

Payam S. RAHMDEL, Minh Nhut NGUYEN, Liying ZHENG

# Optimization of fuzzy CMAC using evolutionary Bayesian Ying-Yang learning

© Higher Education Press and Springer-Verlag Berlin Heidelberg 2011

**Abstract** Cerebellar model articulation controller (CMAC) is a popular associative memory neural network that imitates human's cerebellum, which allows it to learn fast and carry out local generalization efficiently. This research aims to integrate evolutionary computation into fuzzy CMAC Bayesian Ying-Yang (FCMAC-BYY) learning, which is referred to as FCMAC-EBYY, to achieve a synergetic development in the search for optimal fuzzy sets and connection weights. Traditional evolutionary approaches are limited to small populations of short binary string length and as such are not suitable for neural network training, which involves a large searching space due to complex connections as well as real values. The methodology employed by FCMAC-EBYY is coevolution, in which a complex solution is decomposed into some pieces to be optimized in different populations/species and then assembled. The developed FCMAC-EBYY is compared with various neuro-fuzzy systems using a real application of traffic flow prediction.

**Keywords** cerebellar model articulation controller (CMAC), Bayesian Ying-Yang (BYY) learning, evolutionary computation

## 1 Introduction

Fuzzy logic incorporated into neural network is capable to express knowledge of domain experts, as well as to

deal with imprecision and nonlinearity. The cerebellar model articulation controller (CMAC) system [1,2] is a type of associative memory neural network that imitates human's cerebellum, which allows it to learn fast and carry out local generalization efficiently. However, CMAC suffers from its inherent disadvantage of inefficiency in data storage, because the number of quantized states in each dimension makes the memory size grow exponentially with respect to the number of input variables [3]. Instead of using crisp states in the original CMAC, fuzzy CMAC (FCMAC) uses fuzzy sets (fuzzy labels) as the input clusters, which greatly reduce the memory requirement and, in the mean time, provides a human-like thinking ability.

First proposed in Ref. [4] and systematically developed more than one decade, Bayesian Ying-Yang (BYY) learning provides not only a general framework that accommodates typical learning algorithms from a unified perspective but also a new road that leads to improved model selection criteria and Ying-Yang alternative learning with automatic model selection. As summarized in Fig. A2 of Ref. [5], BYY learning consists of two subcategories. One is featured with Ying-Yang best matching for developing typical learning algorithms, which is considered in this paper and also one major focus of Ref. [4], while the other is featured with its favorable nature for model selection [6,7], for which readers are referred to recent systematic review [5], and especially its Fig. A2 for the relations among the BYY learning subcategories.

In our previous work, BYY learning [4–7] is applied in the fuzzification layer of the FCMAC structure, thereafter, referred to as FCMAC-BYY [8,9], to create the fuzzy sets out of the input data. However, the FCMAC-BYY suffers from two problems: First, as the BYY-based fuzzification is separated from the weight training of the neural network, it may end up with local optima. Second, the weights are trained by using gradient-based methods, which may also lead to local optima.

To address the above-said problems, evolutionary computation is introduced in this research to utilize

---

Received July 7, 2010; accepted February 28, 2011

---

Payam S. RAHMDEL (✉)  
School of Engineering and Information Sciences, Middlesex University, London NW4 4BT, United Kingdom  
E-mail: P.rahmdel@mdx.ac.uk

Minh Nhut NGUYEN  
Institute for Infocomm Research, Singapore 138632, Singapore

Liying ZHENG  
School of Computer Science and Technology, Harbin Engineering University, Harbin 150001, China

global search for both the optimal fuzzy clusters and the optimal weights of FCMAC-BYY. The milestone of evolutionary computation (EC) dates back to 1975, when John Holland proposed genetic algorithms (GAs) by simulating the principle of Darwinian natural selection and evolution [10]. This is achieved by maintaining and evolving a population of candidate solutions, referred to as chromosomes, or called individuals, which undergoes a process of evaluation, selection, reproduction (crossover and mutation), and reinsertion [11].

Traditional evolutionary approaches are limited to small populations of short binary string length, whereas neural network training involves a large searching space due to complex connections. One of the solutions to applying EC to neural network training is coevolution, in which a complex solution is decomposed into some pieces to be optimized in different populations/species and then assembled [12]. Typically, the coevolutionary algorithms can be classified into two types, namely, competitive coevolution [13] and cooperative coevolutionary [14].

