2011 IEEE International Conference on Fuzzy Systems
June 27-30, 2011, Taipei, Taiwan

FAPOP: Feature Analysis Enhanced Pseudo Outer-Product Fuzzy Rule Identification System

Sau Wai Tung and Chai Quek
Centre for Computational Intelligence
School of Computer Engineering
Nanyang Technological University
Singapore 639798
Email: swtung@pmail.ntu.edu.sg; ashcquek@ntu.edu.sg

Cuntai Guan
Institute for Infocomm Research
Agency for Science, Technology and Research (A*Star)
Singapore 138632
Email: ctguan@i2r.a-star.edu.sg

***Abstract*—Most existing neural fuzzy systems either overlook the importance of feature analysis; or it is performed as a separate phase prior to the design stage of the systems. This paper proposes a novel neural fuzzy system, named *Feature Analysis Enhanced Pseudo Outer-Product Fuzzy Rule Identification System* (FAPOP), which integrates its design with feature analysis. The objective is two-folds; namely, (1) to improve the interpretability of the system by identifying features relevant to its computational structure; and (2) to improve the accuracy of the system by identifying features relevant to the application problem. The proposed FAPOP model is subsequently employed in a series of benchmark simulations to demonstrate its efficiency as a neural fuzzy modeling system, and excellent performances have been achieved.**

***Index Terms*—Feature analysis, Pseudo-outer product (POP), Categorical Learning Induced Partitioning (CLIP), Nakanishi dataset, Mackey-Glass prediction.**

I. INTRODUCTION

Existing neural fuzzy systems can be broadly classified into two classes; namely, linguistic fuzzy models [1] [2], which are focused on delivering a good level of interpretability to the structures of the models, and precise fuzzy models [3] [4], which are focused on achieving high accuracies in the modeling tasks. While Mamdani models [5] form the main focus in the former class; the latter class comprises mainly of the Takagi-Sugeno-Kang models [6] [7]. In general, having a good fuzzy rulebase interpretability and a high modeling accuracy are two contradictory requirements in the design of a neural fuzzy system [8].

With its success in data-mining applications, feature analysis techniques [9] are often performed either implicitly or explicitly as a *pre-processing* phase prior to the design stage of a neural fuzzy system. The main advantage is an improvement to the modeling accuracy through the removal of redundant/derrogative features and the identification of relevant features for the underlying application problem. Nevertheless, the decoupled stages do not consider the close relationship between feature analysis and system design [10], i.e., to identify features that are important for the application problem *and* are relevant to the output of the neural fuzzy system.

This paper presents the *Feature Analysis Enhanced Pseudo-Outer Product Fuzzy Rule Identification System* (FAPOP), a novel neural fuzzy system which integrates its design with feature analysis. An initial fuzzy rulebase is obtained via Quek and Zhou's pseudo-outer product (POP) rule identification algorithm [11]. Knowledge are then extracted simultaneously from the initial rulebase and a single-pass of the training data to incrementally update a relevance weight for each input feature. All input features are relevant by default; while irrelevant/redundant input features are marked by decreasing relevance weights. Since knowledge from both the initial rulebase and the training data are utilized to determine the relevance weights, this allows FAPOP to identify features that are important for both the system and the application. The objective is to provide a regulated balance between the two contradictory requirements; namely, (1) to improve the interpretability of the system by identifying features relevant to its computational structure; and (2) to improve the accuracy of the system by identifying features relevant to the application problem. Following that, irrelevant input features, together with contradictory and/or identical rules, are removed to maintain a compact structure in FAPOP.

The rest of the paper is organized as follows. The computational structure and the reasoning process of the FAPOP model are described in Section II. The proposed rule identification and feature selection methodologies are introduced in Section III. Section IV evaluates the learning and generalization abilities of FAPOP through a series of benchmark experimental simulations. Lastly, Section V concludes the paper.

