

# Protein Structure Prediction Using Hybrid Neural Network and Fuzzy Inference System

Yongxian Wang, Zhenghua Wang  
School of Computer, National University of Defense Technology,  
410073 Changsha, China  
yongxian\_wang@yahoo.com

Xiaomei Li  
College of Command and Technology of Equipment,  
101416 Beijing, China  
lxmcjh@sohu.com

## Abstract

This work presents a method based on an adaptive neuro-fuzzy inference system (ANFIS) for modeling protein secondary structure prediction which aims at acquiring the unknown structure information of target protein directly from its sequence data which is available. The number of input variables and inference rules are commonly too large, sometimes even huge, to make the model building feasible. To overcome these defects a two-phase process is employed in our model. In the first phase, the selection of number and position of the fuzzy sets of initial input variables can be determined by employing a fuzzy clustering algorithm; and in the second phase the more precise structural identification and optimal parameters of the rule-base of the ANFIS are achieved by an iterative GA updating algorithm. An experiment on three-state secondary structure prediction of protein is reported briefly and the performance of the proposed method is evaluated. The results indicate an improvement in design cycle and convergence to the optimal rule-base within a relatively short period of time, however, at the cost of little decrease in accuracy.

**keyword:** *bioinformatics, protein structure prediction, ANFIS*

## 1 Introduction

The release of the complete human genome sequence in early 2001 was a milestone event that marked the transition of modern biology into a new “post

genome” era. The application of information extraction and knowledge discovery to protein data in biology and life science is increasing in recent years. It is now common knowledge that the 3D structure of a protein can provide valuable information as to its function and mechanism. How to acquire the structure information of a target protein from its primary structure, that is, amino acid sequence, is one of hot spots in this research field. Protein secondary structure prediction is an intermediate step in the prediction of tertiary structure form amino acid sequence.

In the literature, models for protein structure prediction are classified into two main categories [1]. The first class of methods, *de novo* or *ab initio* methods, predict the structure from sequence alone, without relying on similarity at the fold level between the target sequence and those of the known structures. The second class of protein structure prediction methods mainly rely on known structural information instead. In this class of method, many statistical models are developed in recent years. ANNs are applied for analysis of primary protein sequences and modeling structure prediction and classification [11, 3, 8]. The defect of this approach mainly lies in that the built model has no clear meaning which can be explained using domain related knowledge. Both the connection weights and the activation functions in ANNs can hardly be expressed as the form of natural language which is easily understood by human.

Fuzzy set theory allows the use of linguistic concepts for representing quantitative values. Moreover, fuzzy systems can express the process of human reasoning by expressing relationships between concepts as rules. Several methods for identification of fuzzy models have been reported in the literature ([6] among others). Adaptive neuro-fuzzy inference systems (ANFIS) is one of the most popular types of fuzzy neural networks, which combines the advantages of fuzzy system and neural network, in modeling non-linear control system [5]. However, ANFIS’s inputs and rules must be preliminarily given by human experts before the learning process. For a real-world modeling problem, like protein structure prediction, it is common to have tens of potential inputs and more inference rules to the model under ANFIS construction. Without sufficient domain-related knowledge, it is hard to select inputs and to generate appropriate rules, except by using a trial-and-error method. To surmount these difficulties, the methods of selecting input variables and extracting fuzzy rules from original data have attracted many researchers to make efforts. Jang present a quick and straightforward way of input selection for ANFIS learning [7], and in [13], Sugeno and Yasukawa adopted a fuzzy

