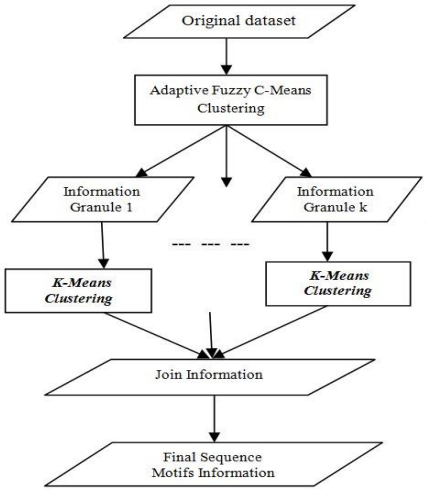


Figure 3. Sketch of AFGM using K-Means Computing Model

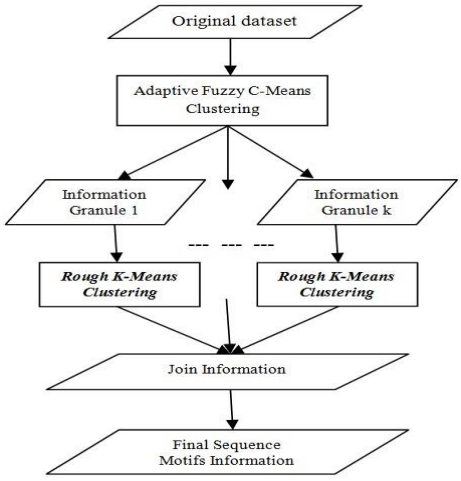


Figure 4. Sketch of AFGM using Rough K-Means Computing Model

1) Adaptive Fuzzy C-Means

Many of the behavioural problems with standard Fuzzy C-Means algorithm are eliminated when we relax probabilistic constraint imposed on membership function. Further Krishnapuram R and Keller JM [19, 23] modified the approach for calculating membership values. Equation (4) shows membership calculation.

$$\sum_{j=1}^{k} \sum_{i=1}^{n} \mu_{j\,(x_i)=n} \quad (4)$$

Here,

$\mu_j (x_i)$ is the membership of x_i in j^{th} cluster

k is the specified number of clusters

n is the number of data points

In Adaptive Fuzzy C-Means (AFCM), the total membership quantifiers for all sample points are equal to n. This flexible approach leads to clustering optimization problem, provides a way to improve cluster robustness. It is in this sense the algorithm is adaptive; that is membership is based on sample size rather than fixed to upper

limit as one in Fuzzy C-Means clustering. The membership values in this method are calculated using Equation (5).

$$\mu_j(x_i) = \frac{n\left(\frac{1}{d_{ji}}\right)^{\frac{1}{m-1}}}{\sum_{k=1}^{p}\sum_{z=1}^{n}\left(\frac{1}{d_{kz}}\right)^{\frac{1}{m-1}}} \quad (5)$$

The Adaptive fuzzy clustering algorithm is efficient in handling data with outlier points. It gives very low membership values for outliers since the sum of distances of points in all the clusters involves in membership calculation.

C. Rough Granular Model

A set of information granules is built using the Rough Granular Model (RGM) and then applying K-Means and Rough K-Means Clustering algorithms to obtain the final information. The RGM process is given below in Fig. 5 and Fig. 6 [23].

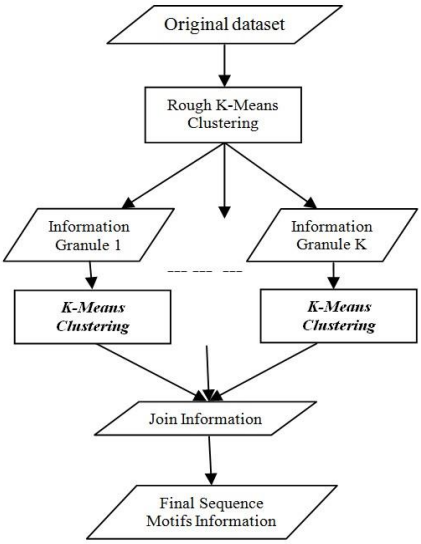


Figure 5. Sketch of AFGM using K-Means Computing Model

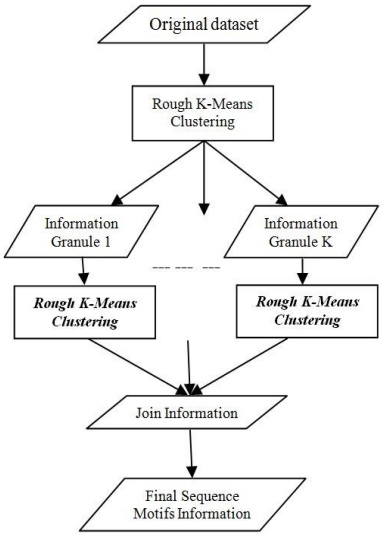


Figure 6. Sketch of AFGM using Rough K-Means Computing Model

D. Rough Clustering

In rough clustering each cluster has two approximations, a lower and an upper approximation. The lower approximation is a subset of the upper approximation. The members of the lower approximation belong certainly to the cluster; therefore they cannot belong to any other cluster. The data objects in an upper approximation may belong to the cluster. Since their membership is uncertain they must be a member of an upper approximation of at least another cluster.