II. FAPOP: ARCHITECTURE AND REASONING PROCESS

The proposed FAPOP model is a five layers neural fuzzy system as shown in Fig. 1. Layer 1 is the input variable nodes IV_i; layer 2 is the antecedent nodes A_{i,j_i}; layer 3 is the rule nodes R_k; layer 4 is the consequent nodes $C_{l_m,m}$; and layer 5 is the output variable nodes OV_m. In the FAPOP model, the input vector is denoted as $x = [x_1, \ldots, x_I]^T$ with corresponding desired output vector denoted as $d = [d_1, \ldots, d_M]^T$; while the computed output vector is denoted as $y = [y_1, \ldots, y_M]^T$. Each input variable node consists of J_i number of antecedent fuzzy labels; while each output variable node consists of L_m number of consequent fuzzy labels. Layer 3 of FAPOP represents the K fuzzy rules in the system such that each node encodes a

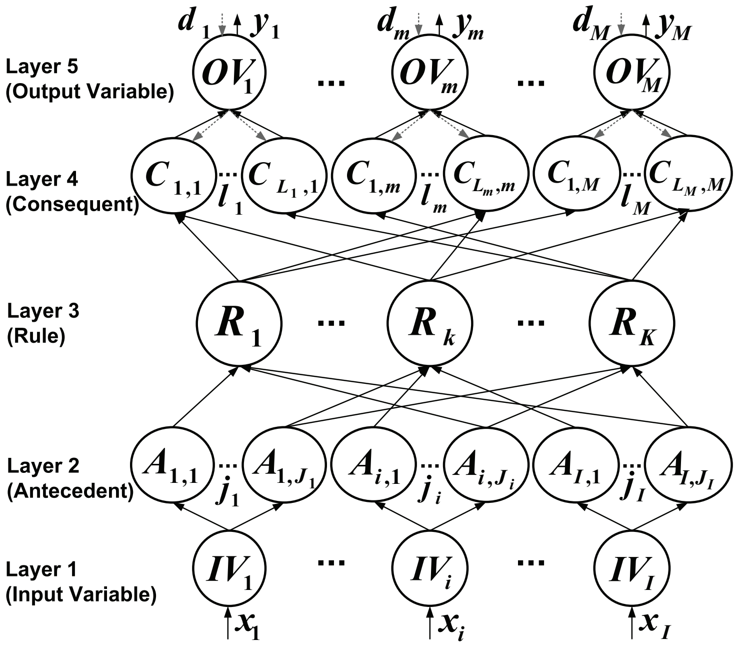


Fig. 1. Architecture of FAPOP.

fuzzy rule of the form given as in (1):

$$\begin{aligned} R_k : \text{IF } x_1 \text{ is } A^{(k)}_{1,j_1} \text{ and } \ldots \text{ and } x_I \text{ is } A^{(k)}_{I,j_I} \\ \text{THEN } y_1 \text{ is } C^{(k)}_{l_1,1} \text{ and } \ldots \text{ and } y_M \text{ is } C^{(k)}_{l_M,M} \end{aligned} \quad (1)$$

where $A^{(k)}_{i,j_i}$ (resp. $C^{(k)}_{l_m,m}$) is the j_i-th antecedent (resp. l_m-th consequent) node associated with the i-th input (resp. m-th output) variable that is connected to the rule node R_k. The tunable parameters of FAPOP are the centres and widths of the fuzzy labels embedded in the antecedent and consequent nodes, where each node defines a gaussian membership function described as in (2):

$$\mu(c, \sigma; x) = e^{-((x-c)^2/\sigma^2)} \quad (2)$$

such that c and σ are the centre and width of the function respectively. Adaptation of the parameters is performed using backpropagation [12].

The reasoning process of FAPOP is represented by the forward solid arrows in Fig. 1 where the input vector x is presented to the system at layer 1. The proposed system then performs inference based on the input vector by propagating the information through layers 2 to 4. Consequently, the system produces a computed output vector y at layer 5. Each corresponding output for an arbitrary node is denoted as fo such that the reasoning process of FAPOP is discussed below:

Layer 1: The input node IV_i directly passes on the input value to the next layer such that its neural operation is described as in (3):

$$fo_i = x_i \ . \quad (3)$$

Layer 2: The antecedent node A_{i,j_i} performs similarity matching of the input value with the corresponding fuzzy label such that its neural operation can be described as in (4):

$$fo_{i,j_i} = \mu_{i,j_i}(c_{i,j_i}, \sigma_{i,j_i}; fo_i) \quad (4)$$

where $\mu_{i,j_i}(c_{i,j_i}, \sigma_{i,j_i}; x)$ refers to the gaussian membership function embedded in A_{i,j_i}.

Layer 3: Each rule node R_k computes the overall degree of similarity between the input vector and the antecedent part of the k-th fuzzy rule such that its firing rate is computed as in (5):

$$fo_k = \min_{i \in \{1 \ldots I\}} fo^{(k)}_{i,j_i} \ . \quad (5)$$

Layer 4: The consequent node $C_{l_m,m}$ performs consequent derivation for the fuzzy rules based on the information from the input vector. Since $C_{l_m,m}$ may serve as output to more than one fuzzy rule, its cumulative neural operation can be described as in (6):

$$fo_{l_m,m} = \max_{k \in \{1 \ldots K_{l_m,m}\}} fo^{(l_m,m)}_k \quad (6)$$

where $K_{l_m,m}$ is the total number of fuzzy rules in FAPOP that shares the same consequent node $C_{l_m,m}$ and $fo^{(l_m,m)}_k$ is the output of the k-th rule that shares $C_{l_m,m}$.