This research aims to incorporate coevolutionary computation into the FCMAC-BYY to search for the optimal fuzzy sets as well as the connection weights. The paper is organized as follows: Section 2 describes the work mechanism of the original FCMAC-BYY. The details of the coevolutionary learning incorporated with FCMAC-BYY are given in Sect. 3. Experimental results are presented and compared in Sect. 4, followed by our conclusion and discussion in Sect. 5.

Leveled equations, graphics, and tables are not prescribed, although the various table text styles are provided. The formatter will need to create these components, incorporating the applicable criteria that follow.

## 2 Work mechanism of FCMAC-BYY

FCMAC-BYY network is a five-layer hierarchical structure, namely, input layer, fuzzification layer, association layer, post association layer, and output layer [13].

### 2.1 Architecture of FCMAC-BYY

The input to FCMAC-BYY is a non-fuzzy data vector corresponding to a measure of the input parameter represented in the respective dimension. The fuzzification layer maps the input patterns into the fuzzy sets through BYY learning, which will be discussed in Sect. 2.2. Thereafter, the association layer associates the fuzzy rules to the memory cell and try to imitate human's cerebellum. The logical AND operation is carried out in this layer to ensure that a cell is activated only when all the inputs associated to it are fired. The weights of the fuzzy rules are derived by the truth value restriction

(TVR) scheme, which will be discussed in Sect. 2.3. The association layer is then mapped to the post association layer where the logical OR operation will fire those cells whose connected inputs are activated. For the output layer, the defuzzification center of area (COA) method is used to compute the output of the structure.

$$\omega_j^{(i)} = \omega_j^{(i-1)} + \frac{\alpha \left( \sum_{k=1}^m f(k) / f(l) \right)}{\varphi} (y_d - y_j), \quad (1)$$

where  $\varphi = \sum_{l=1}^m \left\{ \sum_{k=1}^m f(k) / f(l) \right\}$ ,  $\omega_j^{(i)}$  is the weight of the  $j$ th cell after  $i$  iterations,  $\alpha$  is the learning rate, while  $y_d$  and  $y_j$  are the desired and calculated output respectively, and  $f(l)$  returns the variable  $f\_freq$ . Using the derived  $\varphi$ , Eq. (1) reduces proportionally the learning rate of a cell as its fired frequency increases.

### 2.2 Fuzzification using BYY learning

Treating both  $x$  and  $y$  as random processes, the joint distribution can be calculated by either of these two equations:

$$p(x, y) = p(y|x)p(x), \quad (2)$$

$$q(x, y) = q(x|y)q(y). \quad (3)$$

The breakdown of Eq. (2) follows the Yang concept with the visible domain by  $p(x)$  regarded as a Yang space and the forward pathway by  $p(y|x)$  as a Yang pathway. Similarly, Eq. (3) is regarded as a Ying space and the backward pathway by  $q(x|y)$  as a Ying pathway. Under the BYY learning principle, the difference between the two Bayesian representations in Eqs. (2) and (3) should be minimized. The joint probability density of the  $j$ th dimension that consists of  $K^j$  Gaussian functions is given by

$$p(x_i^j, \Theta^j) = \sum_{y=1}^{K^j} \alpha_y^j G(x_i^j, m_y^j, \sigma_y^j), \quad (4)$$

where  $x_i^j$  is the  $i$ th input data of the  $j$ th dimension,  $\Theta^j$  is a set of finite mixture model parameter that consists of  $\alpha_y^j$  (prior probability),  $m_y^j$  (mean value), and  $\sigma_y^j$  (width of the  $y$ th cluster).

Given the number of clusters,  $K^j$ , the minimization of the Kullback-Leibler divergence will produce the optimal parameter  $\Theta^j$ . The learning procedure can be implemented by iterative expectation-maximization (EM) algorithm. In practice, for each dimension  $j$ , we start with  $K^j=1$  and estimate the parameter  $\Theta^j$  by the EM algorithm based on the given training data. The values are then used to compute a cost value through a cost function. Then, we proceed to  $K^j \rightarrow K^j+1$  and compute the cost function again. After this, we gather a

series of cost function values, and the optimal cluster number,  $K^{j*}$ , is selected from the one with the minimal cost function value. The details of calculating the cost function values are given in our previous work [8].