1) Rough Properties of the Cluster Algorithm

Property 1: a data object can be a member of one lower approximation at most.

Property 2: a data object that is a member of the lower approximation of a cluster is also member of the upper approximation of the same cluster.

Property 3: a data object that does not belong to any lower approximation is member of at least two upper approximations [24].

The Rough K-Means algorithm provides a rough set theoretic flavour to the conventional K-Means algorithm to deal with uncertainty involved in cluster analysis. The Rough K-Means algorithm [8, 9] described as follows:

1. Select initial clusters of n objects into K clusters.
2. Assign each object to the Lower bound (L(x)) or upper bound (U(x)) of cluster/ clusters respectively as: For

each object v, let d (v,xi) be the distance between itself and the centroid of cluster xi. The difference between d (v,xi) / d(v,xj), 1≤ i, j ≤ k is used to determine the membership of v as follows:

- If d (v,xi) / d(v,xj) ≤ thershold, then v ∈U(xi) & v ∈ U(xj). Furthermore, v will not be a part of any lower bound.
- Otherwise, v∈L(xi),such that d(v,xi) is the minimum for 1≤ i ≤ k. In addition, v∈U(xi).

3. For each cluster xi re-compute center according to the following equations the weighted combination of the data points in its lower_bound and upper_bound.

$$x_i = \begin{cases} w_{lower} \times \frac{\sum_{v \in L(x)} v_j}{|L(x)|} + w_{upper} \times \frac{\sum_{v \in U(x)-L(x)} v_j}{|U(x)-L(x)|} & \text{if } |U(x)-L(x) \neq \phi| \\ w_{lower} \times \frac{\sum_{v \in L(x)} v_j}{|L(x)|} & \text{otherwise} \end{cases}$$

where 1≤ j ≤ k. The parameters w_{lower} and w_{upper} correspond to the relative importance of lower and upper bounds. If convergence criterion is met, i.e. cluster centers are same to those in previous iteration, then stop; else go to step2.

E. K-Means Clustering Algorithm

Among all clustering algorithms, K-Means clustering algorithm has the advantages of easy interpretation and implementation, high scalability, and low computation complexity. The K-Means clustering take the user input parameter K, and partitions a set of n objects into K clusters then iteratively updates the centers until no reassignment of patterns to new cluster centers occurs. In every step, each sample is allocated to its closest cluster center and cluster centers are reevaluated based on current cluster memberships [20].

IV. EXPERIMENTAL SETUP

A. Data Set

The dataset obtained from Protein Sequence Culling Server (PISCES) includes 4946 protein sequences [10]. In this work, we have considered 3000 protein sequences to extract sequence motifs that transcend in protein sequences. The threshold for percentage identity cut-off is set as less than or equal to 25%, resolution cut-off is 0.0 to 2.2, R-factor cut-off is 1.0 and length of each sequence varies from 40 to 10,000. Homology Derived Secondary Structure of Proteins (HSSP) frequency profiles is used to represent each segment [4, 5]. The sliding windows with ten successive residues are generated from protein sequences. Each window represents one sequence segment of ten continuous positions. Around 6, 60,364 sequence segments are generated by sliding window method, from 3000 protein sequences. Each sequence segment is represented by 10 X 20 matrix, where ten rows represent each position of sliding window and 20 columns represent 20 amino acids. Fig. 7 shows sliding window technique.

Thus by applying the sliding window technique we can generate n number of sequence segments (10*20 matrices).