Layer 5: The output node OV_m performs defuzzification via the center of averaging method [13] to obtain a crisp output value such that its neural operation can be described as in (7):

$$y_m = fo_m = \frac{\sum_{l_m \in \{1 \ldots L_m\}} fo_{l_m,m} c_{l_m,m} \sigma_{l_m,m}}{\sum_{l_m \in \{1 \ldots L_m\}} fo_{l_m,m} \sigma_{l_m,m}} \quad (7)$$

where $c_{l_m,m}$ and $\sigma_{l_m,m}$ are the centre and width of the gaussian function embedded in $C_{l_m,m}$ respectively.

III. Learning Mechanisms of FAPOP

In the proposed FAPOP model, an initial fuzzy rulebase is obtained via Quek and Zhou's POP rule identification algorithm [11]. A relevance weight for each input feature is then incrementally determined using knowledge extracted from both the initial fuzzy system and a single-pass of the training data. The relevance weight of an input feature reflects its role in describing the underlying application problem and the output of the system. As stated before, all input features are assumed relevant at the beginning. Input features with consistently high relevance weights are selected; while input features with decreasing relevance weights are regarded as redundant and discarded. Subsequently, irrelevant input features, together with contradictory and/or identical rules, are removed to maintain a compact structure in the FAPOP model.

The learning mechanisms of FAPOP are discussed here as follows. Details of the neural computations in the proposed system are described in Section III.A.; Section III.B. reviews the POP algorithm; Section III.C. introduces the feature analysis technique in FAPOP; and Section III.D. illustrates the ideas discussed using a case study.

A. Neural Computations in FAPOP

Neural computations in the proposed FAPOP model are bidirectional, in the forward and backward sense as described below:

1. Forward Operation: The forward operation of FAPOP coincides with its reasoning path as described in Section II.
2. Backward Operation: The backward operation of FAPOP is represented by dotted arrows in Fig. 1, which is a mirrored computation of the forward operation. Each corresponding output for an arbitrary node is denoted as *bo*.

   Layer 5: The output node OV_m directly passes on the output value to the next layer such that its neural operation is described as in (8):

$$bo_m = d_m \; . \tag{8}$$

   Layer 4: The consequent node $C_{l_m,m}$ performs similarity matching of the output value with the corresponding fuzzy label such that its neural operation can be described as in (9):

$$bo_{l_m,m} = \mu_{l_m,m}(c_{l_m,m}, \sigma_{l_m,m}; bo_m) \tag{9}$$

   where $\mu_{l_m,m}(c_{l_m,m}, \sigma_{l_m,m}; x)$ refers to the gaussian membership function embedded in $C_{l_m,m}$.

B. POP algorithm

The POP algorithm consists of two phases in its learning process; namely, fuzzy partitioning and rule identification. In this paper, fuzzy partitioning is performed using the Categorical Learning Induced Partitioning (CLIP) [14]; while a description of the rule identification follows.

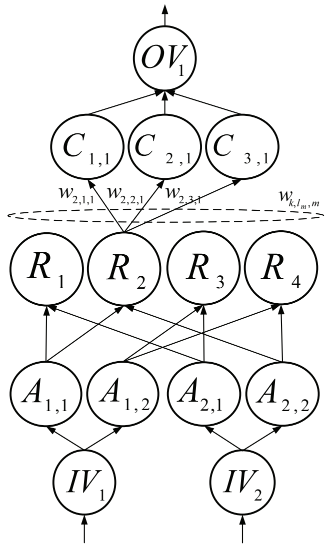


Fig. 2. Illustration for POP rule identification algorithm.

