2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)
July 6-11, 2014, Beijing, China

Structural Classification of Proteins through Amino Acid Sequence using Interval Type-2 Fuzzy Logic System

Thanh Nguyen, Abbas Khosravi, Douglas Creighton and Saeid Nahavandi
Centre for Intelligent Systems Research (CISR), Deakin University
Waurn Ponds, Victoria, Australia, 3216
E-mail: thanh.nguyen@deakin.edu.au

***Abstract*—This paper introduces a new multi-output interval type-2 fuzzy logic system (MOIT2FLS) that is automatically constructed from unsupervised data clustering method and trained using heuristic genetic algorithm for a protein secondary structure classification. Three structure classes are distinguished including helix, strand (sheet) and coil which correspond to three outputs of the MOIT2FLS. Quantitative properties of amino acids are used to characterize the twenty amino acids rather than the widely used computationally expensive binary encoding scheme. Amino acid sequences are parsed into learnable patterns using a local moving window strategy. Three clustering tasks are performed using the adaptive vector quantization method to derive an equal number of initial rules for each type of secondary structure. Genetic algorithm is applied to optimally adjust parameters of the MOIT2FLS with the purpose of maximizing the Q3 measure. Comprehensive experimental results demonstrate the strong superiority of the proposed approach over the traditional methods including Chou-Fasman method, Garnier-Osguthorpe-Robson method, and artificial neural network models.**

I. Introduction

Proteins are large biological molecules comprising one or more chains of amino acids. Proteins serve a variety of functions within living organisms, including catalysing metabolic reactions, replicating DNA, providing structural support for living cells, responding to stimuli, protecting the body from the effect of invading species or substances, and transporting molecules from one location to another. Proteins are distinguished primarily based on their sequence of amino acids. Protein structure is the biomolecular structure of protein molecule, which consists of four distinct levels: primary, secondary, tertiary and quaternary structure. Protein secondary structures, which refer to highly regular local sub-structures, are primarily formed by short and long ranging interactions throughout the protein's folding process. There are two main types of secondary structure, the α helix and the β strand and coil. Other category system can also classify secondary structures into eight classes: α-helix (H), 3_{10} helix (G), π-helix (I), β-strand (E), bridge (B), β-turn (T), bend (S) and coil (C).

Protein secondary structure prediction (PSSP) is important because knowing protein secondary structure can help to comprehend and understand functions of proteins. Secondary structure knowledge may also aid to predict protein's three-dimensional structure. Furthermore, PSSP can be included in threading methods to help the detection of distantly related proteins. Most prediction methods rely upon the local information and the correlation between primary and secondary structure. Early methods applied for PSSP include those of Chou-Fasman (CF) [2], Garnier-Osguthorpe-Robson (GOR) [3] and artificial neural network (NN) models [4, 5]. Since then there has been a vast number of studies concerning PSSP including those of recent works, i.e. [6-11].

However, those methods are not able to capture and handle the imprecision and vagueness inherent in the protein structure data, the more so as the length of amino acid sequences augments and especially the number of proteins available is increasingly advanced. Moreover, the practice of secondary structure assignment of proteins is not always precise due to the limit of chemical technology. Neural networks applied for PSSP usually employ the binary amino acid encoding, which faces a big challenge of computation costs because each amino acid is characterized by an array of size 20. In order to eradicate this burden, this paper introduces a method to characterize amino acids using quantitative properties consisting of solvent exposed area, hydrophobicity, pKa values of ionizing groups COOH and NH_3 and weights or volumes of amino acid residues. These properties of amino acids are not always precisely determined but may vary depending on the environment they are assessed. In other words, they are vague and uncertain therefore we ought to propose a tool to handle them. Fuzzy logic [12] has been introduced and renowned as a powerful mechanism for uncertainty modelling. Mocz [13], Boberg et al. [14] and Hering et al. [15] have already applied type-1 fuzzy logic for PSSP. Though original fuzzy logic, type-1 fuzzy logic (FL), has been introduced almost half a century back, but it has been argued that it is unable to properly handle uncertainties [16]. The type-2 FL [17], which is the extension of the type-1 FL, is able to more efficiently and effectively handle uncertainties mainly due to its three dimensional membership functions [18].

This paper presents a systematic way for PSSP using interval type-2 fuzzy logic system (IT2FLS). This is the first exploration of type-2 FL in protein structure prediction to our best knowledge. Throughout this study, we quantitatively demonstrate the efficiency of this classifier for

978-1-4799-2072-3/14/$31.00 ©2014 IEEE

properly addressing the challenging problem of PSSP. Three classes of structures are distinguished based on the three outputs of the fuzzy system. Data patterns are extracted from local information of amino acids and transformed into numerical representation through a mapping procedure, which utilizes the quantitative attributes of amino acid residues. Before proceeding to the detailed methodology of the study in Section IV, we review some previous fundamental methods used widely in the literature for PSSP including CF, GOR and NN in the next section. Section III proposes a new amino acid encoding scheme. Section V is devoted for experimental results followed by discussions. Concluding remarks are presented in Section VI.

II. Brief review of PSSP techniques

A. Traditional methods: Chou-Fasman & GOR methods

Secondary structure prediction proposed by Chou and Fasman [2] is one of the simplest statistical approaches, which is based on observed frequency of each type of amino acid residues in α helix, β strand and turns of the known protein structures.

Secondary structure prediction method by GOR [3] is another popular method utilizing information theory, which is more complicated compared to the Chou-Fasman method.

The Chou-Fasman method assumes that each amino acid independently influences secondary structure within a window of sequence whereas in the GOR method amino acid flanking the central amino acid residue is supposed to impact the likely secondary structure of the central residue.