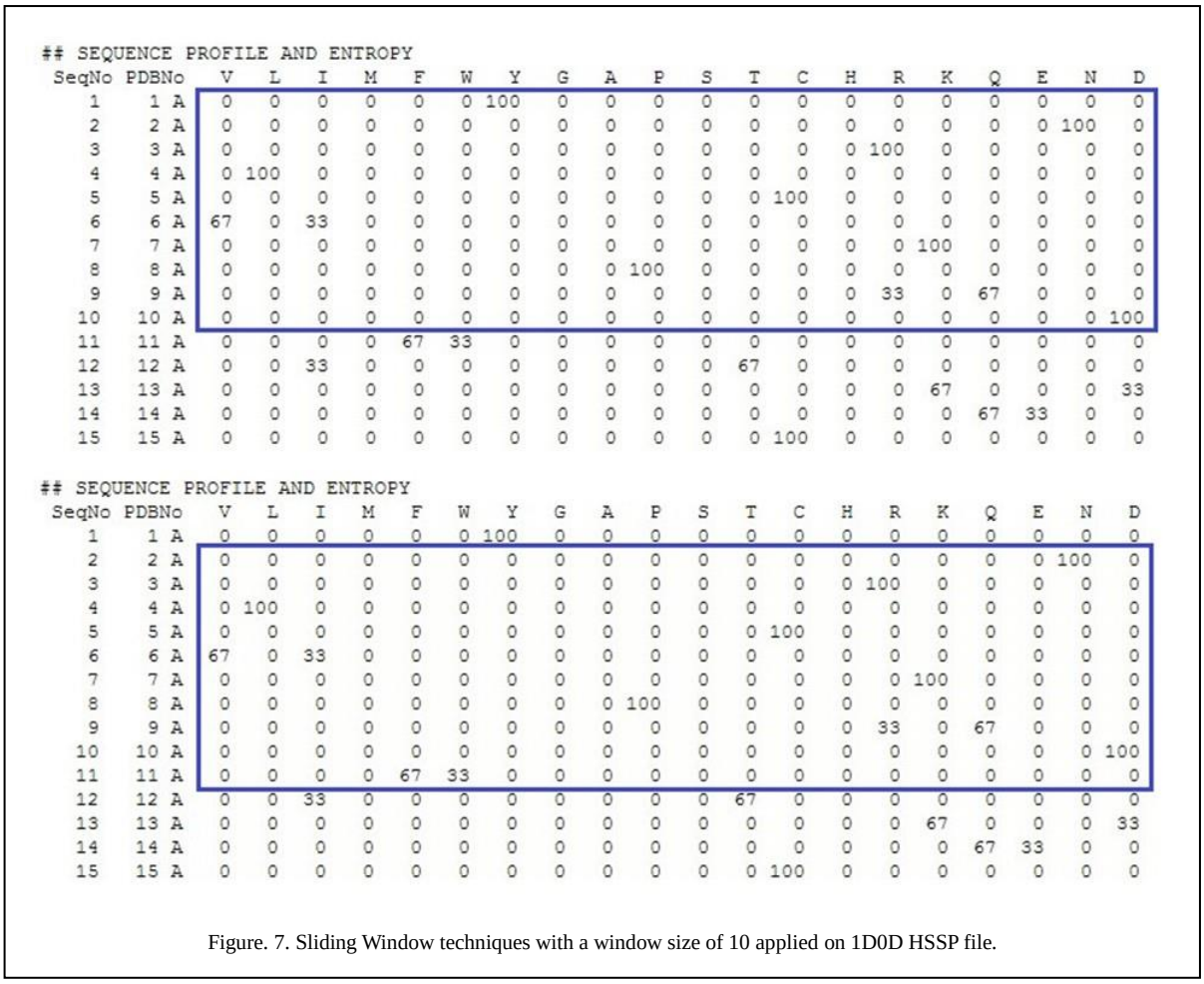
B. Structural similarity measure

A cluster's average structure is calculated using the following formula:

$$\frac{\sum_{i=1}^{WS} \max(P_{i,H}, P_{i,E}, P_{i,C})}{WS} \quad (6)$$

Where ws is the window size and $(P_{i,H})$shows the frequency of occurrence of helix among the segments for the cluster in position i. $(P_{i,E})$ and $(P_{i,C})$ are defined in a similar way. If the structural homology for a cluster exceeds 70%, the cluster can be considered structurally identical [12]. If the strucural homology for the cluster exceeds 60% and is below 70%, the cluster can be considered weakly structurally homologous.

Dictionary of Secondary Structure Proteins (DSSP) assigns secondary structure to eight different classes [21]. These eight structural classes can be reduced to three using reduction method as follows: H, G and I to H (Helices); B and E to E (Sheets); all others to C (Coils) [22].



SEQUENCE PROFILE AND ENTROPY

SeqNo	PDBNo	V	L	I	M	F	W	Y	G	A	P	S	T	C	H	R	K	Q	E	N	D
1	1 A	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0
2	2 A	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0
3	3 A	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0
4	4 A	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	5 A	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0
6	6 A	67	0	33	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	7 A	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0
8	8 A	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0
9	9 A	0	0	0	0	0	0	0	0	0	0	0	0	0	0	33	0	67	0	0	0
10	10 A	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
11	11 A	0	0	0	0	67	33	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	12 A	0	0	33	0	0	0	0	0	0	0	0	67	0	0	0	0	0	0	0	0
13	13 A	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	67	0	0	0	33
14	14 A	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	67	33	0	0
15	15 A	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0

SEQUENCE PROFILE AND ENTROPY

SeqNo	PDBNo	V	L	I	M	F	W	Y	G	A	P	S	T	C	H	R	K	Q	E	N	D
1	1 A	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0
2	2 A	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0
3	3 A	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0
4	4 A	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	5 A	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0
6	6 A	67	0	33	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	7 A	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0
8	8 A	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0
9	9 A	0	0	0	0	0	0	0	0	0	0	0	0	0	0	33	0	67	0	0	0
10	10 A	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
11	11 A	0	0	0	0	67	33	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	12 A	0	0	33	0	0	0	0	0	0	0	0	67	0	0	0	0	0	0	0	0
13	13 A	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	67	0	0	0	33
14	14 A	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	67	33	0	0
15	15 A	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0

Figure. 7. Sliding Window techniques with a window size of 10 applied on 1D0D HSSP file.