For simplicity, assume an application problem with 2 inputs-1 output as shown in Fig 2. Two fuzzy labels are determined by the CLIP algorithm for each input space; while three fuzzy labels are determined for the output space. The POP algorithm considers all possible rules, such that four fuzzy rules are initialized. The system is fully connected from the rule layer to the consequent layer with weights $w_{k,l_m,m}$, $k = 1 \ldots 4$, $l_m = 1 \ldots 3$, $m = 1$. Fig 2 illustrates the rule identification process for the rule node R_2:

1. Initialization: Set $w_{k,l_m,m} = 0$.
2. Weight Update: Through a single-pass of the training data, update the weights as follows:

$$w_{k,l_m,m} = \sum_N fo_k \cdot bo_{l_m,m}$$

   where N is the size of the training data.
3. Rule Formulation: Select the m-th consequent node for R_k as one with the maximum weight given as follows:

$$C^{(k)}_{l_m,m} = C_{l^\star_m,m}$$

   where $w_{k,l^\star_m,m} = \max_{l_m} w_{k,l_m,m}$.

C. Feature Analysis in FAPOP

The forward-and-backward neural computations (as described in Section III.A.) are defined to compute the instantaneous relevance weight of an input feature i when a data pair $[x, d]$ is presented to FAPOP. The instantaneous relevance weight $\widehat{\omega_i}$ is computed as in (10):

$$\widehat{\omega_i} = \max_{j_i}\left[\min_k\left(fo^{(k)}_{i,j_i}, bo^{(k)}_{l_m,m}\right)\right] \tag{10}$$

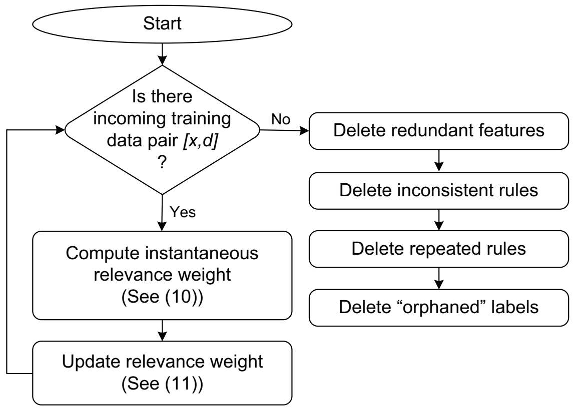


Fig. 3. Flowchart of the feature analysis process employed in FAPOP.

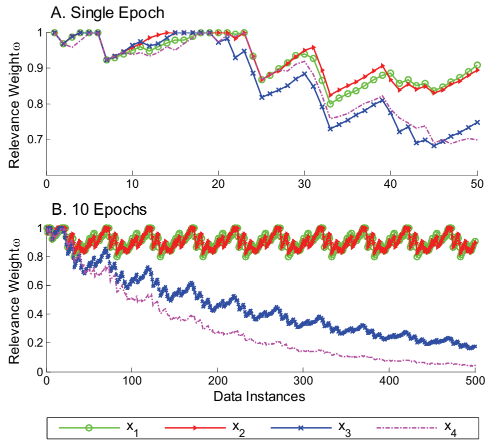


Fig. 4. Updates of the relevance weights for input features x_1–x_4 in the case study: A. For a single-pass of the training data; and B. For 10 passes of the training data.

where $fo_{i,j_i}^{(k)}$ (resp. $bo_{l_m,m}^{(k)}$) is the computed output of the antecedent node A_{i,j_i} (resp. consequent node $C_{l_m,m}$) that is connected to the rule node R_k. Knowledge is extracted from the structure of each rule node R_k, consisting of its antecedent and consequent parts, for the computation of the instantaneous relevance weight $\widehat{\omega}_i$. Subsequently, $\widehat{\omega}_i$ is given by the maximum activations of its respective fuzzy labels. That is, the instantaneous relevance weight $\widehat{\omega}_i$ of an input feature i represents the activation level of feature i by the current data pair based on the knowledge encoded in FAPOP.

Feature analysis in FAPOP proceeds with incrementally updating a relevance weight for each input feature. The relevance weight ω_i for an input feature i is computed as in (11):

$$\begin{aligned} &\omega_i\left(0\right) = 1\ ; \\ &\omega_i\left(t+1\right) = \min\left[1,\ \omega_i\left(t\right) - \omega_i\left(t\right)\cdot\left(\eta_d - 1\right)\cdot\left(\widehat{\omega}_i\left(t\right) - h\right)\right] \end{aligned} \tag{11}$$

where $\eta_d < 1$ is a natural decaying constant and h is the harsh factor. The relevance weight of an input feature i is initialized to unity, assuming a high level of relevance since $0 < \omega_i \leq 1$. Harsh factor h, $0 \leq h \leq 1$, determines the level of stringency in the feature selection process, i.e., for a large value of h, only highly relevant features are selected; while for a lower value of h, some redundant features might be included in the selection. This is because when $\widehat{\omega}_i\left(t\right) < h$, we have $\omega_i(t+1) < \omega_i(t)$. Hence when h is a large value and most instants of $\widehat{\omega}_i$ fall below h, the resultant relevance weight of input feature i will be decreasing over time. On the other hand, when $\widehat{\omega}_i\left(t\right) > h$, we have $\omega_i(t+1) > \omega_i(t)$. Hence when h is a small value and most instants of $\widehat{\omega}_i$ are greater than h, the resultant relevance weight of input feature i will be increasing over time. Nevertheless, ω_i will be cap at unity.