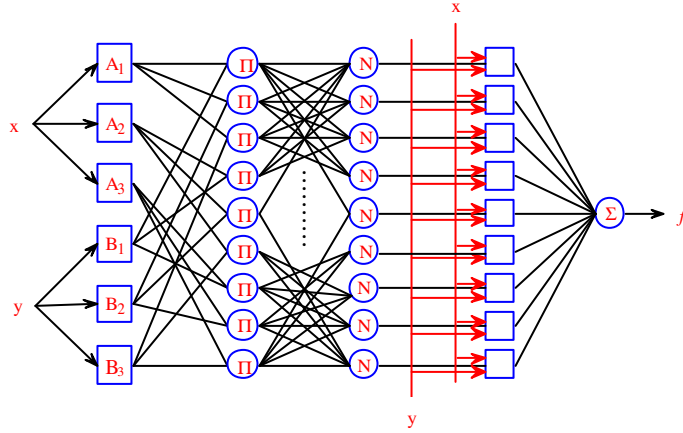


Figure 1: ANFIS architecture of first-order Sugeno fuzzy model with 2-input, 1-output using grid partition of input space [7]

clustering method, namely, fuzzy c-means, to identify the structure of a fuzzy model and extract the control rules by its input-output (I/O) data. In 1999, Fahn, Chin-Shyurng et al discussed a fuzzy rules generation method based on evolutionary algorithms (EA) and multilayer perceptions (MLP) [4].

In this paper, we propose a novel approach which has a two-phase processing, input selection phase and following fuzzy inference rule generation phase. A fuzzy clustering method is employed in the first phase to screen and choose appropriate indices in hundreds of physicochemical and biological properties of amino acids, and to select the inputs of the following generated rules. In the second phase, we use a method of optimization based on GAs, which result in a reasonable number of inference rules for training later.

This paper is organized with the following fashion. In the remainder of this section, we briefly introduce the concept of ANFIS and GA, while in Sect. 2 the method of input selection and rule generation in our model is described respectively. Application to the problems of protein secondary structure prediction and its experimental results are demonstrated in Sect. 3. Finally Sect. 4 gives concluding remarks.

## 1.1 ANFIS

The Sugeno neuro-fuzzy model was proposed in an effort to formalize a systematic approach to generating fuzzy rules from an input-output data set [13]. A typical fuzzy rule in a Sugeno fuzzy model has the format

$$\text{If } x_1 \text{ is } A_1 \text{ and } x_2 \text{ is } A_2, \dots, x_n \text{ is } A_n \text{ then } y = f(x_1, x_2, \dots, x_n)$$

where  $x_i$ s and  $y$  are input and output variables respectively,  $A_i$ s are fuzzy sets in the antecedent, and  $f$  is a crisp function in the consequent.

To simplify the model, Jang J.-S Roger proposed a five-layer network architecture ANFIS (Adaptive Neuro-Fuzzy Inference System) to put the fuzzy model into the framework of adaptive networks that can compute gradient vectors systematically. Fig. 1 illustrates graphically this reasoning mechanism, more details can be found in [7].

## 1.2 GAs

Genetic algorithms are search algorithms base on the mechanics of natural selection and natural genetics. Unlike many classical optimization techniques, genetic algorithms do not rely on computing local derivatives to guide the search process. GAs generally consist of three fundamental operators: reproduction, crossover and mutation. Given an optimization problem, GAs encode the parameters into finite bit strings, and then run iteratively using these three operators in a random way but based on the fitness function evolution to achieve the basic tasks. Readers may refer to [10] for more details about GAs.

GAs differ fundamentally from the conventional search techniques. For example, they (1) consider a population of points, not a single point, (2) use probabilistic rules to guide their search, no deterministic rules, (3) GAs include random elements, which help to avoid getting trapped in local minima (4) Unlike many other optimization methods, GAs do not require derivative information and complete knowledge of the problem structure and parameters.

## 2 Methods

As mentioned earlier, a two-phase process is performed in modeling for protein structure prediction.

As the first step, the input variables of ANFIS must be selected carefully because modeling such a problem as protein structure prediction will involve tens of hundreds of potential inputs, and we need to have a way to quickly determine the priorities of these potential inputs and use them accordingly.