C. Distance Measure

The city block metric is more suitable for this field of study since it will consider every position of the frequency profile equally. The city block metric is used for calculating the difference between a sequence segment and the centroid of a given sequence cluster. Han and Baker also chose the city block metric because of complications associated with the use of Euclidean metric for clustering algorithms [7]. The following formula is used to calculate the distance between two sequence segments:

$$\text{Distance} = \sum_{i=1}^{L}\sum_{j=1}^{N}|F_k(i,j) - F_c(i,j)| \quad (7)$$

where L is the window size and N is 20 which represent 20 different amino acids. F_k (i j) is the value of the matrix at row i and column j used to represent the sequence segment. F_c (i,j) is the value of the matrix at row i and column j used to represent the centroid of a give sequence cluster.

D. Davis-Bouldin Index (DBI) Measure

The DBI measure [11] is a function of the inter-cluster and intra-cluster distance. A good cluster result should reflect a relatively large inter-cluster distance and a relatively small intra-cluster distance. The DBI measure combines both distance information into one function, which is defined as follows:

$$DBI = \frac{1}{k}\sum_{p=1}^{k}\max_{p\neq q}\left\{\frac{d_{intra}(C_p)+d_{intra}(C_q)}{d_{inter}(C_p,C_q)}\right\}, where \quad (8)$$

$$d_{intra}(C_p) = \frac{\sum_{i=1}^{n_p}\|g_i - g_{pc}\|}{n_p} \; and$$

$$d_{inter}(C_p, C_q) = \|g_{pc} - g_{qc}\|$$

K is the total number of clusters, d_{intra} and d_{inter} denote the intra-cluster and inter-cluster distances respectively. n_p is the number of members in the cluster C_p. The intra-cluster distance defined as the average of all pair wise distances between the members in cluster P and cluster P's centroid g_{pc}. The inter-cluster distance of two clusters is computed by the distance between two clusters' centroids. The lower DBI value indicates the high quality of the cluster result.

E. HSSP-BLOSUM62 Measure

BLOSUM62 [5] (Fig. 8) is a scoring matrix based on known alignments of diverse Sequences.

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V	B	Z	X	*
A	4	-1	-2	-2	0	-1	-1	0	-2	-1	-1	-1	-1	-2	-1	1	0	-3	-2	0	-2	-1	0	-4
R	-1	5	0	-2	-3	1	0	-2	0	-3	-2	2	-1	-3	-2	-1	-1	-3	-2	-3	-1	0	-1	-4
N	-2	0	6	1	-3	0	0	0	1	-3	-3	0	-2	-3	-2	1	0	-4	-2	-3	3	0	-1	-4
D	-2	-2	1	6	-3	0	2	-1	-1	-3	-4	-1	-3	-3	-1	0	-1	-4	-3	-3	4	1	-1	-4
C	0	-3	-3	-3	9	-3	-4	-3	-3	-1	-1	-3	-1	-2	-3	-1	-1	-2	-2	-1	-3	-3	-2	-4
Q	-1	1	0	0	-3	5	2	-2	0	-3	-2	1	0	-3	-1	0	-1	-2	-1	-2	0	3	-1	-4
E	-1	0	0	2	-4	2	5	-2	0	-3	-3	1	-2	-3	-1	0	-1	-3	-2	-2	1	4	-1	-4
G	0	-2	0	-1	-3	-2	-2	6	-2	-4	-4	-2	-3	-3	-2	0	-2	-2	-3	-3	-1	-2	-1	-4
H	-2	0	1	-1	-3	0	0	-2	8	-3	-3	-1	-2	-1	-2	-1	-2	-2	2	-3	0	0	-1	-4
I	-1	-3	-3	-3	-1	-3	-3	-4	-3	4	2	-3	1	0	-3	-2	-1	-3	-1	3	-3	-3	-1	-4
L	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4	-2	2	0	-3	-2	-1	-2	-1	1	-4	-3	-1	-4
K	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5	-1	-3	-1	0	-1	-3	-2	-2	0	1	-1	-4
M	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5	0	-2	-1	-1	-1	-1	1	-3	-1	-1	-4
F	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6	-4	-2	-2	1	3	-1	-3	-3	-1	-4
P	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7	-1	-1	-4	-3	-2	-2	-1	-2	-4
S	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4	1	-3	-2	-2	0	0	0	-4
T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5	-2	-2	0	-1	-1	0	-4
W	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11	2	-3	-4	-3	-2	-4
Y	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	-1	-3	-2	-1	-4
V	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4	-3	-2	-1	-4
B	-2	-1	3	4	-3	0	1	-1	0	-3	-4	0	-3	-3	-2	0	-1	-4	-3	-3	4	1	-1	-4
Z	-1	0	0	1	-3	3	4	-2	0	-3	-3	1	-1	-3	-1	0	-1	-3	-2	-2	1	4	-1	-4
X	0	-1	-1	-1	-2	-1	-1	-1	-1	-1	-1	-1	-1	-1	-2	0	0	-2	-1	-1	-1	-1	-1	-4
*	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	1

Figure 8. BLOSUM62 Matrix.

By using this matrix, we may access the consistency of the amino acids appearing in the same position of the motif information generated by our method. Because different amino acids appearing in the same position should be close to each other, the corresponding value in the BLOSUM62 matrix will give a positive value. Hence, the measure is defined as the following [13].