Fig. 3 shows a flowchart of the feature analysis process employed in the proposed FAPOP model. With each incoming training data pair $[x, d]$, the instantaneous relevance weight $\widehat{\omega}_i$ of an input feature i is computed as the activation of the input feature by this current data pair based on the knowledge encoded in the system. FAPOP then proceeds with updating the relevance weight ω_i of an input feature i using (11). After a single-pass of the training data, redundant features in the proposed model are identified and removed from the system. Redundant features are recognized by their decreasing relevance weights. Subsequently, contradictory/inconsistent rules with lower weightages are removed. An inconsistent rulebase occurs when there exists two rules such that the antecedent conditions are similar but the resultant consequences differ; while the weightage of a fuzzy rule is defined as $\sum_m w_{k,l_m^*,m}$. Following that, the system deletes repeated rules, i.e. , rules with similar antecedent and consequent segments. Finally, some of the fuzzy labels might be "orphaned" when all fuzzy rules associated with them have been deleted. The orphaned fuzzy labels are also removed to ensure that the resultant structure of FAPOP is compact.

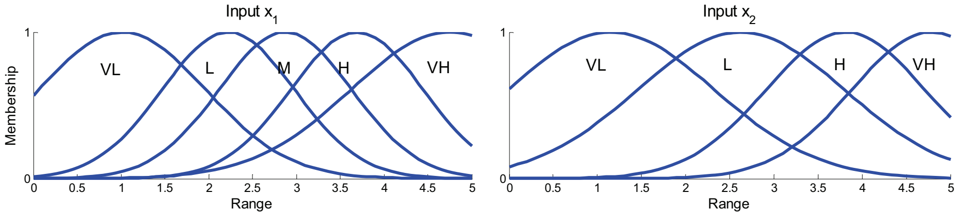
D. Case Study

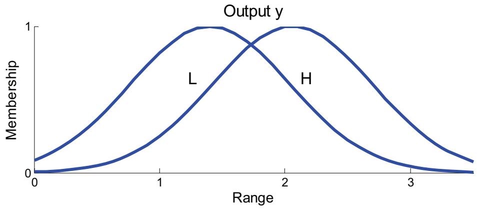
This section illustrates the proposed learning mechanisms in the FAPOP model with a case study. The dataset is generated by a nonlinear equation as described in (12):

$$y = \left(1 + x_1^{-2} + x_2^{-1.5}\right)^2 \tag{12}$$

where x_1 and x_2 are the two inputs in the system; while y is the output. Following the problem described in [15], a total of 50 data pairs are generated for $0 \leq x_1, x_2 \leq 5$. Two random variables x_3 and x_4, in the range of $[0, 5]$, are subsequently added as dummy inputs. Hence, the application problem is a 4 inputs-1 output system, such that input features x_3 and x_4 are expected to be indifferent to the output.

A total of 5, 4, 7 and 5 fuzzy labels are identified by the CLIP algorithm for the four input spaces respectively; and 3 fuzzy labels are determined for the output space. That means, 700 fuzzy rules have been initialized by the POP algorithm such that the initial structure of FAPOP is given





Rule	x_1	x_2	y
1	VH	VH	L
2	VH	H	L
3	VH	L	L
4	H	VH	L
5	H	H	L
6	M	VH	L

L – LOW   M – MEDIUM   H – HIGH   V – VERY

Fig. 5. Fuzzy clusters identified for inputs x_1–x_2 and output y in the case study; and part of the identified rulebase in FAPOP.

by the configuration 4-21-700-3-1. Following that, FAPOP proceeds with the feature analysis process to identify relevant input features to the system and the application. Fig. 4 shows the updates for the relevance weights for the four inputs. After a single-pass of the training data (see Fig. 4.A.), the decline in the relevance weights for input features x_3 and x_4 are much faster compared to those for x_1 and x_2, indicating that they might be irrelevant. To emphasize the disparity in the relevance weights between the relevant and irrelevant features, the training data are cycled for ten epochs in this experiment as shown in Fig. 4.B. From the figure, the relevance weights of the redundant input features x_3 and x_4 approach zero with the increase in the number of running epochs, while that of the relevant input features x_1 and x_2 are maintained consistently at a high level. In general, irrelevant/redundant input features are marked by decreasing relevance weights, where the decline can be emphasize with an increase in the number of running epochs. Nevertheless, it should be noted that a large training dataset will lead to an increase in the time complexity with an increase in the number of running epochs. In this paper, the number of running epochs for the experiments is capped at 10.