### 2.3 Truth value restriction inference scheme

There are three general approaches in the use of neural network techniques to perform fuzzy inference: TVR, compositional rule of inference scheme (CRI), and approximate analogical reasoning schema (AARS) [15]. The TVR offers a consistent rule base, a strong theoretical foundation, and it is more logical and spontaneous to the human reasoning process as compared to CRI and AARS [16]. TVR uses inference rules to obtain the truth-values of the consequents from that of the antecedents. Suppose that the fuzzy sets of the antecedents  $Y$  (i.e., the fuzzification layer of CMAC) and the fuzzy sets of the consequents  $B$  (i.e., output layer of CMAC) have been determined after training. Given an arbitrary input  $x$ , its actual value is not exactly any particular antecedent value. Accordingly, the actual output is also not exactly categorized to the consequents  $B$  but to some other fuzzy set  $B'$ .

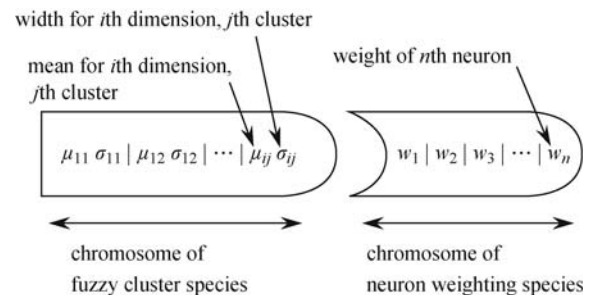
## 3 FCMC-BYY with coevolutionary learning

In traditional evolutionary computation, the expected solution is searched among a population of individuals via genetic operations, such as crossover and mutation. However, single-population evolutionary algorithms often perform poorly — manifesting stagnation and convergence to local optima when confronted with complex problem, or strong interdependencies among the components of the solution. Coevolution learning allows simultaneous evolution of two or more species with coupled fitness. Such coupled evolution favors the discovery of complex solutions. Typically, the coevolving species can be classified into two types: competitive coevolutionary algorithm, which obtains exclusivity on a limited resource, and cooperative coevolutionary algorithm that gains access to some hard-to-attain resource.

On the other hand, traditional binary GA has some drawbacks when applying to multidimensional and high-precision numerical problems. The situation can be improved if GA in real numbers is used. Each chromosome is coded as a vector of floating point numbers that has the same length as the solution vector. A large domain can thus be handled. Much research effort has been spent to improve the performance of real-coded GA (RCGA). In general, RCGA involves three operations: selection, crossover, and mutation. The selection operation is used to select the chromosomes from the population with respect to some probability distribution based on fitness

values. The crossover operation is used to combine the information of the selected chromosomes (parents) and generate the offspring. The mutation operation is used to change the offspring genes [17].

In this research, coevolutionary computation and real-coded GA are applied to FCMAC-BYY, referred to as FCMAC-EBYY, in which the fuzzy sets and the weights are evolved independently and then cooperated to form the final solution. Coevolutionary learning is integrated into the optimization of fuzzy clusters and weights of the FCMAC-BYY network. There are two kinds of species in the FCMC-EBYY, namely, fuzzy cluster species and weight species. In the species of fuzzy clusters, all the chromosomes are initialized with the fuzzification results obtained by the BYY learning. In the specie of connection weights, all the chromosomes are initialized with randomly generated real values. Such a treatment keeps a good mixture of the optimized outputs from BYY fuzzification and random processes, so as to speed up the searching process. An example of the chromosome representations is shown in Fig. 1, where  $\mu_{ij}$  and  $\sigma_{ij}$  represent the mean values and widths of the  $i$ th dimension in the  $j$ th cluster of fuzzy cluster species, respectively, and  $w_1, w_2, \dots, w_n$  represent the weights of the corresponding rules in the neuron weighting species.