If k=0: HSSP-BLOSUM62 measure = 0

Else If k=1:

If $HSSP_i > 10\%$: HSSP-BLOSUM62 measure = $BLOSUM62_{ii}$

If $8\% \leq HSSP_i < 10\%$: HSSP-BLOSUM62 measure = $\frac{1}{2} BLOSUM62_{ii}$

Else: HSSP-BLOSUM62 measure = $\dfrac{\sum_{i=1}^{k-1}\sum_{j=i+1}^{k} HSSP_i.HSSP_j.BLOSUM62_{ij}}{\sum_{i=1}^{k-1}\sum_{j=i+1}^{k} HSSP_i.HSSP_j}$

F. Parameter Setup

For FCM granular fuzzification factor is been set to 1.15 and number of clusters is equal to ten. In order to separate information granules from FCM results, the membership threshold is set to 18% [23]. The function that decides how many numbers of clusters should be in each information granule is given below:

$$C_k = \frac{n_k}{\sum_{i=1}^{m} n_i} \times Total\ number\ of\ clusters \quad (9)$$

where C_k denotes the number of clusters assigned to information granule k. n_k is the number of members belonging to information granule k. m is the number of clusters in Fuzzy C-Means. In this technique we are able to indentify 900 clusters.

For Adaptive Fuzzy C-Means, fuzzification factor is considered as 1.15 and membership threshold is set to 13% [23]. Number of clusters in each granule is been decided by the function given below:

$$C_k = \frac{n_k}{\sum_{i=1}^{m} n_i} \times Total\ number\ of\ clusters \quad (10)$$

where C_k denotes the number of clusters assigned to information granule k. n_k is the number of members belonging to information granule k. m is the number of clusters in Adaptive Fuzzy C-Means. In this technique we are able to indentify 901 clusters.

For Rough K-Means, epsilon value is considered as 1.001 and number of clusters in each granule is been decided by the function given below:

$$C_k = \frac{n_k}{\sum_{i=1}^{m} n_i} \times Total\ number\ of\ clusters \quad (11)$$

where C_k denotes the number of clusters assigned to information granule k. n_k is the number of members belonging to information granule k. m is the number of clusters in Rough K-Means. In this technique we are able to indentify 900 clusters.

V. Experimental Results

TABLE I Summary of the results obtained by the FCM

Granules	Number of Members	Number of Clusters	Data Size (in MB)
Granule 0	76090	85	56.1
Granule 1	39915	45	29.7
Granule 2	60151	45	44.22
Granule 3	265960	297	196.02
Granule 4	120024	134	88.44
Granule 5	23348	26	17.16
Granule 6	9612	11	7.26
Granule 7	151631	169	111.54
Granule 8	45472	51	33.66
Granule 9	13666	15	9.9
Total	805869	900	594
Original Data Set	660364	900	465

Table I is the summary of the results from FCM granular. Although the total segment increased from 660364 to 805869, we achieved the goal of reduced data size is to deal with one information granule at a time .

TABLE II Summary of the results obtained by the AFCM

Granules	*Number of Members*	*Number of Clusters*	*Data Size (in MB)*
Granule 0	20675	28	18.48
Granule 1	35324	48	31.68
Granule 2	215674	292	192.72
Granule 3	62388	85	56.1
Granule 4	4376	6	3.96
Granule 5	125769	170	112.2
Granule 6	2409	3	1.98
Granule 7	65409	89	58.74
Granule 8	2824	4	2.64
Granule 9	129761	176	116.16
Total	664609	901	595
Original Data Set	660364900	900	465

Table II is the summary of the results from AFCM granular. Although the total number of members increased from 562745 to 721390, we only deal with one information granule at a time. Therefore, we achieved the goal of reduced space-complexity.

TABLE III SUMMARY OF THE RESULTS OBTAINED BY THE RKM

Granules	*Number of Members*	*Number of Clusters*	*Data Size (in MB)*
Granule 0	122260	167	110.49
Granule 1	11112	15	9.92
Granule 2	6794	9	5.95
Granule 3	7552	10	6.62
Granule 4	167789	229	151.50
Granule 5	3369	5	3.31
Granule 6	44961	61	40.36
Granule 7	143504	196	129.67
Granule 8	37645	51	33.74
Granule 9	115378	157	103.87
Total	660364	900	595(Round off)
Original Data Set	660364	900	465

Table III is the summary of the results from RKM granular. The total number of members is exactly same as original data set but identifies more number of hidden highly structure motif patterns.