B. Neural network model

The principle of NN application in PSSP is based on the amino acid binary encoding scheme (e.g. see [4, 5]). Each amino acid residue is characterized by a binary array of size 20. The element corresponding to the amino acid type in the given position is set to 1, whilst all other positions are set to 0. A sliding window is employed in input amino acid sequence to encode the input layer, and the secondary structure of the central residue in the window will be predicted. The structural state of a given residue and the eight residues on either side of the prediction point is found to be statistically correlated. Therefore, a window of size 17 is deployed. Accordingly, the input layer of the NN comprises $R = 17x20$ input nodes, i.e. 17 groups of 20 inputs each.

The output layer of the NN consists of three nodes corresponding to three secondary structural states (or classes), which are also encoded using a binary scheme: [1 0 0] for coil, [0 1 0] for sheet, and [0 0 1] for helix.

III. Protein Encoding Strategy

A. Amino acid encoding

As the relationships between amino acid sequence and secondary structure of proteins need to be explored, the 20 amino acids must be numerically encoded before processing. The traditional binary mapping where each amino acid is represented by an array of 20 digits 0 and 1 has been widely used in the literature. This orthogonal encoding uses a lot of inputs and is computationally demanding (memory and convergence time issues). For example, for a short sequence of 5 residues, the processing systems must adopt 100 inputs. We therefore initiate a new effective encoding scheme for the 20 amino acids. A range of properties/attributes deemed affecting the secondary structure of proteins is utilized to characterize the 20 amino acids. The attributes include: hydrophobicity, volume, solvent exposed area, and pK_a values of the ionizing groups of amino acids. Note that all these attributes are normalized into the interval [0-1] to avoid the influence of one attribute to another due to the difference in scales of attributes.

1) Amino acid hydrophobicity

Hydrophobicity relatively measures how soluble an amino acid is in water. These values may vary depending on the pH level of the solution. Hydrophobic amino acids are often found in the interior whereas hydrophilic amino acids are usually in contact with the aqueous environment. There are various scales proposed so far including Kyte and Doolittle [19], and Wimley and White [20]. Palliser and Parry [21] investigated 100 hydrophobicity scales and claimed that locating β-strands on the surface of proteins can be helped by using these scales. As patterns of hydrophobic amino acids may aid structure prediction [22], the utilization of this amino acid attribute could help improve the PSSP. This paper employs the most recent hydrophobicity scale of Hessa et al. [23]. Unlike the others, more negative values are corresponding to greater hydrophobicity in the Hessa et al. scale.

2) pK_a values of the ionizing groups of amino acids

An acid dissociation constant, K_a, measures quantitatively the strength of an acid in solution, which is usually represented as a quotient of the equilibrium concentrations (in mol/L), denoted by $[HA]$, $[A^-]$ and $[H^+]$:

$$K_a = \frac{[A^-][H^+]}{[HA]} \quad (1)$$

Because K_a values span on many orders of magnitude, a logarithmic value of the acid dissociation constant, pK_a, is commonly used in practice.

$$pK_a = -\log_{10} K_a \quad (2)$$

pK_a values can be represented in the other form, which is well known the Henderson-Hasselbalch equation: $pK_a = pH + \log_{10}[HA] / [A-]$ [24].

In this paper, in order to characterize the 20 amino acids, we employed pK_a values of the ionizing groups of amino acids, pK_{a1} represents the carboxyl group (COOH) whilst pK_{a2} is of the ammonium ion (NH_3).

Forsyth et al. [25] investigated 24 proteins of known structure and found the empirical relationships between protein structure and carboxyl pK_a values though these relationships are not very precise. There are a number of sources that cause the lack of precision. The uncertain findings from that research again advocate the use of fuzzy logic in modelling amino acids through their pK_a values.

3) Solvent exposed area (SEA) of amino acids ($Å^2$)

The SEA of an amino acid in a protein represents the extent the amino acid is accessible to the solvent

surrounding the protein. Regarding the SEA of amino acids, there are two categories are distinguished: hydrophobic and hydrophilic. Amino acids buried inside the structure of protein and thus shielded from the solvent are classified as hydrophobic, whilst amino acids close to the surface and thus exposed to the solvent are hydrophilic. However, this classification is not always precise due to the popularity of many biological rules exceptions. Hydrophilic residues are often found to be buried in the native structure and hydrophobic residues are often found close to the protein surface. The imprecision and uncertainty in measurement of the SEA of amino acid residues accordingly inspire the employment of fuzzy logic to characterize amino acids in PSSP. As numerous studies have found the significance of the amino acid SEA in understanding protein structures and functions [26-28], utilizing the SEA therefore would enhance the predictability of the protein secondary structure. In this paper, triples representing SEA of amino acids are employed as part of the amino acid representation.

4) Volume or weight of amino acids

The van der Waals volume of a protein molecule is the enclosed space composed of the van der Waals spheres of the constituent atoms. The van der Waals of small molecules can be computed by adding the values of constituent atoms or chemical groups if they are not structurally strained.

Molecular weight of the protein composition is a sum of the molecular weights of the individual amino acid residues removing one H_2O molecule per peptide pond (the water weight is 18.01, alpha-amino group is 8.56 and alpha-carboxyl group is 3.56). This calculation is based on the assumption that no covalent modification has been applied to proteins.

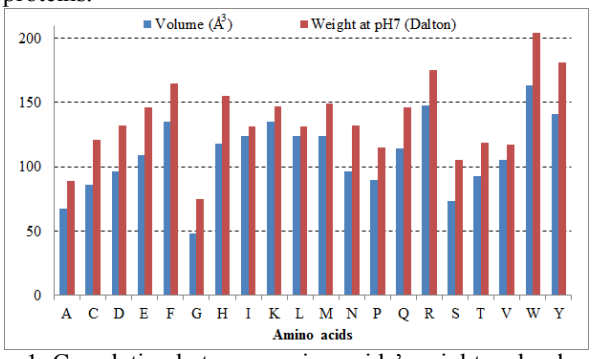


Fig. 1. Correlation between amino acids' weight and volume

As structural and biochemical characteristics of an amino acid in general or volume/mass in particular influence the local structural conformation of proteins [29], the inclusion of volume or weight in amino acid characterization would augment the PSSP performance. Moreover, because volume and weight of amino acids show a strong correlation as depicted in Fig. 1, we therefore just utilize only volume to characterize amino acids.