The rule generation and updating based on GAs is the main task of second phase in our modeling. In [12] GAs are employed to optimize the fuzzy set and the shape and type of MFs, however, the conflict between the generated rules can occur during the iterative optimization procedure. Recently Yan Wu [15] proposed a new similar method but the consequent of inference rule is restricted within a specific form. The method proposed here amends these drawbacks and optimize all the parameters and structures of the neuro-fuzzy system simultaneously. During the optimization the redundancy and conflict between temporarily generated rules can be cut off dynamically.

## 2.1 Feature Extraction

In the problem of protein secondary structure prediction, given a primary sequence of the target protein (i.e. the sequence of amino acid), we hope the final well-constructed model can assign a secondary structure type for each residue in the target sequence. In general, which secondary structure type should be assigned for a residue relies upon its position in the sequence, the types of neighbor residues and their physicochemical and biological properties.

The database AAindex (Amino Acid Index Database) is a collection of published indices of different physicochemical and biological properties of amino acids. Its AAindex1 section currently contains 494 indices [9]. A initial set containing 120 indices from AAindex database is carefully chosen by hand in our model, which covers the secondary structure propensities, the bias and hydrophobicity of every residue, and other physicochemical properties. A normal cluster analysis using the correlation coefficient as the distance between two indices results in a smaller set which only remains 32 indices. Those whose similarities are less than 0.85 are excluded such that only one representative index remains.

A followed normalization and fuzzification procedure is performed to make the range of each index lie in interval  $[0, 1]$  and to make the modeling in a fuzzy style. We classify the domain of each normalized index into three fuzzy sets, denote “high”, “midterm” and “low” value, respectively. Three different membership functions are adopted illustrated as Fig. 2.4, each has

three adjustable parameters to control its shape and range.

## 2.2 Training Data Selection

A high quality data set extracted from the Brookhaven Protein Data Bank (PDB) is used in our model [2]. We include entries which match the following: (1) Those whose structure are determined by X-ray diffraction, since their high-quality in measure. (2) Those program DSSP can produce an output which included in PDBFind database, since we want to use the PDBFind's assignment of protein secondary structure. (3) The protein that physical chain has no break, in PDB, a minus sign ("-") indicates a break in the sequence of chain. (4) Chains with a length of greater than 80 amino acids.

To avoid the disturbance of redundant information, a representative subset is selected from the extracted set of chains above, which contains entries included in PIR Non-Redundant Reference Protein Database (PIR-NREF) [14]. Furthermore, a smaller data subset consisting of 1000 distinct protein chains is picked out as train data and check data in our experiment described in section 3.

## 2.3 Fuzzy Cluster Analysis of Inputs

The architecture of ANFIS used in our experiment takes as input a local  $N$ -mer (i.e. fixed-size window) of amino acids (the typical window width  $N = 17$ ), centered around the residue for which the secondary structure is being predicted. This approach has been widely applied and proven to be quite successful in early work. However, we do not take directly the residues itself in the  $N$ -mer as the input variables, but as their certain attributes, thus more information is considered about the environment of the target residue. A problem in such a model is that the huge size of inputs will handicap the model construction and make the underlying model less concise and transparent.

For instance, if we use a 17-mer window, each residue has 32 attributes derived from selected indices mentioned earlier, then the total input of ANFIS will up to 544 ( $= 17 \times 32$ ). Furthermore, if we employ a input-space grid partitioning and assign 3 fuzzy sets for each input in the following training phase, the number of candidate if-then rules will reach  $3^{544}$  (an order of  $10^{259}$ ) ! It makes the training procedure painful, if not impossible.