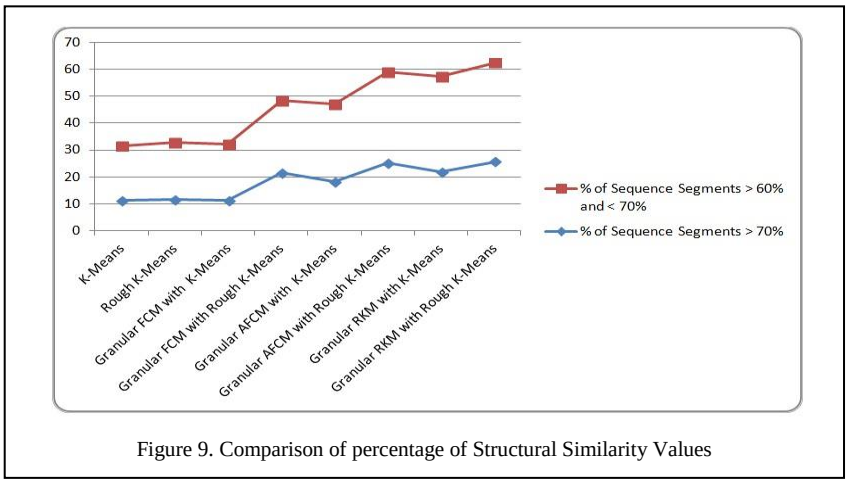


Figure 9. Comparison of percentage of Structural Similarity Values

Fig. 9 has been interpreted from table IV. From the Fig. 9 we state that the number of strong and weak clusters have been increased in Granular RKM with Rough K-Means technique as well as percentage of sequence segments have also been increased considerably.

TABLE IV COMPARISON RESULTS OF DIFFERENT ALGORITHMS

	K-Means	Rough K-Means	Granular FCM with K-Means	Granular FCM with Rough K-Means	Granular AFCM with K-Means	Granular AFCM with Rough K-Means	Granular RKM with K-Means	Granular RKM with Rough K-Means
No. of Clusters >70% Structural Similarity	100	103	101	195	164	228	196	231
No. of Clusters > 60% and < 70% Structural Similarity	184	193	188	241	260	304	320	332
% of Sequence Segments > 70%	11.11	11.44	11.22	21.67	18.20	25.31	21.78	25.67
% of Sequence Segments > 60% and < 70%	20.44	21.44	20.89	26.78	28.86	33.74	35.56	36.89
DBI Measure	6.2409	6.1985	4.2163	3.7339	3.9268	3.6186	3.8721	3.6005
Avg. HSSP-BLOSUM62	0.5268	0.6010	0.6125	0.6617	0.7325	0.7901	0.8125	0.8227

Table IV shows the comparative results obtained from different algorithms and granularization methods. From above table IV, we can infer that RKM with Rough K-Means method able to identify more number of hidden motif patterns.

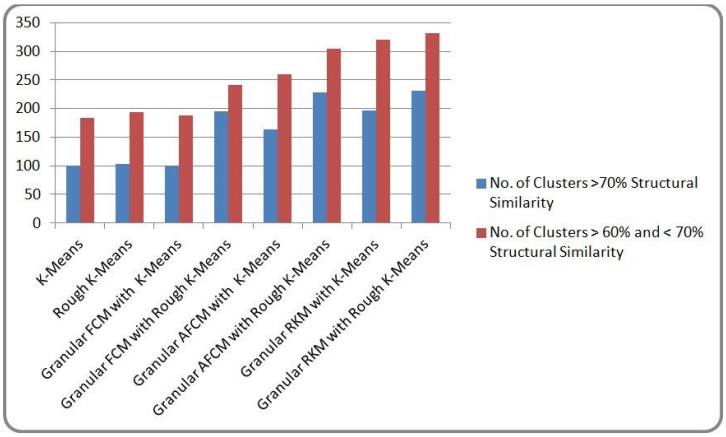


Figure 10. Comparison of Structural Similarity Values

Fig. 10 shows percentage of structural similarity belonging to clusters obtained from different methods and different granular computing techniques. Fig. 10 has been interpreted from table IV. From the Fig. 10, we state that the number of strong and weak clusters have been increased in RKM with Rough K-Means.

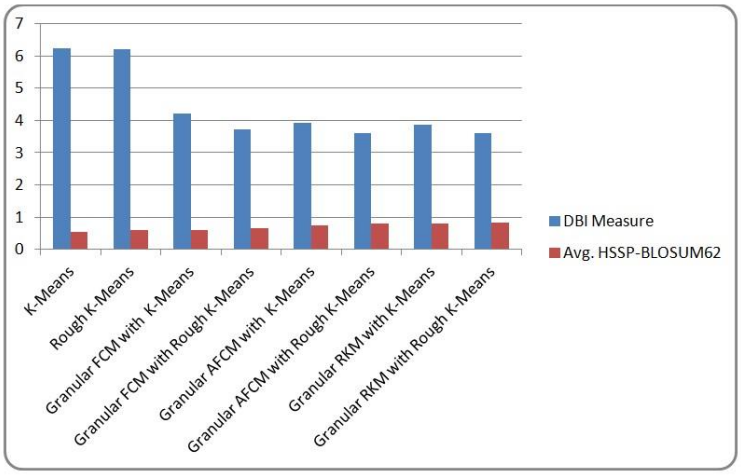


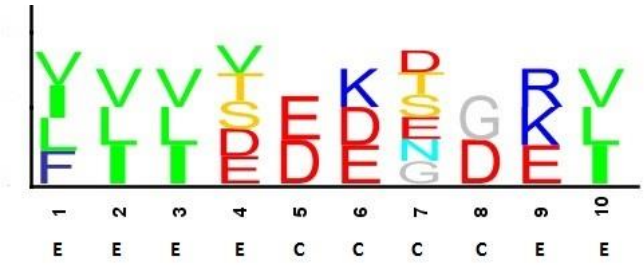
Figure 11. Comparison of DBI and BLOSUM62 measure values