B. Output mapping

Generally, there are two possible ways for the amino acid and structure assignment: secondary structure must be given to amino acids based on the examination of the structure coordinates of the atoms in the PDB file or the three-dimensional structure has been solved. Three popular algorithms used for protein secondary structure assignment are DSSP [30], STRIDE [31] and DEFINES [32]. The widely used algorithm is DSSP, which is the method of mapping between the atomic-resolution coordinates of the protein and the secondary structure. Eight categories of secondary structure are assigned as follows. Three types including 3_{10} helix (G), α helix (H), π helix (I) are realized by having a repeated sequence of hydrogen bonds in which the residues are three, four or five residues separately respectively. The other types include beta bridge (B), which is a longer set of hydrogen bonds, beta bulges (E), turns (T), featuring hydrogen bonds typical of helices, loops (S), regions of high curvature, blank or (C), no other rules applies or loops. The three type category classifies the protein structure into α helix, β strand and coils.

Table. 1. Protein secondary structure reduction

Structure	Reduced structure
H, G, I	H (helix)
E, B	E (strand)
All other	C (coil)

In this paper, the above three-type category of protein structure (Table 1) is employed to demonstrate the performance of MOIT2FLS in PSSP. The next section briefly summarizes IT2FLS and steps to design a MOIT2FLS for protein structure prediction.

IV. Type-2 Fuzzy Logic Systems for PSSP

A. Fuzzy Background

A fuzzy logic system (FLS) is called a type-1 FLS (T1 FLS) if it is described completely using type-1 fuzzy sets (T1 FSs) whilst a FLS that uses at least one type-2 fuzzy set (T2 FS) is called a type-2 FLS (T2 FLS) [33, 34]. A T2 FLS has more degrees of freedom than does a T1 FLS because it comprises more parameters. It therefore suggests that T2 FLS has the potential to outperform a T1 FLS because of its larger number of design degrees of freedom. If uncertainties vanish, a type-2 FLS diminishes to a type-1 FLS.

Basically the T2 FLS structure is similar to that of the T1 FLS. The major differences are in using the T2 FSs (rather than T1 FSs) in antecedent parts of fuzzy rules and the output processor. The output processor of a T1 FLS transforms a T1 FS to a crisp number whilst a T2 FLS has two components in the output processor. The first is a type reduction that transforms a T2 FS into a T1 FS and the second is the defuzzifier that transforms a T1 FS into a crisp number. A general T2 FLS requires extensive computational cost and complicated implementation compared to a T1 FLS. A special case of T2 FLS, interval type-2 FLS (IT2 FLS) has been widely used for reduced computational burden [35]. In this paper, we proposed a new IT2FLS with 3 outputs corresponding with 3 classes of the protein secondary structure.

B. Multi-Output Interval Type-2 Fuzzy Logic Systems

The T2 FLSs [36] deployed on the basis of T2 FSs. A general T2 FS is represented in three dimensions. The membership degree is not a crisp number but it is a FS.

Third dimension is the degree of the membership function (MF) at each point on footprint of uncertainty (FOU), which is the two-dimensional domain.

Since computation of the general T2 FSs is extensively complicated, ones tend to use interval type-2 FSs (IT2 FSs) because of its simplicity. IT2 FSs represent the membership degree by an interval rather than a FS. The third dimension value in the IT2 FS is the same everywhere so that it is ignored and only the FOU is used to describe the IT2 FS.

Similar to T1 FLSs and based on different fuzzy rule types, e.g. Mamdani, Takagi-Sugeno-Kang (TSK) or Tsukamoto, there are corresponding different IT2 FLSs can be implemented. In this paper, the special case of TSK fuzzy rule is employed to develop an IT2 FLS [37] for the protein secondary structure prediction. The IT2 FLS has demonstrated its effectiveness in a number of applications in the literature, e.g. see [38-41] for recent studies.

Some variants of IT2 FLS are recognized depending on the MFs used in the antecedent and consequent parts are IT2 FSs and/or T1 FSs. We deploy herein the MOIT2FLS where antecedents are IT2 FSs and consequents are interval T1 FSs. Assume the IT2 FLS consists of K rules and p antecedents in each rule, denote the lth rule by R^l as follows:

R^l: IF x_1 is $\tilde{F}_1^l$ and ... and x_p is $\tilde{F}_p^l$, THEN $Y_1^l = C_1^l$ and

$$Y_2^l = C_2^l \text{ and } Y_3^l = C_3^l$$

where $l = 1,\dots,K$. $\tilde{F}_i^l$ is the ith IT2 FS defined by a lower and upper bound MF:

$$\mu_{\tilde{F}_i^l}(x_i) = \left[\underline{\mu}_{\tilde{F}_i^l}(x_i), \overline{\mu}_{\tilde{F}_i^l}(x_i)\right]$$

and C_i^l is an interval T1 FS characterized by its centre and spread c_i^l and s_i^l respectively:

$$C_i^l = \left[c_i^l - s_i^l, c_i^l + s_i^l\right]$$

where $i = 0,1,\dots,p$. Assume the input vector $x = (x_1, x_2, \dots, x_p)$, an IT2 FLS inference ought to go through the following steps:

- Compute the lower and upper membership degree of x_i on the corresponding antecedent part: $\underline{\mu}_{\tilde{F}_i^l}(x_i)$ and $\overline{\mu}_{\tilde{F}_i^l}(x_i)$.
- Compute the firing strength interval of the lth rule: $F^l = [\underline{f}^l, \overline{f}^l]$ where:

$$\underline{f}^l = \underline{\mu}_{\tilde{F}_i^l}(x_i) * \underline{\mu}_{\tilde{F}_i^l}(x_i) * \dots * \underline{\mu}_{\tilde{F}_i^l}(x_i)$$
$$\overline{f}^l = \overline{\mu}_{\tilde{F}_i^l}(x_i) * \overline{\mu}_{\tilde{F}_i^l}(x_i) * \dots * \overline{\mu}_{\tilde{F}_i^l}(x_i)$$

- Compute the output interval of the lth fuzzy rule for each of the three outputs, y_j^l, which is an interval T1 FS: $Y_j^l = [\underline{y}_j^l, \overline{y}_j^l]$ where $j = 1,2,3$.

$$\underline{y}_j^l = c_j^l - s_i^l$$
$$\overline{y}_j^l = c_0^l + s_i^l$$

- The final crisp value of each output of the IT2 FLS model is calculated by combining the corresponding outputs of K rules:

$$Y_j = \left[\underline{Y}_j, \overline{Y}_j\right]$$
$$= \int_{y_j^1 \in [\underline{y}_j^1, \overline{y}_j^1]} \dots \int_{y_j^K \in [\underline{y}_j^K, \overline{y}_j^K]} \int_{f^1 \in [\underline{f}^1, \overline{f}^1]} \dots \int_{f^K \in [\underline{f}^K, \overline{f}^K]} 1 \Big/ \frac{\sum_{l=1}^K f^l y_j^l}{\sum_{l=1}^K f^l}$$

In order to obtain a crisp output for the IT2 TSK FLS, a type-reduction and a defuzzifier are needed. The most popular type reduction is that of the iterative Karnik-Mendel procedure [42, 43]. This method however was found deficient and thus several other methods have been proposed in [44, 45]. Recently, Wu and Nie [46] introduced an enhancement on the iterative algorithm with stop condition (IASC) proposed in [47] for type-reduction. The method presented in [46], also the so-called EIASC algorithm, demonstrated more efficiency in type-reduction compared to previous methods. This paper accordingly employs the EIASC algorithm to calculate the values of $\underline{Y}$ and $\overline{Y}$. The brief presentation of the EIASC is as follows.

a) Calculating $\underline{Y}$:

1) Sort $\underline{y}^l$ $(l = 1,2,\dots,K)$ in increasing order and assign the sorted $\underline{y}^l$ by the same name, but now $\underline{y}^1 < \underline{y}^2 \dots < \underline{y}^K$. Link f^l with their corresponding $\underline{y}^l$ and renumber them so that their index matches to the renumbered $\underline{y}^l$.

2) Initialize

$$a = \sum_{l=1}^K \underline{y}^l \underline{f}^l,\ b = \sum_{l=1}^K \underline{f}^l,\ \underline{Y} = \underline{y}^K \text{ and } L = 0$$

3) Compute

$$L = L + 1,\ a = a + \underline{y}^L(\overline{f}^L - \underline{f}^L),$$
$$b = b + \overline{f}^L - \underline{f}^L,\ \underline{Y} = a/b$$

4) If $\underline{Y} \le \underline{y}^{L+1}$, stop; otherwise, go to Step (3).

b) Calculating $\overline{Y}$:

1) Sort $\bar{y}^l$ $(l = 1,2,\dots,K)$ in increasing order and assign the sorted $\bar{y}^l$ by the same name, but now $\bar{y}^1 < \bar{y}^2 \dots < \bar{y}^K$. Relate f^l with their corresponding $\bar{y}^l$ and renumber them so that their index links to the renumbered $\bar{y}^l$.

2) Initialize

$$a = \sum_{l=1}^K \bar{y}^l \bar{f}^l,\ b = \sum_{l=1}^K \bar{f}^l,\ \bar{Y} = \bar{y}^1 \text{ and } R = 0$$

3) Compute

$$a = a - \bar{y}^R(\overline{f}^R - \underline{f}^R),\ b = b - \overline{f}^R + \underline{f}^R,$$
$$\bar{Y} = a/b,\ R = R - 1$$

4) If $\bar{Y} \ge \bar{y}^R$, stop; otherwise, go to Step (3).

Finally, the crisp output of the IT2 FLS is derived as a mean of $\underline{Y}$ and $\overline{Y}$: $y = (\underline{Y} + \overline{Y})/2$. The above procedure is computed for each of the three outputs. The "winner-take-all" rule is used to infer the final output of the protein secondary structure as detailed below:

```
function Y = winner-take-all(Y1, Y2, Y3)
    if(Y1 >= max(Y2, Y3)
        Y = "H";
    elseif(Y2 > max(Y1, Y3)
        Y = "E";
    else
        Y = "C";
    end
end
```

Various IT2 MFs can be used in the IT2 FLS such as IT2 triangular, trapezoidal, Gaussian, Cauchy, Laplace, or

general bell-shaped MFs. In this paper, the IT2 Gaussian MF is employed as a demonstration:

$$\mu_{\tilde{F}_i^l}(x_i,k) = exp\left[-\frac{1}{2}\left(\frac{x_{i,k}-m_i^l}{\sigma_i^l}\right)\right] = N\left(m_i^l,\left[\sigma_{i,1}^l,\sigma_{i,2}^l\right]\right) \quad (3)$$

Once the type of MFs, the number of inputs and the number of fuzzy rules are determined, an IT2 FLS can be constructed. The determination of parameters of a FLS is extremely important because its performance depends mainly on this process. Theoretically, parameters of a FLS are commonly obtained by experts. The experts' knowledge however is limited. The more so if the number of rules increases and various proteins need to be investigated. Genetic algorithm (GA) is a popular tool to train FLSs to optimally tune their parameters. The following subsection presents the GA method and its application to train the MOIT2FLS.