A clustering strategy is used to overcome this problem, and the candidate inputs are divided into groups and one or several members of each group has to be in the set of final inputs to the model under consideration. Specifically, given a fixed number of groups, say,  $K$ , we use a fuzzy  $c$ -means clustering technique to fulfill this task. The method is as follows:

1. Chose the initial representation of cluster centers. Firstly, estimate the entropy of the joint distribution of each dual group (position in N-mer, attribute class) with respect to the appropriate type of secondary structure (for instance,  $\alpha$ -helix,  $\beta$ -sheet, coil, etc) using a statistical method. Then we select the least  $K$  value as the candidate cluster centers of  $K$  groups.
2. Use standard fuzzy  $c$ -means clustering algorithm to divide the whole input-space into  $K$  subspace according to the initial centers chosen in the preceding step.
3. Depending on the complexity of modeling, one or more elements get the chance to enter the set of final inputs.

Virtually, if only one representative member is chosen from every group in the last step described above, a pretty effective candidate is the initial group center, and the fuzzy clustering step can be omitted as a result accordingly.

## 2.4 Encoding Template

The encoding of whole neuro-fuzzy system can be categorized two main parts, encoding fuzzy rules and encoding the MFs of fuzzy set.

Three type of fuzzy sets are used for indicating “low”, “medium” and “high” respectively. Fig. 2.4 illustrates their MFs, and each with three control parameters: the center of MFs  $m$  and the interval range of target variable  $[a, b]$ . The fuzzy if-then rule has the form described in Sect. 1.1 and the architecture of ANFIS has be shown in Fig. 1.

For the convenience of the following process, we treat the 2-dimension input variable (position in N-mer, attribute class) as a flat one-dimension vector as the input of ANFIS shown in Fig. 1, by rearranging it in a either row-first or column-first way. In the antecedent of each fuzzy rule, digit 1, 2 and 3 denote the fuzzy set “low”, “medium” and “high” respectively, and a digit zero indicates the fuzzy set is non-existent. When consider the three

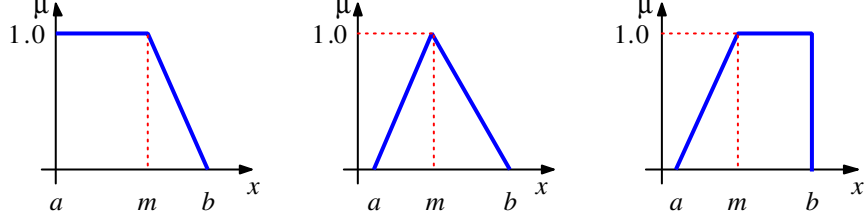


Figure 2: Membership functions of fuzzy set used in our modeling protein secondary structure prediction, which describe “low”, “medium”, and “high” respectively.

class of secondary structure (e.g.  $\alpha$ -helix,  $\beta$ -sheet and coil), they can be denoted by digit 1, 2 and 3 respectively. For instance, the integer string “**1 0 3 2 1**  $p_0 p_1 p_2 p_3 p_4$ ” will encoding the rule “if  $x_1$  is low and  $x_3$  is high and  $x_4$  is medium, then  $y = p_0 + \sum_{i=1}^4 p_i x_i$ ”. Where  $p_i$ s are parameters in the consequent of rule, and encoded in a floating point number.

Encoding the MFs of fuzzy set concerns the center of the MFs, the span of triangular MFs, the overlaps between two adjacent MFs, etc. All these parameters are encoded into decimal integer strings in the gene and every gene takes value in  $\{0, 1, 2, \dots, 9, 10\}$ .

## 2.5 Learning Procedure

Let  $x$  be the variable to be concerned,  $x \in [a, b]$ , a dynamic interval. There are  $n$  fuzzy sets about variable  $x$  ranged on  $[a, b]$ . Initially the center of the  $i$ -th MF locates at

$$m_i = a + i * h \quad i = 1, 2, \dots, n \quad h = (b - a)/(n + 1).$$

During the iteration, the center of MF, the right-bottom corner of triangular MF, the factor of overlap of two adjective MFs and the left-bottom corner of triangular MF can be updated by equation (1)-(4) respectively:

$$m_i \leftarrow m_i + \omega * (c_j - 5), \quad (1)$$

$$b_i \leftarrow m_i + (m_{i+1} - m_i) * (1 - c_j/10) + \Delta \quad (2)$$

$$\delta_{i,i+1} \leftarrow 0.3 + (1 - c_j/10) * (0.8 - 0.3) \quad i = 1, \dots, n - 1 \quad (3)$$

$$a_i \leftarrow b_{i-1} - \frac{\delta_{i-1,i}(m_i - m_{i-1})}{1 - \delta_{i-1,i}} \quad i = 2, \dots, n \quad (4)$$

where  $c_j$  denote the value of  $j$ -th gene in the chromosome,  $\Delta$  and  $\omega$  are some small adjustable variable ( $\simeq 0.01$  in practice) as compensates for  $b_i$ s and  $a_i$ s respectively. The completeness of MFs in the system can be guaranteed [7].

The common genetic operation can be applied here. However, the crossover operation must be carefully processed according our encoding manner for each fragment of gene in the chromosome has a fixed range. Given a pair of genes, a normalizing process takes precedence of the crossover operation to guarantee the gene value still in an appropriate range. Moreover, the structure of neuro-fuzzy system can be optimized dynamically during the iteration. If the fitness increase very slowly in the optimization and updating the parameters takes little effect, then an additional fuzzy rule with the random initial parameters will be added and be joined in the GAs. We record the occurrence of activation for each rule in the learning, and if the occurrence equals to zero, which indicates that rule has no use in the rule sets and can remove it safely. However, those who have the same antecedent of rule but a different consequent should remain to improve the performance of the system.

The learning algorithm whose framework is proposed by Yan Wu ([15]) can be summarized as follows.

```

initialize the number of MFs and fuzzy rules.
initialize the parameters of GA, include:
    initial population P(t)(t=0),
    initial fitness (best-fit(0)=0),
    terminate condition,
    probabilities of crossover and mutation.
evaluate the fitness P(0).
while (not terminate condition)
    t = t + 1;
    select P(t) from P(t-1) in a roulette wheel way;
    perform crossover and mutation to give new improved population;
    if ( fitness of P(1) > best-fit(t) ) then
        best-fit(t) = fitness of the best chromosome;
        adjust the parameters according the last subsection;
        if ( best-fit(t) is too small and increase too slowly ) then
            add a new rule;
            initialize the new parameters and generate a new P(t);
        end if

```

```
end if
end while
remove the redundant rules.
```

### 3 Results and Discussions

The data set in our experiment is extracted as described in Sect. 2.2, the initial candidate inputs of ANFIS are organized into groups of 17-mer fragment of amino acid (a local window with width 17 residue). Each residue position is represented as 32 attributes which describe the physicochemical and biological properties of the corresponding amino acid. In the input selection phase, 50 inputs are chosen among 544 initial candidate inputs in the form of vectors transformed from original two-dimension input matrix (row for position in 17-mer and column for index of residue). Even now there are too much possible combination in the set of rules if grid partitioning is employed in the input space. So in the following process, rule generation approach described in last section are applied. A cross-validation technique is employed in this process, wherein training data include 900 entries derived from PDB as explained earlier and the remainder as the check data. We set the initial number of MFs for each input variable be 3 as shown in Fig. 2.4, and the number of rules be 100. After a time-consuming optimization iteration (500 epochs) described in last section, the final prediction neuro-fuzzy model is built which contains 52 entries of inference rule. Defuzzification of the consequent of the resulting rule has been shown that nearly a half of the output is “ $\alpha$ -helix”.

Using such a model, an  $MSE = 0.1028$  is obtained for the train data and an  $MSE = 0.0935$  for the test data. The accuracy of the ensemble three-class prediction can up to 69.7%, more specifically, the accuracy of  $\alpha$ -helix,  $\beta$ -sheet are 74.1% and 67.3%, and the performance is a little lower than that of model built earlier in our research group. The details of result data and analysis will be given in another paper.