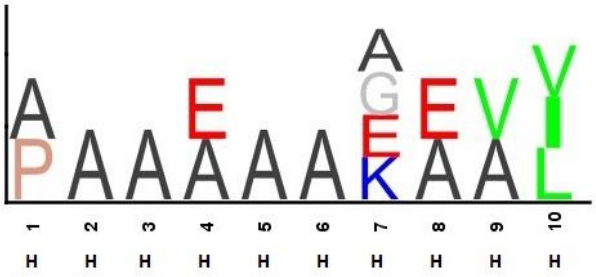
Fig. 11 shows DBI and HSSP-BLOSUM62 measure values obtained from different methods and different granular computing techniques.

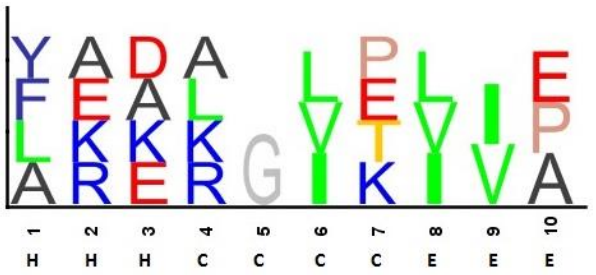
Low DBI measure value indicates the improvement of the quality of clusters RKM with Rough K-Means technique. High HSSP-BLOSUM62 value shows that RKM with Rough K-Means indicates that motif patterns are more significant.

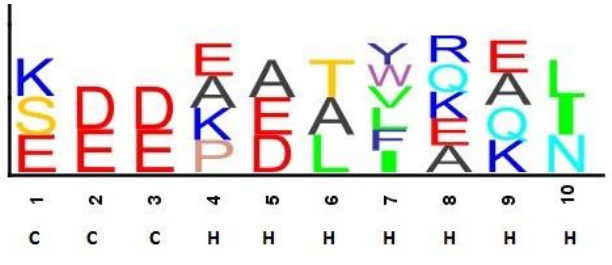
A. Sequence Motifs

Four different motif patterns obtained from RKM granular with Rough K-Means process are shown in motif tables I to IV. The following format is used for representation of each sequence motif table. Instead of using existing format, in this paper protein logo representation has been used [18].

Motif Table I
Sheets-Coils Motif
Number of Sequence Segments:171
Structural Similarity: 73.1


Motif Table II
Helices Motif
Number of Sequence Segments:258
Structural Similarity: 72.91


Motif Table III
Helices-Coils-Sheets Motif
Number of Sequence Segments:231
Structural Similarity: 71.13


Motif Table IV
Coils-Helices Motif
Number of Sequence Segments:102
Structural Similarity: 72.06


The above motif tables I-IV show the number of sequence segments belonging to this motif, percentage of structural similarity. The graph demonstrates the type of amino acid frequently appearing in the given position by amino acid logo. It only shows the amino acid appearing with a frequency higher than 8%. The

height of symbols within the stack indicates the relative frequency of each amino or nucleic acid at that position.

The x-axis label indicates the representative secondary structure (S), the hydrophobicity value (Hyd.) of the position. The hydrophobicity value is calculated from the summation of the frequencies of occurrence of Leu, Pro, Met, Trp, Ala, Val, Phe, and Ile.

VI. CONCLUSION

In this study, the granular computing models such as FGM and AFGM have studied and implemented. The RGM has been proposed in order to approximate some of the segments so as to include more similar segments in each granule. Further, the granules obtained in each of the above methods are clustered using K-Means and Rough K-Means. The highly structured clusters are used to construct the motif patterns. The main objective of generating more motif patterns has been achieved with the proposed rough granular approach and Rough K-Means clustering. It is believed that this granular strategy is a very useful and powerful for bioinformatics research involving an extremely large database.

ACKNOWLEDGMENT

The second author would like to thank the presented work supported by Special Assistance Programme of University Grants Commission, New Delhi, India (Grant No. F.3-50/2011 (SAP II)).

REFERENCES

[1] N. Hulo, C. J. a. Sigrist, V. Le Saux, P. S. Langendijk-Genevaux, L. Bordoli, A. Gattiker, E. De Castro, P. Bucher, and A. Bairoch, "Recent improvements to the PROCITE database", *Nucleic Acids Research*, vol. 32, no. Database, pp. D134-137, 2004.

[2] T. K. Attwood, M. Blythe, D. R. Flower, A. Gaulton, J. E. Mabey, N. Naudling, L. McGregor, A. Mitchell, G. Moulton, K. Paine, and P. Scordis, "PRINTS and PRINTS-S shed light on protein ancestry", *Nucleic Acids Research*, vol. 30, no. 1, pp. 239-241, 2002.

[3] S. Henikoff, J. G. Henikoff and S.Pietrokovski, "Blocks+: a non redundant database of protein Alignment blocks derived from multiple compilation", *Bioinformatics*, vol. 15, no. 6, pp. 417-479, 1999.