After the redundant features (x_3 and x_4) have been identified, they are removed from the FAPOP model. Inconsistent rules of lower weightages, as well as identical rules, are also removed to maintain a compact structure in FAPOP. Finally, orphaned fuzzy labels are deleted from the system. The final structure of FAPOP is given by the configuration 4-9-20-2-1, i.e., there are 20 identified rules in the proposed model and one orphaned fuzzy label in the output space is also deleted. As seen, there is a significant reduction of 97% in the the number of identified rules.

Fig. 5 illustrates the identified fuzzy clusters in FAPOP for the input spaces x_1 and x_2, and that for the output space y. Since the fuzzy clusters are highly ordered, clear semantic meanings can be attached to the fuzzy clusters. A total of 20 fuzzy rules are identified by the FAPOP model, with six rules corresponding to a low output value, and fourteen rules corresponding to a high output value. Subsequently, the six rules are listed in Fig. 5. One can easily verify that the derived rulebase is consistent, with the identified fuzzy rules reflecting the relationship between the inputs and the output of the system, i.e., based on (12), for relatively high values of inputs x_1 and x_2, the resultant output y of the system is low. The reverse is also true for a high output value y.

IV. Experimental Results

This section illustrates the effectiveness of the proposed FAPOP model as a neural fuzzy modeling system by employing it in two benchmarking simulations; namely, (1) the three sets of experiments in Nakanishi's dataset [16] [17]; and (2) the prediction of Mackey-Glass chaotic system [18].

A. Nakanishi Dataset

The learning and modeling abilities of the proposed FAPOP model are evaluated using three sets of experiments; namely, (a) a nonlinear system; (b) the human operation of a chemical plant; and (c) the daily pricing of a stock in a stock market. The datasets are extracted from published papers in [16] and [17]. Each of the three datasets is split into three groups A, B, and C where A and B form the training dataset and C is the testing data. The benchmark for comparisons are the accuracies on the testing data (calculated as the mean squared error MSE) and the correspondence between the computed output with the testing data (calculated as the Pearson correlation coefficient R). The experimental results of FAPOP are subsequently benchmarked against the following models; namely, Hebb-R-R [2]; POPFNN [19]; RSPOP [8]; five of the six reasoning methods in [16] (Sugeno P&P-G, Sugeno P, Sugeno P-G, Mamdani, and Turksen IVCRI); ANFIS [3]; EFuNN [20]; DENFIS [4]; and FITSK [21].

1) A Nonlinear System: The objective of this experiment is to identify and model the underlying principles of a nonlinear system. In the original dataset, there were four inputs x_1–x_4 and one output y. Before the feature analysis process,

TABLE II
CONSOLIDATED EXPERIMENTAL RESULTS FOR THE NAKANISHI DATASET.

	Nonlinear System		Chemical Plant		Stock Prediction		
Model	MSE (Rank)	R (Rank)	MSE (Rank)	R (Rank)	MSE (Rank)	R (Rank)	Average Rank
Hebb-R-R	0.185 (1)	0.911 (2)	2.423×10^4 (2)	0.998 (2)	15.14 (1)	0.947 (1)	1.5
POPFNN	0.270 (3)	0.877 (3)	5.630×10^5 (8)	0.946 (9)	76.22 (10)	0.733 (10)	7.2
RSPOP	0.383 (7)	0.856 (4)	2.124×10^5 (5)	0.983 (7)	24.86 (2)	0.922 (2)	4.5
Sugeno P&P-G	0.345 (6)	0.828 (7)	2.897×10^5 (7)	0.973 (8)	94.58 (12)	0.706 (11)	8.5
Sugeno P	0.776 (12)	0.558 (12)	6.372×10^5 (9)	0.933 (12)	35.47 (4)	0.883 (4)	8.8
Sugeno P-G	0.467 (9)	0.845 (6)	1.931×10^6 (12)	0.990 (6)	168.9 (13)	0.700 (12)	9.7
Mamdani	0.862 (13)	0.490 (13)	6.580×10^5 (10)	0.937 (11)	40.84 (7)	0.865 (7)	10.2
Turksen IVCRI	0.706 (11)	0.609 (11)	2.581×10^5 (6)	0.993 (4)	93.02 (11)	0.661 (13)	9.3
ANFIS	0.286 (4)	0.853 (5)	2.968×10^6 (13)	0.780 (13)	38.06 (5)	0.875 (6)	7.7
EFuNN	0.566 (10)	0.720 (10)	7.247×10^5 (11)	0.946 (9)	72.54 (9)	0.756 (9)	9.7
DENFIS	0.411 (8)	0.805 (9)	5.240×10^4 (4)	0.995 (3)	69.82 (8)	0.810 (8)	6.7
FITSK	0.336 (5)	0.828 (7)	3.862×10^4 (3)	0.993 (4)	33.78 (3)	0.883 (4)	4.3
FAPOP	0.186 (2)	0.951 (1)	1.440×10^4 (1)	0.999 (1)	38.47 (6)	0.898 (3)	2.3