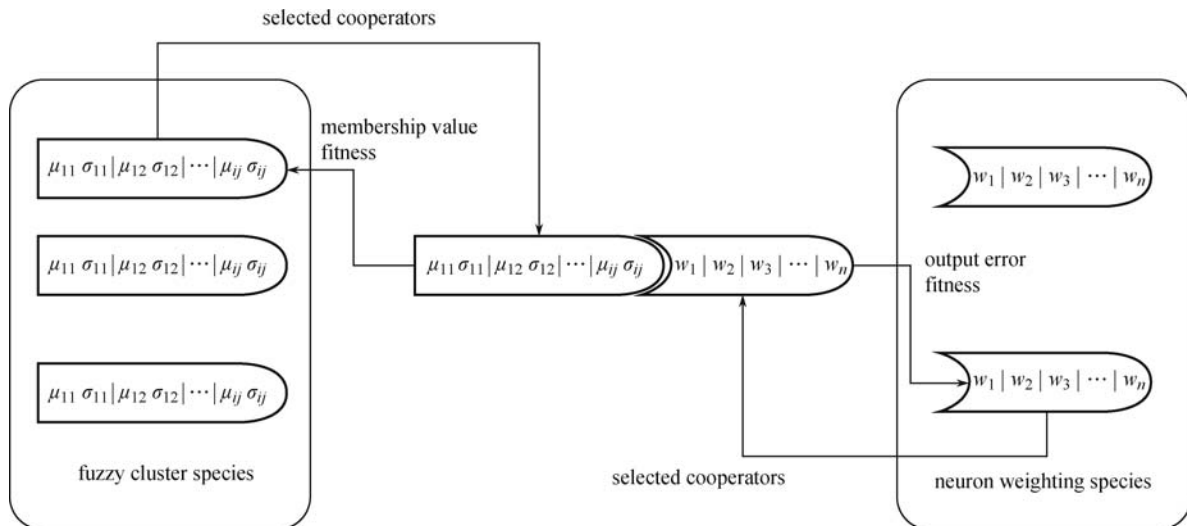
**Fig. 1** Chromosome representations for species of fuzzy clusters and connection weights

The following are the details of the five processes in coevolutionary learning: evaluation, selection, reproduction (crossover and mutation), and reinsertion.

#### 1) Evaluation

The purpose of the evaluation process is to determine how well a trial solution would perform when it is given a set of test data and their respective results. Using the training data as a gauge, we can find the better fitting solutions, in return, boast a high possibility of gaining similar results in testing. The detailed view of the fitness evaluation process is illustrated in Fig. 2.

In the fuzzy cluster species, the membership value of training data for each cluster is used as the fitness function of each chromosome to measure that the degree of the training data belongs to the fuzzy clusters. On the other hand, the output error is used as the fitness function in the connection weights species to train the weights of the neurons in the rule layer. As shown in



**Fig. 2** Fitness evaluation of two species in FCMAC-EBYY

Fig. 2, an individual undergoing fitness evaluation establishes cooperation with one or more representatives of the other species to construct a candidate FCMAC. A selected individual from the fuzzy cluster species will cooperate with the selected individuals from the connection weights species to form a full representation of the structure.

## 2) Selection

Selection is the process of selecting a trial solution to the mating pool for reproduction process. The stochastic universal sampling (SUS) is employed in the FCMAC-EBYY, in which the chromosomes are mapped one-to-one into contiguous segments of a line, where the size of the segment of each chromosome relates to its fitness value. Then, equally spaced pointers are placed along the line. The number of pointers on the line corresponds to the number of individuals to be selected.

## 3) Crossover

In the crossover process, new offsprings are reproduced such that they possess parts of both parent's genetic characteristics, just like its counterpart in nature. Intermediate recombination is suitable to fulfill crossover between two real-valued parent chromosomes [18]. Each variable in the offspring is the result of combining the variables in the parents according to the above expression with a new  $\alpha$  chosen for each pair of parent genes. Intermediate crossover is capable of producing new variables within a slightly larger hypercube than that defined by the parents but constrained by a range.

## 4) Mutation

Mutation can be accomplished by replacing the parameter values with random selection of new values. These new values, which are selected from the allowable ranges for the respective dimension, ensure that the mutated values are within reasonable regions. Similar to the crossover operator, the probabilistic rate of the mutation

determines the frequency of mutations. In all, mutation allows an increase in the level of possible exploration of the search space without adversely affecting the convergence characteristics.