C. Training MOIT2FLS by GA

A GA [48-50] is an unorthodox search or optimization technique operated on a population of n artificial individuals. Individuals are characterized by chromosomes (or genomes) S_k, $k = \{1, \dots, n\}$. The chromosome is a string of symbols, which are called genes, $S_k = (S_{k1}, \dots, S_{kM})$, and M is a string length. Individuals are evaluated via calculation of a fitness function. To evolve through successive generations, GA performs three basic genetic operators: selection, crossover and mutation. Through chromosomes' evolution, GA searches for the best solution(s) in the sense of the given fitness function. We employ GA to train the complicated FLSs comprising many parameters. The fitness function is the success rate Q_3 of the training fuzzy models, computed using the formula:

$$Q_3 = \frac{P_\alpha + P_\beta + P_{coil}}{N} \quad (4)$$

where N is the number of residues being predicted and P_α is the number of secondary structure of type α, which is correctly predicted. Parameters of fuzzy models are coded into genes of the GA chromosomes/individuals. With a population of individuals, GA can simultaneously explore different parts of the training model's parameter space and thus it is able to find the global solution to maximize the Q_3 function aiming at obtaining optimal parameters for the MOIT2FLS.

The GA training usually starts from a randomly initialized population and ends when it meets the determined stopping criteria. Since training process costs much time and is often trapped in local minima, the initialization of parameters is a nontrivial issue. In this paper, we utilize the Adaptive Vector Quantization (AVQ) clustering method [51] to identify the centres of IT2 Gaussian MFs in the antecedent part and the centres of interval T1 FSs in the consequent part. The well-separated distribution of the resulting clusters from the AVQ method is useful in identifying the allocation of fuzzy rules in the IT2 FLS.

We organize the corresponding input and output data into a unique observation of $p + 3$ dimensions where p is the number of inputs and three outputs corresponding to the three protein secondary structures. Denote x_i is the ith organized observation $(i = 1, \dots, N)$, x_i is presented as follows:

$$x_i = [input_i^1, input_i^2, \dots, input_i^p, output_i^1, output_i^2, output_i^3]$$

where $input_i^j$ is the jth input of the ith observation and $output_i^j$ is the output jth of the ith observation $(j = 1,2,3)$. By clustering the sample of N observations having the above format, we are able to derive the K resulting clusters corresponding with K fuzzy rules of the MOIT2FLS. Since the AVQ clustering is completed, centres of the resulting clusters are assigned to centres of the IT2 Gaussian MFs, which are employed in the antecedents of the fuzzy rules.

The centres of the output interval of each rule will be assigned equal to the output value of the corresponding cluster. The widths of the IT2 Gaussian MFs and the spreads of the output interval of each rule are initialized randomly but they are checked to ensure satisfying their corresponding constraints. Running the AVQ clustering a number of times equal to the GA population size, we are able to obtain the initial population for GA.

D. MOIT2FLS for PSSP

After assigning secondary structures to amino acid sequence, the amino acid sequence is parsed into learnable reasoning patterns using the local window moving strategy. An odd integer $n = 2m + 1$ where $m > 0$ is an integer is determined for the input window size. In this study, the window size is assigned to 17, which was found to be optimal in [3-5]. The output is represented by the triple where the helix, strand and coil structures are [1 0 0], [0 1 0] and [0 0 1] respectively.

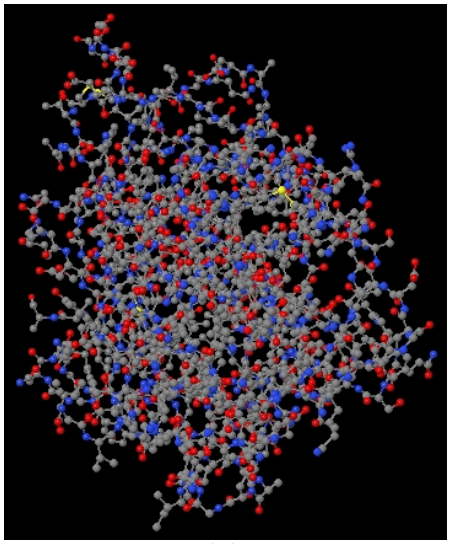
After clustering by the AVQ algorithm, the multi-output fuzzy rules can be constructed to build the MOIT2FLS, which then is trained using the GA with 20 evolving generations (termination condition). Once the MOIT2FLS has been trained, a new amino acid pattern can be put into the system to obtain values of three outputs. Values of three outputs are compared and applied the principle "winner-take-all" to determine the secondary structure of the central residue of the pattern.

V. Experimental Results

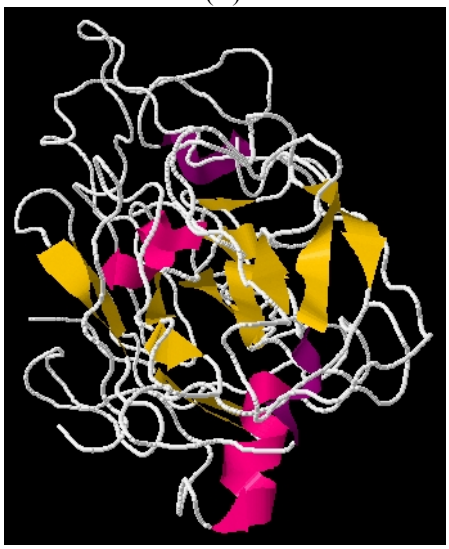
A. Datasets

The dataset in this study consists of 62 proteins which have been previously used in [5, 52]. Proteins with a number of residues greater than 200 are chosen for testing process (22 proteins presented in Table 2). The length of proteins are varied and summed up to more than 6300 residues in total. Three classes of secondary structures are distinguished including helices, sheets and coil. Helix and sheet types occupy 25% and 23% of the dataset respectively whilst the popular type is coil with 52%.

Fig. 2 shows the molecular view of the Acprotease protein and its secondary structure. Whilst the coil (in white) occupies the most proportion of the whole protein, the helix and strand are much less in pink and yellow.