TABLE I
EXPERIMENTAL RESULTS OF FAPOP FOR THE NAKANISHI DATASET.

	Nonlinear System	Chemical Plant	Stock Prediction
# Rules before feature analysis	108	81	17496
# Rules after feature analysis	12	12	6
Rule reduction %	89	85	99
MSE	0.186	1.440×10^4	38.47
R	0.951	0.999	0.898

108 fuzzy rules are initialized in FAPOP; while FAPOP identifies input feature x_3 as redundant after feature analysis. Subsequently, the application problem becomes a 3 inputs-1 output system. A total of twelve fuzzy rules are then identified by FAPOP for this experiment (see Table I). Consolidated experimental results on the benchmarking measures are given in Table II.

2) Human Operation of a Chemical Plant: The proposed FAPOP model is employed to model the human operation of a chemical plant in this experiment. Although there were five inputs x_1–x_5 and one output y in the original dataset, only selected relevant input features, x_1–x_3 and x_5, are used in defining the rules. Subsequently, a total of twelve fuzzy rules are extracted from the FAPOP model as shown in Table I. Consolidated experimental results on the benchmarking measures are given in Table II.

3) Daily Pricing of a Stock in a Stock Market: Using various statistics concerning a stock collected from a stock market, the proposed FAPOP model is employed to perform stock price prediction. Out of the original ten input features x_1–x_{10}, two features, x_7 and x_9, are redundant and discarded from the FAPOP model. A total of six fuzzy rules are then identified by FAPOP, indicating a remarkable reduction of 99% in the number of identified rules (see Table I). For comparison, the consolidated experimental results on the benchmarking measures are given in Table II.

4) Discussion: Table I shows the experimental results for the proposed FAPOP model in the Nakanishi dataset. There are significant reductions in the number of identified rules in all the three experiments. By keeping the number of identified rules low, this helps to improve the overall interpretability of the system.

Table II shows the consolidated experimental results for the Nakanishi dataset between FAPOP and the benchmarking models. FAPOP outperforms all the benchmarking models in the first two tasks by consistently ranking first in terms of MSE and R. In the task of modeling a nonlinear system, FAPOP achieves a MSE value of 0.186, similar to Hebb-R-R. In addition, there is a slight improvement of 4.39% in the R value compared to second place Hebb-R-R. For the second task of modeling a chemical plant, the proposed FAPOP model delivers an outstanding performance by achieving a reduction of 40.6% in terms of calculated MSE and a slight improvement of 0.001 in terms of correlation R compared to the second position Hebb-R-R. For the final task of stock prediction, the performance of FAPOP is comparable with FITSK, Sugeno P and ANFIS. It should also be noted that Hebb-R-R uses 20 fuzzy rules; while RSPOP uses 29 fuzzy rules [2]. Comparatively, FAPOP uses only six rules in the prediction task. This explains the slight compromise in the MSE and R values achieved. On average, the proposed FAPOP model achieves a ranking of 2.3 for all the three tasks, putting it at second position. This is a significant improvement compared to respective members in the family of pseudo outer-product-based neural fuzzy systems, i.e., POPFNN with an average ranking of 7.2, and RSPOP with an average ranking of 4.5. As a further comparison, the training time of the three members in the POP neural fuzzy systems are listed in Table III. As seen, the time complexity of the proposed FAPOP model is comparable with the first generation POPFNN model. This is a direct consequence of the POP rule identification algorithm,

TABLE III
TRAINING TIME(MS) OF MEMBERS OF THE POP NEURAL FUZZY SYSTEMS.