[4] C. Sander and R. Schneider, "Database of homology-derived protein Structures and the structural meaning of sequence alignment", *Proteins Struct. Funct. Genet*. vol. 9, no. 1, pp. 56–68, 1991.

[5] Henikoff, S. and Henikoff, J. G. (1992), Amino Acid Substitution Matrices from Protein Blocks, Proceedings of the *National Academy of Sciences of the United States of America*. 89, 10915-10919.

[6] Zhong, W., Altun, G., Harrison, R., Tai, P. C. & Pan, Y. (2005) "Improved K-Means clustering algorithm for exploring local protein sequence motifs representing common structural property", *NanoBioscience, IEEE Transactions* on. 4, 255-265.

[7] K. F. Han and D. Baker, "Recurring local sequence motifs in proteins", J. Mol. Biol., vol. 251, no. 1, pp. 176–187, 1995.

[8] P. Lingras, C. West, "Interval set clustering of web users with rough K-Means", J. *Intell. Inform. Syst*. 23 (2004) 5–16.

[9] P. Lingras, R. Yan, C. West, "Comparison of conventional and rough K-Means clustering", in: International conference on Rough Sets, Fuzzy Sets, Data Mining and Granular Computing, Lecture Notes in Artificial Intelligence, vol. 2639, *Springer, Berlin*, 2003, pp. 130–137.

[10] G. Wang and R. L. Dunbrack, Jr., "PISCES: a protein sequence-culling server," *Bioinformatics*, vol. 19, no. 12, pp. 1589-1591, 2003.

[11] Bernard Chen, Phang C. Tai, Robert Harrison, and Yi Pan, "FGK model: A Efficient Granular Computing Model for Protein Sequence Motifs Information Discovery", *IASTED CASB 2006*, Dallas, proceeding pp. 56-61.

[12] Bernard Chen, Phang C. Tai, Robert Harrison, and Yi Pan, "FIK model: A Novel Efficient Granular Computing Model for Protein Sequence Motifs and Structure Information Discovery", *IEEE BIBE 2006*, Washington D.C., proceeding, pp. 20-26.

[13] E. Elayaraja, K. Thangavel, M. Chitralegha, T. Chandrasekhar, "Extraction of Motif Patterns from Protein Sequences using SVD with Rough K-Means Algorithm", *International Journal of Computer Science Issues (IJCSI)*, vol. 9, Issue 6, No. 2, pp. 350-356, ISSN (Online): 1694-0814,2012.

[14] Eskin, E. and Pevzner, P. A. Finding composite regulatory patterns in DNA sequences. *Bioinformatics*, 18 (Suppl. 1), 354–363, 2002.

[15] Lawrence, C. E., Altschul, S. F., Boguski, M. S., Liu, J. S., Neuwald, A. F. & Wootton, J. C. (1993). Detecting subtle sequence signals: a Gibbs sampling strategy for multiple alignment. Science, 62, 208-214.

[16] Bailey TL, Boden M, Buske FA, Frith M, Grant CE, Clementi L, Ren J, Li WW and Noble WS: MEME SUITE: Tools for motif discovery and searching. *Nucleic Acids Research 2009.*

[17] Bhattacharya, S. (2009). Gibbs Sampling Based Bayesian Analysis of Mixtures with Unknown Number of Components. *Sankhya. Series B*. To appear.

[18] B.Chen, P.C Tai, R.Harrison and Y.Pan, "Super GSVM-FE model for protein Sequence Motif Information Extraction", in proc.IEEE symposium on *Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, 2007, pp. 317322.

[19] E. Cox, Fuzzy Modelling and Genetic Algorithms for Data Mining Exploration, *Elsevier*, 2005.

[20] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: A review," *ACM Comput. Surv*., vol. 31, no. 3, pp. 264–323, 1999.

[21] W.Kabsch and C.Sander, "Dictionary of protein secondary structure pattern recognition of hydrogen-bonded and geometrical features", *Biopolymers*, vol. 22, pp. 2577-2637, 1983.

[22] Cuff JA, Barton GJ., "Evaluation and improvement of multiple sequence methods for protein secondary structure prediction", *Proteins 1999*; 34:508 –519.
[23] M. Chitralegha and K. Thangavel "Protein sequence motif patterns using adaptive Fuzzy C-Means granular computing model", *Proceedings of the IEEE International Conference on Pattern Recognition, Informatics and Medical Engineering (PRIME)*, IEEE xplore, pp. 96 – 103, Print ISBN: 978-1-4673-5843-9, 2013 .
[24] Peters G., "Some refinements of rough k-means clustering". *Pattern Recognition Letters 25(12)*, pp. 1481-1491, 2006.
[25] Henikoff, S., Henikoff, J.G., Alford, W.J. & Pietrokovski, S., "Automated construction and graphical presentation of protein blocks from unaligned sequences", Gene 163, GC17-26 (1995).
[26] Kyle L. Jensen et al., "A Generic motif discovery algorithm for sequential data", Bioinformatics, vol. 22, no.1, pp. 21-28, 2006.