(A)



(B)

Fig. 2. The molecule view of the Acprotease protein (A) and its secondary structure (B).

B. Results and Discussions

In order to have an unbiased comparison between NN and FS models, both are developed and trained 20 times. Then, the average Q3 values are calculated and reported in Table 2 along with results of the CF and GOR methods.

Fig. 3 shows the analogous variation of the CF, GOR, NN and FS performance in PSSP across 22 proteins. In more details, for a specific protein, if a method gains high prediction accuracy, the other methods almost do the same and vice versa. Take an example, in case of the Carboxypeptidase A (1cpa) protein, the FS attained the worst performance at 55.9% accuracy so do the other methods when NN, GOR and CF models also obtained the worst performance at 44.8%, 39.7% and 32.8% respectively. On the other hand, in case of the Hemoglobin (2mhb) protein, the FS reached the highest accuracy at 87% whilst NN, GOR and CF models also achieved relatively high accuracy, at 81.5%, 59.3% and 50% respectively.

Table 2. Q3 accuracy for testing dataset

No.	Proteins	iden	CF (%)	GOR (%)	NN (%)	FS (%)
1	Acprotease	1apr	40.32	51.61	72.58	79.03
2	Aproteinase	1app	45.90	67.21	60.66	67.30
3	Actinidin	2act	42.50	60.00	70.00	67.75
4	Arabinose binding	1abp	39.53	46.51	65.12	72.09
5	Beta trypsin	1ptn	56.10	65.85	65.08	75.26
6	Carbonic anhydrase C	1cac	60.42	54.17	70.83	67.48
7	Carboxypeptidase A	1cpa	32.76	39.66	44.83	55.94
8	Concanavalin A	3cna	52.27	61.36	70.45	72.55
9	Gamma trypsin A	2gch	47.73	59.09	75.00	68.27
10	Hemoglobin	2mhb	50.00	59.26	81.48	87.04
11	Lactate dehydrydrogenase	4lhd	46.03	58.73	55.56	61.56
12	Lambda Fab	1fab	37.80	52.44	63.41	68.29
13	Papain	8pap	38.46	71.79	74.36	69.67
14	Phophoglycerate mutase	3pgm	46.51	46.51	79.07	72.00
15	Subtilisin BPN	1sbt	36.54	53.85	51.92	61.74
16	Thermolysin	2tln	36.67	53.33	66.67	69.67
17	Tosyl elastase	1est	66.67	66.67	60.00	66.67
18	Triosephosphate isomerase	1tim	47.83	69.57	65.22	68.87
19	Glyceraldehyde dehydrogenase	1gpd	39.68	52.38	58.73	60.83
20	Alcohol dehydrogenase	4adh	47.22	55.56	56.94	60.08
21	Gluthatione reductase	2grs	40.45	58.43	44.94	57.93
22	Rhodanese	1rhd	63.64	70.91	66.86	69.27
			46.14	57.95	64.53	68.15

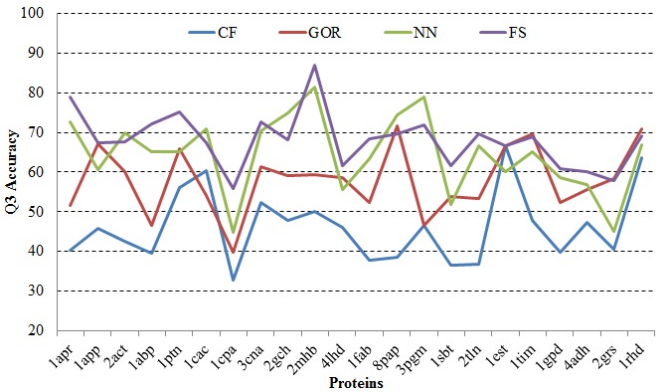


Fig. 3. Analogous variation of four PSSP models

It is obvious that the CF method has the worst performance compared to the other three investigated methods. On average the CF method just acquired the Q3 accuracy approximately 46.1%. This is understandable due to the limitation and simplicity of the CF method. The highest accuracy the CF method achieved occurs in the case of the Tosyl elastase (1est) protein with nearly 66.7% of the Q3 rate, which is the equivalent best compared to the GOR and MOIT2FLS method.

Notably the GOR method is much superior to the CF method because it utilizes the information theory in prediction calculation. The average Q3 accuracy of the GOR method is approximately 58%, which is 12% higher that of the CF method. In cases of the Papain (8pap), Triosephosphate isomerase (1tim), Gluthatione reductase (2grs) and Rhodanese (1rhd) proteins, the GOR method even works slightly better than the proposed MOIT2FLS. However, in the rest of the cases, it shows poorer performance compared to the MOIT2FLS.

The NN model reached the second best among the examined PSSP models with around 64.5% accuracy. Although in some proteins the FS is worse than the NN, but in most of the cases, the FS is higher than NN. In some proteins, FS can dominate the NN model up to over 10% of accuracy. For example, in case of the Gluthatione reductase (2grs) protein, the FS achieved the accuracy at 57.9% whilst the NN model is 13% lower than that of the FS, at just 44.9% accuracy. Alternatively, in case of the Carboxypeptidase A (1cpa) protein, the FS model outperforms the NN model up to 11%, 55.9% compared to 44.8% respectively. Furthermore, the FS model is also significantly superior to the NN model in cases of the Beta trypsin (1ptn) and Subtilisin BPN (1sbt) proteins, where the FS model respectively obtained 10.2% and 9.8% higher than those of the NN model.