## 5) Reinsertion

After the reproduction process is completed, a new population of chromosomes is produced. This new population will then be combined with the old population to form the resultant population. The elitist strategy is employed to select a number of old chromosomes to be reinserted into the resultant population. This strategy chooses a pre-specified number of fittest chromosomes in the old population to be inserted into the resultant population, while the rest of the chromosomes are discarded. The deleted cluster parameters are replaced by the chromosomes created from the reproduction process. This is done to retain the best quality chromosomes in the population. Once the resultant population has been produced, it marks the completion of one generation.

## 4 Experimental results

In this section, the FCMAC-EBYY network is used to conduct experiments on a real application of traffic flow prediction. In our experiments, the evolutionary computation stops when the number of generations reaches up to 500 (with some initial elite individuals obtained from BYY fuzzification). The population size is set to 30, crossover rate = 0.8, and mutation rate = 0.06 based on our empirical experience.

The raw traffic flow data for the simulation is obtained from Refs. [19,20]. The data were collected at a site (Site 29) located at exit 15 along the east-bound Pan Island Expressway (PIE) in Singapore using loop detectors embedded beneath the road surface. There are a total of five lanes at the site, two exit lanes, and three straight

lanes for the main traffic. For this experiment, only the traffic flow data for the three straight lanes were considered. The traffic data set has four input attributes, i.e., the time and the traffic density of the three lanes.

The purpose of this simulation is to model the traffic flow trend at the site using FCMAC-EBYY. It is then used to obtain prediction for the traffic density of each lane at time  $t + \tau$ , where  $\tau = 5, 15, 30, 45, 60$  min. For the

