Exploring Highly Structure Similar Protein Sequence Motifs using Granular Computing Model based on Adaptive FCM

E.Elayaraja, K.Thangavel, M.Chitralegha, T.Chandrasekhar

Abstract— Protein sequence motifs are very important to the analysis of biologically significant conserved regions to determine the conformation, function and activities of the proteins. These sequence motifs are identified from protein sequence segments generated from large number of protein sequences. All generated sequence segments may not yield potential motif patterns. In this paper, short recurring segments of proteins are explored by utilizing a granular computing strategy. Initially, Fuzzy C-Means (FCM) and Adaptive Fuzzy C-Means clustering algorithms (AFCM) are used to separate the whole dataset into several smaller informational granules and then succeeded by K-Means and Rough K-Means clustering algorithms on each granule to obtain the final results. By comparing the results of two different granular techniques shows that Adaptive FCM granular with Rough K-Means clustering is capable to capture better motif patterns suggests that our granular computing model which combined AFCM granular with Rough K-Means have a high chance to be applied in some other bioinformatics research fields.

Index Terms— Protein Sequence Motifs, DBI, HSSP-BLOSUM62, Fuzzy C-Means, Adaptive Fuzzy C-Means, Rough K-Means.

1 Introduction

PROTEINS are involved in each and every body functions including nutrient transportation, muscle building, metabolism regulation, etc. Hence, proteins are essential for human health. Some of the most important functions of proteins are to regulate the expression of other proteins. Higher order structures such as motifs and domains are said to be some of components of proteins. The term “protein sequence motif” denotes amino-acid sequence pattern and has biological significance. These motif patterns may be able to predict other proteins’ structural or functional areas, such as binding sites, conserved domains, prosthetic attachment site, etc. Several popular motif databases, such as: PROSITE [1], PRINTS [2], BLOCKS [3], etc, share the weakness that most of the data are developed base on multiple alignments.

The most important motif finding tools are MITRA [14], Block Maker [15], MEME [16], and Gibbs Sampling [17]. But, these methods will generate motif patterns only for a single protein sequence and may carry only a little information about conserved sequence regions which transcend protein families because the size of input dataset is limited. Instead, in this paper, a huge number of segments are generated using sliding window technique [10] and patterns are extracted from the selected segments. Multiple protein sequences are represented by their corresponding HSSP file [4]. To avoid the high computational cost caused by a huge input dataset, we applied granular computing models that utilized Fuzzy C-Means and Adaptive Fuzzy C-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 join the information generated by all granules and obtain the final sequence motif information. Three evaluation methods are applied in this paper such as Structural similarity, DBI measure, and a novel HSSP-BLOSUM62 evaluation method.

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, including two clustering algorithms have been explained. Experimental Setup is explained in section 4. In section 5, Experimental Results are explained. In Section 6 concludes the paper with directions for further enhancement.

2 Related Work

Many computational approaches have been introduced for the problem of motif identification in a set of biological sequences which are classified based on type of motifs discovered. Finding recurring sequence motifs through K-Means clustering was initially done by Han and Baker [7]. In order to overcome the problem of sensitivity of initial points for cluster centers, a greedy method proposed by Zhong.et al [6] is used for initialization method for clustering.

Motif detection from a huge amount of sequences is a challenging task and not all the segments generated are so important. Therefore, Bernard Chen [18] has proposed Super Granular SVM Feature Elimination. In this approach the original dataset is first partitioned using Fuzzy C-Means clustering and then for each partition Greedy K-Means

- *E.Elayaraja is currently pursuing Ph.D., in Computer Science in Periyar University, Salem, India. E-mail: elayarajaphd.e@gmail.com*
- *K. Thangavel is currently working as Professor and Head, Department of Computer Science in Periyar University, Salem, India. E-mail: drktvelu@yahoo.com*

  M.Chitralegha is currently pursuing Ph.D., in Computer Science in Periyar University, India. E-mail: achitra_legha04@yahoo.co.in
- *T. Chandrasekhar is currently pursuing Ph.D., in Computer Science in Bharthiyar University, India. E-mail: ch_ansekh80@rediffmail.com*

clustering algorithm is been implemented. In this paper, our goal is to produce more clusters with good structural similarity.

3 GRANULAR COMPUTING STRATEGIES

Granular computing represents information in the form of aggregates. For a huge and complicated problem, it uses the divide-and-conquer concept to split the original task into several smaller subtasks to save time and space complexity. The process of splitting is comprehends the problem without including meaningless information called 'information granules'.