VI. Conclusions

Amino acids are encoded using a new paradigm based on their quantitative properties including solvent exposed area, hydrophobicity, pK_a values of ionizing groups, and volume. This encoding scheme reduces the computational cost but on the other hand augments the uncertainty in the modelling. The fuzzy logic systems, type-2 in particular, are thus employed to handle the uncertainties. The MOIT2FLS is proposed herein for PSSP. Data patterns are organized to compose of 3 outputs, i.e. H, E or C corresponding to three structure classes. The AVQ clustering method is employed to derive multi-output fuzzy rules that are critical component of the MOIT2FLS. MOIT2FLS models are then trained through a deployment of GA. The MOIT2FLS aims to explore the relationships between sequence and structure due to the fact that the organization of amino acids relatively affects the secondary structure of residues. The dominance of the MOIT2FLS over the CF, GOR and NN models in PSSP resulted from two factors. First is the encoding scheme via amino acid properties rather than the traditional binary encoding, which is computationally costly. The proposed encoding scheme represents each amino acid by an array of size 7, which is much less than the traditional binary encoding with size of 20 for each amino acid. The second factor is the imprecise modelling capacity of type-2 fuzzy system, which has been introduced to overcome the limitation of type-1 fuzzy logic. The excellent capability of type-2 fuzzy logic in uncertainty handling, as widely adopted in the field of control system or engineering, has furthermore demonstrated in the field of computational biology as showcased in this paper.

Acknowledgment

This research is supported by the Australian Research Council (Discovery Grant DP120102112) and the Centre for Intelligent Systems Research (CISR) at Deakin University.

References

[1] A. J. F. Griffiths, W. M. Gelbart, J. H. Miller, and R. C. Lewontin, "*Modern Genetic Analysis"*, 7th edn. WH Freeman Publishers: New York, NY, 1999.

[2] P. Y. Chou and G. D. Fasman, "Empirical predictions of protein conformation", *Annual Review of Biochemistry*, vol. 47, pp. 251-276, 1978.

[3] J. Garnier, D. J. Osguthorpe, and B. Robson, "Analysis of the accuracy and implications of simple methods for predicting the secondary structure of globular proteins", *Journal of Molecular Biology*, 120 (1), 97-120, 1978.

[4] N. Qian and T. J. Sejnowski, "Predicting the secondary structure of globular proteins using neural network models", *Journal of Molecular Biology*, *202*(4), 865-884, 1988.

[5] L. H. Holley and M. Karplus, "Protein secondary structure prediction with a neural network", *Proceedings of the National Academy of Sciences*, *86*(1), 152-156, 1989.

[6] F. Bettella, D. Rasinski, and E. W. Knapp, "Protein secondary structure prediction with SPARROW", *Journal of Chemical Information and Modeling*, *52*(2), 545-556, 2012.

[7] E. Faraggi, T. Zhang, Y. Yang, L. Kurgan, and Y. Zhou, "SPINE X: Improving protein secondary structure prediction by multistep learning coupled with prediction of solvent accessible surface area and backbone torsion angles", *J. Comput. Chem.*, 33: 259-267, 2012.

[8] M. H. Zangooei and S. Jalili, "Protein secondary structure prediction using DWKF based on SVR-NSGAII", *Neurocomputing*, *94*, 87-101, 2012.

[9] Y. Wei, J. Thompson, and C. A. Floudas, "CONCORD: a consensus method for protein secondary structure prediction via mixed integer linear optimization", *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, 468(2139), 831-850, 2012.

[10] S. Saraswathi, J. L. Fernández-Martínez, A. Koliński, R. L. Jernigan, and A. Kloczkowski, "Fast learning optimized prediction methodology (FLOPRED) for protein secondary structure prediction", *Journal of Molecular Modeling*, *18*(9), 4275-4289, 2012.

[11] J. K. Leman, R. Mueller, M. Karakas, N. Woetzel, and J. Meiler, "Simultaneous prediction of protein secondary structure and transmembrane spans", *Proteins: Structure, Function, and Bioinformatics*, 81, 1127-1140, 2013.

[12] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, pp. 338-353, 1965.

[13] G. Mocz, "Fuzzy cluster analysis of simple physicochemical properties of amino acids for recognizing secondary structure in proteins," *Protein Science*, 4, 1178-1187, 1995.

[14] J. Boberg, T. Salakoski, and M. Vihinen, "Accurate prediction of protein secondary structural class with fuzzy structural vectors," *Protein Engineering*, *8*(6), 505-512, 1995.

[15] J. A. Hering, P. R. Innocent, and P. I. Haris, "Neuro-fuzzy structural classification of proteins for improved protein secondary structure prediction," *Proteomics*, 3, 1464-1475, 2003.

[16] J. M. Mendel, "Type-2 fuzzy sets and systems: An overview," *IEEE Computational Intelligence Magazine*, vol. 2, pp. 20-29, 2007.

[17] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning, Part I.," *Information Science*, vol. 8, pp. 199-249, 1975.

[18] Q. Liang and J. M. Mendel, "Interval type-2 fuzzy logic systems: Theory and design," *IEEE Transactions on Fuzzy Systems*, vol. 8, pp. 535-550, 2000.

[19] J. Kyte and R. F. Doolittle, "A simple method for displaying the hydropathic character of a protein," *Journal of Molecular Biology*, *157*(1), 105-132, 1982.

[20] W. C. Wimley and S. H. White, "Experimentally determined hydrophobicity scale for proteins at membrane interfaces," *Nature Structural Biology*, *3*(10), 842-848, 1996.

[21] C. C. Palliser and D. A. Parry, "Quantitative comparison of the ability of hydropathy scales to recognize surface β-strands in proteins," *Proteins: Structure, Function, and Bioinformatics*, *42*(2), 243-255, 2001.

[22] D. W. Mount, "*Bioinformatics: Sequence and genome analysis"*. Cold Spring Harbour Laboratory Press: Cold Spring Harbour, 2nd, 2004.

[23] T. Hessa, H. Kim, K. Bihlmaier, C. Lundin, J. Boekel, H. Andersson, I. Nilsson, S. H. White, and G. von Heijne, "Recognition of transmembrane helices by the endoplasmic reticulum translocon," *Nature*. 2005 Jan 27; 433(7024):377-81, supplementary data.