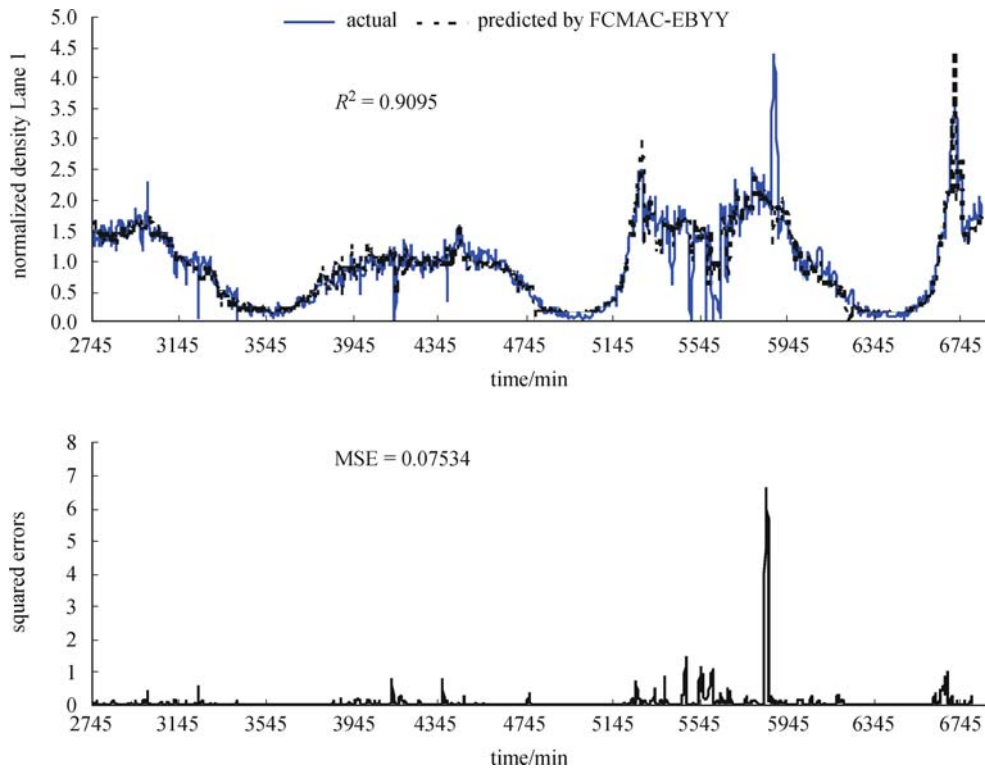


Fig. 3 Prediction and squared errors of FCMAC-EBYY at  $\tau = 5$  min

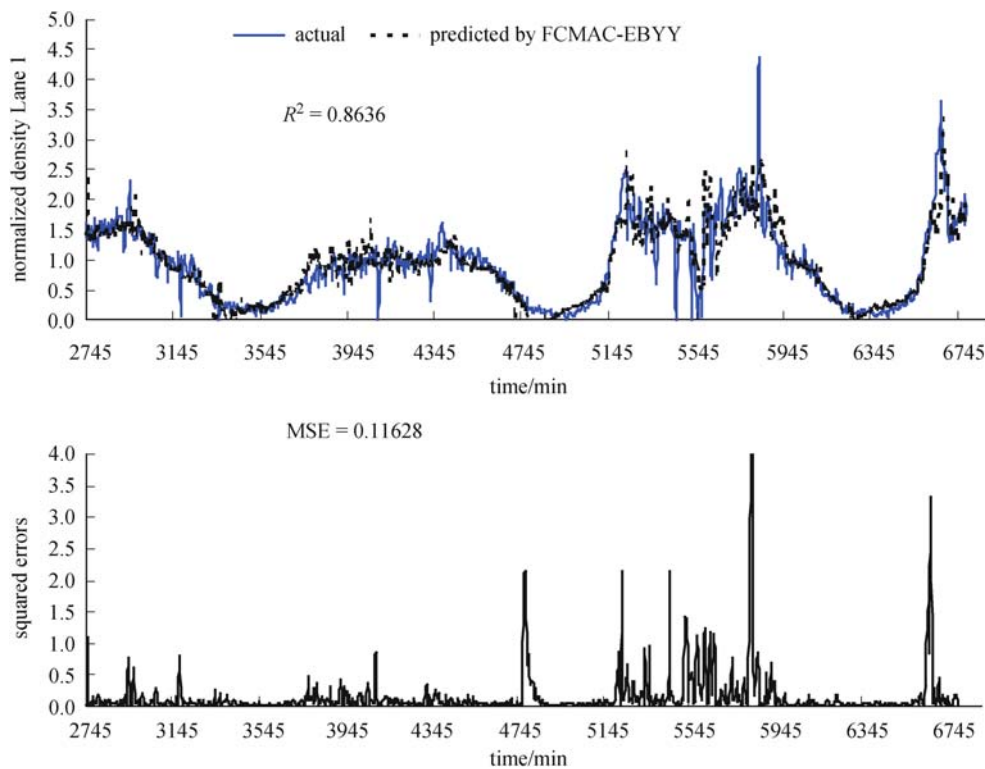


Fig. 4 Prediction and squared errors of FCMAC-EBYY at  $\tau = 60$  min

simulation, three cross-validation groups, CV1, CV2, and CV3, of training and test sets are used. The square of the Pearson product-moment correlation value (denoted as  $R^2$ ) is used to compute the accuracy of the predicted traffic trends obtained using FCMAC-EBYY network. The prediction and squared errors of Lane 1 density using online FCMAC-EBYY are shown in Figs. 3 and 4 at  $\tau = 5$  min and  $\tau = 60$  min of CV1, respectively.

## 5 Conclusion

In this paper, an architecture of FCMAC is implemented using EBYY learning. After the initial cluster parameters are obtained through BYY learning, evolutionary computation is utilized to simultaneous evolution of two species, namely, fuzzy clusters and connection weights species. This strategy offers exploring the search domains of both the fuzzy clusters and the connection weights. FCMAC-EBYY is applied to traffic flow prediction.

**Acknowledgements** This research was supported by the Ministry of Knowledge Economy (MKE), Korea, under the Information Technology Research Center (ITRC) supervised by the National IT Industry Promotion Agency (NIPA) (NIPA-2010-(C1090-1021-0002)). It was sponsored by Daegu Gyungpook Development Institute 2010.