3.1 Fuzzy Granular Model

As this model works by building a set of information granules by Fuzzy C-Means (FCM) and then applying K-Means and Rough K-Means Clustering algorithms to obtain the final information. Major advantages of the Fuzzy Granular Model (FGM) is to reduced time and space complexity, and higher quality granular information results. The FGM process is given below in Fig. 1 and Fig. 2 [12].

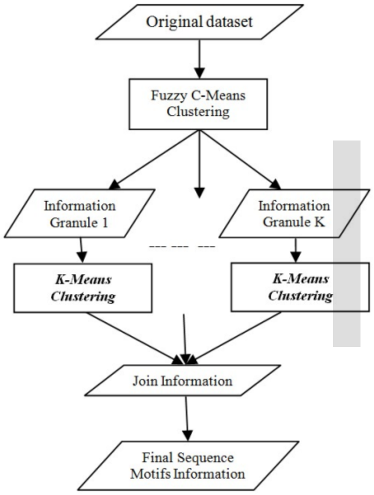


Fig. 1. Sketch of FGM using K-Means Computing Model

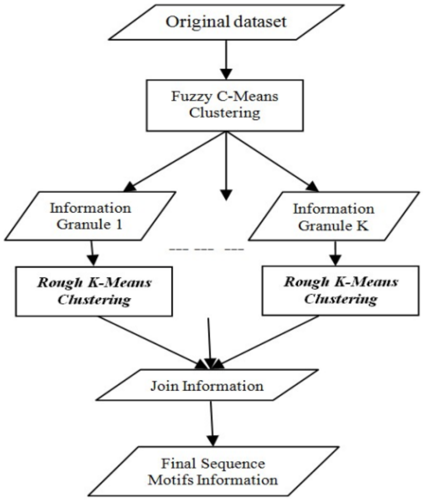


Fig. 2. Sketch of FGM using Rough K-Means Computing Model

Fuzzy C-Means (FCM) is a clustering algorithm which allows one segment of data is belongs to one or more clusters. This method is frequently used in pattern recognition. This algorithm is to minimize the following objective function [14]:

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

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:

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

This iteration will stop when $\max_{ij}\left\{\left|U^{(k+1)} - U^{(k)}\right|\right\} < \delta$ 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 function of

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^{m} . x_i}{\sum_{i=1}^{N} u_{ij}^{m}}$$

3. Update $U^{(k)}$and $U^{(k+1)}$by using equation (2).

4. if $\left|U^{(k+1)} - U^{k}\right| < Ɛ$ then stop; otherwise return to step 2

3.2 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].

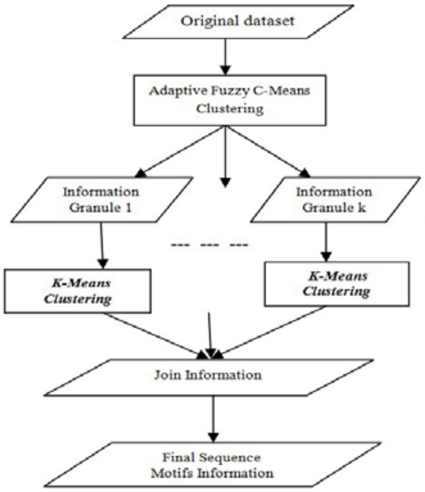


Fig. 3. Sketch of AFGM using K-Means Computing Model

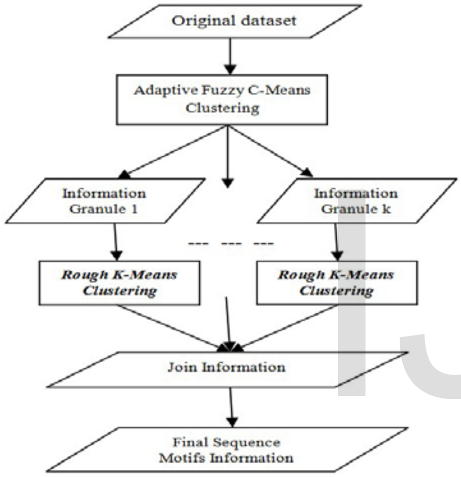


Fig. 4. Sketch of AFGM using Rough K-Means Computing Model.