[24] D. L. Nelson, A. L. Lehninger, and M. M. Cox, "*Lehninger Principles of Biochemistry"*. New York: W.H. Freeman, 2008.

[25] W. R. Forsyth, J. M. Antosiewicz, and A. D.Robertson, "Empirical relationships between protein structure and carboxyl pK_a values in proteins," *Proteins: Structure, Function, and Bioinformatics*, *48*(2), 388-403, 2002.

[26] T. Hamelryck, "An amino acid has two sides: a new 2D measure provides a different view of solvent exposure", *Proteins: Structure, Function, and Bioinformatics*, 59(1), 38-48, 2005.

[27] E. Durham, B. Dorr, N. Woetzel, R. Staritzbichler, and J. Meiler, "Solvent accessible surface area approximations for rapid and accurate protein structure prediction", *Journal of Molecular Modeling*, 15(9), 1093-1108, 2009.

[28] P. Li, G. Pok, K. S. Jung, H. S. Shon, and K. H. Ryu, "QSE: A new 3-D solvent exposure measure for the analysis of protein structure", *Proteomics*, 11(19), 3793-3801, 2011.

[29] C. Schaefer and B. Rost, "Predict impact of single amino acid change upon protein structure", *BMC Genomics*, 13(Suppl 4), S4, 1-10, 2012.

[30] W. Kabsch and C. Sander, "Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features," *Biopolymers*, *22*(12), 2577-2637, 1983.

[31] D. Frishman and P. Argos, "Knowledge-based protein secondary structure assignment," *Proteins: Structure, Function, and Bioinformatics*, *23*(4), 566-579, 1995.

[32] F. M. Richards and C. E. Kundrot, "Identification of structural motifs from protein coordinate data: Secondary structure and first-level supersecondary structure," *Proteins: Structure, Function, and Bioinformatics*, *3*(2), 71-84, 1998.

[33] N. N. Karnik and J. M. Mendel, "Operations on type-2 fuzzy sets," *Fuzzy Sets and Systems*, vol. 122, pp. 327-348, 2001.

[34] R. John, and S. Coupland, "Type-2 fuzzy logic: A historical view," *IEEE Computational Intelligence Magazine*, vol. 2, pp. 57-62, 2007.

[35] J. M. Mendel, R. I. John, and F. Liu, "Interval type-2 fuzzy logic systems made simple," *IEEE Transaction Fuzzy Systems*, vol. 14, no. 6, pp. 808-821, 2006.

[36] N. N. Karnik, and J. M. Mendel, "Type-2 fuzzy logic systems," *IEEE Transactions on Fuzzy Systems*, vol. 7, pp. 643-658, 1999.

[37] J. M. Mendel and R. I. B. John, "Type-2 fuzzy sets made simple," *IEEE Transactions on Fuzzy Systems*, vol. 10, no. 2, pp. 117-127, 2002.

[38] A. Khosravi, S. Nahavandi, and D. Creighton, "Short term load forecasting using interval type-2 fuzzy logic systems," *Proceedings of the IEEE International Conference on Fuzzy Systems*, pp. 502-508, 2011.

[39] N. Boumella, K. Djouani, M. Boulemden, "A robust interval type-2 TSK fuzzy logic system design based on chebyshev fitting," *International Journal of Control, Automation and Systems*, vol. 10, no. 4, pp. 727-736, 2012.

[40] A. Khosravi, S. Nahavandi, D. Creighton, and D. Srinivasan, "Interval type-2 fuzzy logic systems for load forecasting: A comparative study," *IEEE transactions on Power Systems*, vol. 27, no. 3, pp. 1274-1282, 2012.

[41] T. Nguyen, A. Khosravi, S. Nahavandi, & D. Creighton, "Neural network and interval type-2 fuzzy system for stock price forecasting," *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pp. 1-8, 2013.

[42] N. N. Karnik and J. M. Mendel, "Centroid of a type-2 fuzzy set," *Information Sciences*, vol. 132, pp. 195-220, 2001.

[43] J. M. Mendel, "*Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions,*" Upper Saddle River, NJ: Prentice-Hall, 2001.

[44] D. Wu and J. M. Mendel, "Enhanced Karnik-Mendel algorithms," *IEEE Transactions on Fuzzy Systems*, vol. 17, no. 4, pp. 923-934, 2009.

[45] C. Y. Yeh, W. H. Jeng, and S. J. Lee, "An enhanced type-reduction algorithm for type-2 fuzzy sets," *IEEE Transactions on Fuzzy Systems*, vol. 19, no. 2, pp. 227-240, 2011.

[46] D. Wu and M. Nie, "Comparison and practical implementation of type-reduction algorithm for type-2 fuzzy sets and systems," in *Proceedings of IEEE International Conference on Fuzzy Systems*, Taipei, Taiwan, pp. 2131-2138, June 2011.

[47] K. Duran, H. Bernal, and M. Melgarejo, "Improved iterative algorithm for computing the generalized centroid of an interval type-2 fuzzy set," in *Proceedings of North American Fuzzy Information Processing Society*, New York, pp. 1-5, May 2008.

[48] J. H. Holland, "*Adaptation in Natural and Artificial Systems,*" University of Michigan Press, Ann Arbor, 1975.

[49] D. E. Goldberg, "*Genetic algorithms in Search, Optimization, and Machine Learning,*" Addison Wesley, Massachusetts, USA, 1989.

[50] C. R. Reeves and J. E. Rowe, "*Genetic Algorithms: Principles and Perspectives: A Guide to GA Theory*", Kluwer Academic Publishers, 2002.

[51] B. Kosko, "*Neural Networks and Fuzzy Systems,*" Prentice-Hall, Englewood Cliffs, NJ, 1991.

[52] W. Kabsch and C. Sander "How good are predictions of protein secondary structure?", *FEBS Lett* 155 (2): 179-82, 1983.