In another experiment we divided the problem into three sub-model, and each sub-model is used to predict only one class of protein secondary structure, that is, one for  $\alpha$ -helix, the other for  $\beta$ -sheet and a third one for coil. The final prediction has been determined from the outputs of three sub-model in a committee mechanism. However, the performance increases just a little more, 0.7% for  $\alpha$ -helix and 1.3% for  $\beta$ -sheet.

## 4 Conclusions

This paper has described a method of hybrid neural network and fuzzy system. The main characteristics of the method is a two-phase processing technique in an ANFIS. An experiment on three-class secondary structure prediction of protein using this method is reported and the results indicate the method proposed has advantages of comprehensibility, high precision, good generalization and rapid convergence in rule generation. Compared with the traditional method of the same application, this method has a little decline of accuracy in prediction task. Future works include the use of a hybrid of gradient descent algorithms and the GAs for further optimization of parameters in consequent of if-then inference rules.

## Acknowledgements

This work is supported partially by the National Natural Science Foundation of China (NSFC) under grant: 69933030.

## References

- [1] David Baker and Andrej Sali. Protein structure prediction and structural genomics. *Science*, 294(5540):93–96, 2001.
- [2] H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, I.N. Shindyalov, and P.E. Bourne. The protein data bank. *Nucleic Acids Research*, 28:235–242, 2000.
- [3] Chris H.Q. Ding and Inna Dubchak. Multi-class protein fold recognition using support vector machines and neural networks. *Bioinformatics*, 17(4):349–358, 2001.
- [4] Chin-Shyurng Fahn, Kou-Torng Lan, and Zen-Bang Chern. Fuzzy rules generation using new evolutionary algorithms combined with multilayer perceptrons. *IEEE Transactions on Industrial Electronics*, 46(6):1103–1113, 1999.

- [5] Jyh-Shing Roger Jang. ANFIS: Adaptive-network-based fuzzy inference system. *IEEE Transactions on Systems, Man and Cybernetics*, 23(0018-9472):665–685, 1993.
- [6] Jyh-Shing Roger Jang. Neuro-fuzzy modeling for dynamic system identification. In *Fuzzy Systems Symposium, 1996. 'Soft Computing in Intelligent Systems and Information Processing'*, *Proceedings of the 1996 Asian*, pages 320–325, 1996.
- [7] Jyh-Shing Roger Jang, Chuen-Tsai Sun, and Eiji Mizutani. *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*. Prentice Hall, 1997.
- [8] Harpreet Kaur and G.P.S. Raghava. An evaluation of  $\beta$ -turn prediction methods. *Bioinformatics*, 18(11):1508–1514, 2002.
- [9] S. Kawashima and M. Kanehisa. AAindex: Amino acid index database. *Nucleic Acids Research*, 28:374, 2000.
- [10] Melanie Michell. *An Introduction to Genetic Algorithms (Complex Adaptive Systems)*. MIT Press, 1998.
- [11] Burkhard Rost and Chris Sander. Improved prediction of protein secondary structure by use of sequence profiles and neural networks. *Proceedings of the National Academy of Sciences of the USA*, 90(16):7558–7562, 1993.
- [12] Yuhui Shi, R. Eberhart, and Yaobin Chen. Implementation of evolutionary fuzzy systems. *IEEE Transactions on Fuzzy Systems*, 7(1063-6706):109–119, 1999.
- [13] T. Sugeno, M.; Yasukawa. A fuzzy-logic-based approach to qualitative modeling. *IEEE Transactions on Fuzzy Systems*, 1(1):7–31, 1993.
- [14] Cathy H. Wu, Hongzhan Huang, Leslie Arminski, *et al.* The protein information resource: an integrated public resource of functional annotation of proteins. *Nucleic Acids Research*, 30:35–37, 2002.
- [15] WU Yan. A novel method of fuzzy inference neural network optimization based on GA. *Computer Engineering*, 28:23–25, 2002.