3.2.1 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 and Keller [19] modified the approach for calculating membership values. Equation (3) shows membership calculation.

$$\sum_{j=1}^{k}\sum_{i=1}^{n}\mu_{j\,(x_i)=n} \qquad (3)$$

Here,

μ_j (x_i) is the membership of x_i in jth 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 (4)

$$\mu_j(x_i) = \frac{n\left(\frac{1}{d_{ji}}\right)^{\frac{1}{m-1}}}{\sum_{k=1}^{p}\sum_{z=1}^{n}\left(\frac{1}{d_{kz}}\right)^{\frac{1}{m-1}}} \qquad (4)$$

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.

3.3 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].

3.4 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.

3.4.1 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.

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

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.

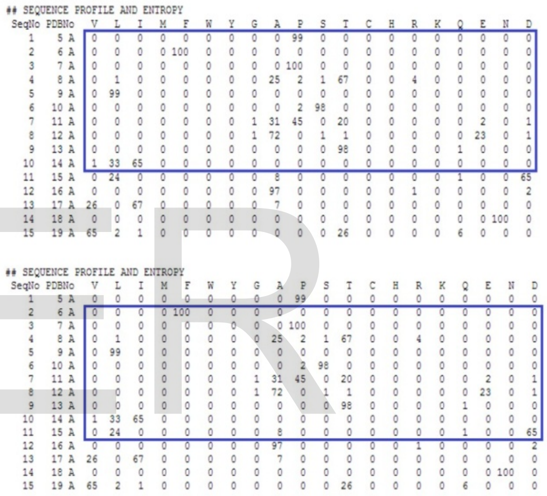
----

4 Experimental Setup

4.1 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 are used to represent each segment [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. 5 shows sliding window technique.



```
## 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     5 A  0   0   0   0   0   0   0   0   0  99   0   0   0   0   0   0   0   0   0   0
    2     6 A  0   0   0   0 100   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    3     7 A  0   0   0   0   0   0   0   0   0 100   0   0   0   0   0   0   0   0   0   0
    4     8 A  0   1   0   0   0   0   0   0  25   2   1  67   0   0   4   0   0   0   0   0
    5     9 A  0  99   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    6    10 A  0   0   0   0   0   0   0   0   0   2  98   0   0   0   0   0   0   0   0   0
    7    11 A  0   0   0   0   0   0   0   1  31  45   0  20   0   0   0   0   0   2   0   1
    8    12 A  0   0   0   0   0   0   0   1  72   0   1   1   0   0   0   0   0  23   0   1
    9    13 A  0   0   0   0   0   0   0   0   0   0   0  98   0   0   0   0   1   0   0   0
   10    14 A  1  33  65   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
   11    15 A  0  24   0   0   0   0   0   0   8   0   0   0   0   0   0   0   1   0   0  65
   12    16 A  0   0   0   0   0   0   0   0  97   0   0   0   0   0   1   0   0   0   0   2
   13    17 A 26   0  67   0   0   0   0   0   7   0   0   0   0   0   0   0   0   0   0   0
   14    18 A  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0 100   0
   15    19 A 65   2   1   0   0   0   0   0   0   0   0  26   0   0   0   0   6   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     5 A  0   0   0   0   0   0   0   0   0  99   0   0   0   0   0   0   0   0   0   0
    2     6 A  0   0   0   0 100   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    3     7 A  0   0   0   0   0   0   0   0   0 100   0   0   0   0   0   0   0   0   0   0
    4     8 A  0   1   0   0   0   0   0   0  25   2   1  67   0   0   4   0   0   0   0   0
    5     9 A  0  99   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
    6    10 A  0   0   0   0   0   0   0   0   0   2  98   0   0   0   0   0   0   0   0   0
    7    11 A  0   0   0   0   0   0   0   1  31  45   0  20   0   0   0   0   0   2   0   1
    8    12 A  0   0   0   0   0   0   0   1  72   0   1   1   0   0   0   0   0  23   0   1
    9    13 A  0   0   0   0   0   0   0   0   0   0   0  98   0   0   0   0   1   0   0   0
   10    14 A  1  33  65   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
   11    15 A  0  24   0   0   0   0   0   0   8   0   0   0   0   0   0   0   1   0   0  65
   12    16 A  0   0   0   0   0   0   0   0  97   0   0   0   0   0   1   0   0   0   0   2
   13    17 A 26   0  67   0   0   0   0   0   7   0   0   0   0   0   0   0   0   0   0   0
   14    18 A  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0 100   0
   15    19 A 65   2   1   0   0   0   0   0   0   0   0  26   0   0   0   0   6   0   0   0
```

Fig. 5. Sliding Window techniques with a window size of 10 applied on 3CA8 HSSP file.