## References

1. Albus J S. Data storage in the cerebellar model articulation controller (CMAC). *Transaction of the ASME, Journal of Dynamic Systems, Measurement, and Control*, 1975, 97(3): 228–233
2. Albus J S. A new approach to manipulator control: the cerebellar model articulation controller (CMAC). *Transaction of the ASME, Journal of Dynamic Systems, Measurement, and Control*, 1975, 97(3): 220–227
3. Shi D, Quek C, Tilani R, Fu J. Product demand forecasting with a novel fuzzy CMAC. *Neural Processing Letters*, 2007, 25(1): 63–78
4. Xu L. Bayesian-Kullback coupled YING-YANG machines: unified learning and new results on vector quantization. In: *Proceedings of the International Conference on Neural Information Processing*. 1995, 977–988
5. Xu L. Bayesian Ying-Yang system, best harmony learning and five action circling. *Frontiers of Electrical and Electronic Engineering in China*, 2010, 5(3): 281–328
6. Xu L. Advances on BYY harmony learning: information theoretic perspective, generalized projection geometry, and independent factor autodetermination. *IEEE Transactions on Neural Networks*, 2004, 15(4): 885–902
7. Xu L. BYY harmony learning, structural RPCL, and topological self-organizing on mixture models. *Neural Networks*, 2002, 15(8–9): 1125–1151
8. Nguyen M N, Shi D, Quek C. FCMAC-BYY: fuzzy CMAC using Bayesian Ying-Yang learning. *IEEE Transactions on Systems, Man and Cybernetics – Part B*, 2006, 36(5): 1180–1190
9. Shi D, Nguyen M N, Zhou S, Yin G. Fuzzy CMAC with incremental Bayesian Ying-Yang learning and dynamic rule construction. *IEEE Transactions on Systems, Man and Cybernetics – Part B*, 2010, 40(2): 548–552
10. Holland J. *Adaptation in Natural and Artificial Systems*. Ann Arbor: University of Michigan Press, 1975
11. Howell M N, Gordon T J, Brandao F V. Genetic learning automata for function optimization. *IEEE Transactions on Systems, Man, and Cybernetics – Part B*, 2002, 32(6): 804–815
12. Shi D, Dong C, Yeung D S. Neocognitron's parameter tuning by genetic algorithms. *International Journal of Neural Systems*, 1999, 9(6): 497–509
13. Pena-Reyes C A, Sipper M. Fuzzy CoCo: a cooperative-coevolutionary approach to fuzzy modeling. *IEEE Transactions on Fuzzy Systems*, 2001, 9(5): 727–737
14. García-Pedrajas N, Hervás-Martínez C, Ortiz-Boyer D. Cooperative coevolution of artificial neural network ensembles for pattern classification. *IEEE Transactions on Evolutionary Computation*, 2005, 9(3): 271–302
15. Quek C, Zhou R W. POPFNN: a pseudo outer-product based fuzzy neural network. *Neural Networks*, 1996, 9(9): 1569–1581
16. Nafarieh A, Keller J M. A new approach to inference in approximate reasoning. *Fuzzy Sets and Systems*, 1991, 41(1): 17–37
17. Yao X. Evolving artificial networks. *Proceedings of the IEEE*, 1999, 87(7): 1423–1447
18. Mülenbein H, Schlierkamp-Voosen D. Predictive models for the breeder genetic algorithm: I. Continuous parameter optimization. *Journal of Evolutionary Computation*, 1993, 1(1): 25–49
19. Quek C, Pasquier M, Lim B B S. POP-TRAFFIC: a novel fuzzy neural approach to road traffic analysis and prediction. *IEEE Transactions on Intelligent Transportation Systems*, 2006, 7(2): 133–146
20. Nguyen M N, Shi D, Quek C, Ng G S. Traffic prediction using Ying-Yang fuzzy cerebellar model articulation controller. In: *Proceedings of the 18th International Conference on Pattern Recognition*. 2006, 3: 258–261



Payam S. RAHMDEL received B.Sc in electrical and electronics engineering from Islamic Azad University of Sabzevar, Iran in 2008. He is currently Ph.D student at the School of Engineering and Information Sciences at Middlesex University, London, UK. His research interest is biomedical image processing, machine vision and vision-based robot navigation.



Minh Nhut NGUYEN received the B.Eng degree and M.Phil degree in computer engineering from Ho Chi Minh City University of Technology, Vietnam in 2001 and 2005 respectively. He obtained his Ph.D in computer science from Nanyang Technological University, Singapore. Dr. Nguyen is now with Institute for Infocomm Research, Singapore. His research interests include machine learning, fuzzy sets theory,

pattern recognition and neural networks.



Liying ZHENG received the Ph.D degree in control theory and control engineering from Harbin Engineering University (HEU) in China in 2003. She has been serving as an associate professor in HEU since 2005. Her current research interests include image segmentation and pattern recognition.