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

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].

4.2 Structural similarity measure

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

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

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.

4.3 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:

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

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.

4.4 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:

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

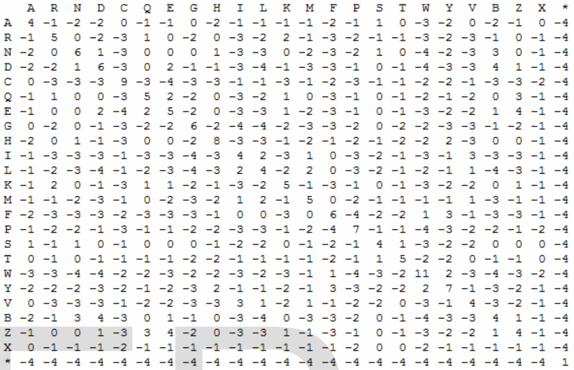
$$d_{intra}(C_p) = \frac{\sum_{i=1}^{n_p}\|g_i - g_{pc}\|}{n_p} \quad and$$

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

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 pairwise 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.

4.5 HSSP-BLOSUM62 Measure

BLOSUM62 [5] (Fig. 6.) 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

Fig. 6. 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= $\dfrac{\sum_{i=1}^{k-1}\sum_{j=i+1}^{k} HSSP_i.HSSP_j.BLOSUM62_{ij}}{\sum_{i=1}^{k-1}\sum_{j=i+1}^{k} HSSP_i.HSSP_j}$

4.6 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:

$$C_k = \frac{n_k}{\sum_{i=1}^{m} n_i} \times Total\ number\ of\ cluster$$

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:

$$C_k = \frac{n_k}{\sum_{i=1}^{m} n_i} \times Total\ number\ of\ cluster$$

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.

5 Experimental Results

TABLE 1

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	67	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 1 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 2

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	660364	900	465

one information granule at a time. Therefore, we achieved the goal of reduced space-complexity.

Table 3 shows the comparative results obtained from different algorithms and granularization methods. From above table 3, we can infer that Adaptive FCM with Rough K-Means method able to identify more number of hidden motif patterns.

Fig. 7 has been interpreted from table 3. From the below Fig. 7 we state that the number of strong and weak clusters have been increased in Granular AFCM with Rough K-Means technique as well as percentage of sequence segments have also been increased considerably.

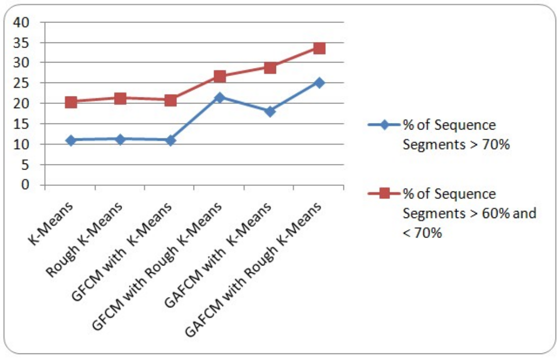


Fig. 7. Comparison of percentage of Structural Similarity Values

TABLE 3

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
No of clusters >70% Structural Similarity	100	103	101	195	164	298
No of clusters > 60% and < 70% Structural Similarity	184	193	188	241	260	304
% of Sequence Segments > 70%	11.11	11.44	11.22	21.67	18.20	33.07
% of Sequence Segments > 60% and < 70%	20.44	21.44	20.89	26.78	28.86	33.74
DBI Measure	6.2409	6.1985	4.2163	3.7339	3.9268	3.6186
Avg HSSP-BLOSUM62	0.5268	0.6010	0.6125	0.6617	0.7325	0.7901

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

Fig. 8 shows percentage of structural similarity belonging to clusters obtained from different methods and different granular computing techniques. Fig. 8 is been interpreted from table 3. From the blow figure we state that the number of strong and weak clusters have been increased in Adaptive FCM with Rough K-Means.

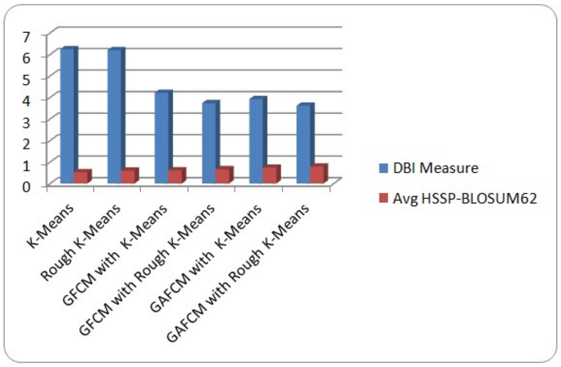


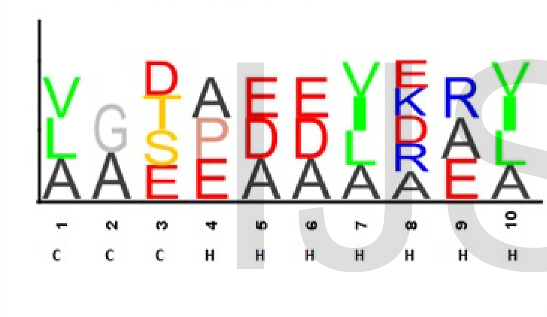
Fig. 9. Comparison of DBI and HSSP-BLOSUM62 measure

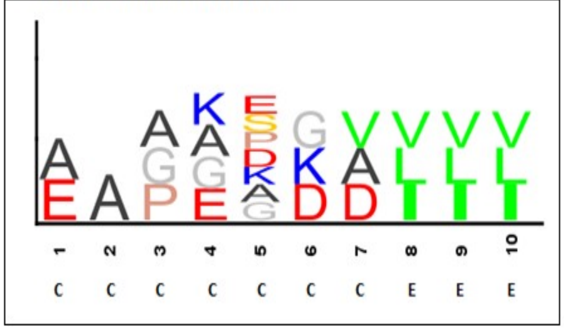
values

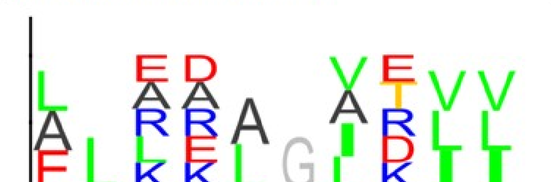
Fig. 9 shows DBI and HSSP-BLOSUM62 measure values obtained from different methods and different granular computing techniques.

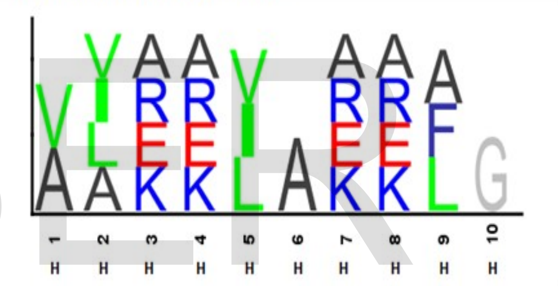
5.1 Sequence Motifs

Four different motif patterns obtained from adaptive Fuzzy C-Means with Rough K-Means process are shown in motif tables 1-4. 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 1
Coils-Helices Motif
Number of Sequence Segments:170
Structural Similarity: 72.59


Motif Table 2
Coils-Sheets Motif
Number of Sequence Segments:252
Structural Similarity: 75.04


Motif Table 3
Helices-Coils-Sheets Motif
Number of Sequence Segments:455
Structural Similarity: 77.43


Motif Table 4
Helices-Coils Motif
Number of Sequence Segments:212
Structural Similarity: 86.60


- The above motif tables 1-4 shows 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.

6 Conclusion

In this study, two different novel granular computing models such as Fuzzy C-Means granular with K-Means and Rough K-Means clustering and then Adaptive Fuzzy C-Means granular with K-Means and Rough K-Means clustering methods have been proposed to identify hidden protein sequence motifs. These models are used to split the whole dataset into several information granules and analyze each granule to identify motif information. Analysis of sequence motifs also shows that granular computing technology may detect some subtle sequence information overlooked by K-Means and Rough K-Means clustering algorithms alone. By comparing the results of two different granular techniques shows that Adaptive FCM with Rough K-Means is capable to filter outliers and to capture better results. It is believed that this novel strategy is a very powerful tool 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, NewDelhi, 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.