Model	Nonlinear System	Chemical Plant	Stock Prediction
POPFNN	46	485	1,082,866
RSPOP	188	202	184
FAPOP	62	468	1,012,500

where all possible combination of rules is considered, and hence the size of the initial rulebase increases exponentially with an increase in the number of input dimensions. This can be seen in the stock prediction experiment where the initial number of rules is 17496 in FAPOP (see Table II). As a result, it is time consuming and memory dependent at the beginning of the learning. On the other hand, RSPOP proposes a new rule identification algorithm, where the initial identified rulebase consists of only 50 rules prior to rule reduction [8]. This, subsequently, results in a significant saving in the time complexity. In conclusion, one can reduce the time complexity of the proposed FAPOP model by adopting alternative rule identification algorithms [22] [14] which are tailored to the numerical training data.

B. *Prediction of Mackey-Glass Chaotic System*

The learning and generalization abilities of the proposed FAPOP model is evaluated by employing it in a benchmark comparison involving the prediction of a chaotic system. The chaotic system is generated by a delay differential equation as described in (13):

$$\frac{\partial x(t)}{\partial t} = \frac{0.2x(t-\tau)}{1+x^{10}(t-\tau)} - 0.1x(t) \tag{13}$$

which was first investigated by Mackey and Glass [18]. A fourth-order Runge-Kutta method was applied to compute the numerical approximation of the series with conditions as follows: $\tau = 17$ and initial condition $x(0) = 1.2$. Four past values are used to predict the present value, where

$$\begin{aligned} \text{input vector} &= [x(t-24), x(t-18), x(t-12), x(t-6)] \ ; \\ \text{output vector} &= [x(t)] \ . \end{aligned}$$

A total of 1000 data pairs are extracted from the interval $t \in [118, 1117]$. The first 500 pairs are then used as training set; while the remaining 500 pairs are used for testing. The benchmark for comparisons are the accuracies on the testing data (given as the root mean squared error RMSE) and the interpretability of the systems (given as the number of identified fuzzy rules). The experimental results of FAPOP are subsequently benchmarked against the following models; namely, FALCON-ART [1]; FITSK [21]; and EFuNN [20].

Table IV shows the performance comparison for the Mackey-Glass chaotic system between the proposed FAPOP model and the benchmarking models. FAPOP achieves a RMSE value of 0.017, falling behind only the EFuNN model in terms of accuracy achieved on the testing data. Nevertheless, EFuNN uses 330% more rules compared to FAPOP (a jump from 20 to 86) in this prediction task, greatly decreasing the interpretability of the system. On the average, the proposed FAPOP is able to maintain a good balance between the two contradictory requirements, i.e., delivering a good fuzzy rulebase interpretability by identifying 20 fuzzy rules, and achieving a high modeling accuracy of 0.017.

TABLE IV
CONSOLIDATED EXPERIMENTAL RESULTS FOR THE MACKEY-GLASS SYSTEM.

Model	RMSE	# Rules
FALCON-ART	0.040	30
FITSK	0.034	16
EFuNN	0.010	86
FAPOP	0.017	20

V. CONCLUSION

This paper presents a novel neural fuzzy system named *Feature Analysis Enhanced Pseudo-Outer Product Fuzzy Rule Identification System* (FAPOP). A key strength of FAPOP is its feature analysis integrative design such that a regulated balance is achieved between the two contradictory interpretability-accuracy requirements in the design of a neural fuzzy system. This is done via two channels; namely, (1) improve the interpretability of the system by identifying features relevant to its computational structure; and (2) improve the accuracy of the system by identifying features relevant to the application problem. Subsequently, the learning mechanisms in the proposed FAPOP model are showcased through a case study where the system is able to identify a compact and highly interpretable rulebase by identifying relevant features to the system and application. Finally, FAPOP is employed in a series of benchmarking simulations to demonstrate its effectiveness as a neural fuzzy modeling system, and excellent performances have been achieved.

REFERENCES

[1] C. J. Lin and C. T. Lin, "An ART-based fuzzy adaptive learning control network," *IEEE Transactions on Fuzzy Systems*, vol.5, no.4, pp.477–496, Nov. 1997.

[2] F. Liu, C. Quek and G. S. Ng, "A novel generic hebbian ordering-based fuzzy rule base reduction approach to Mamdani neuro-fuzzy system," *Neural Computations*, vol.19, no.6, pp.1656–1680, Jun. 2007.

[3] J. S. R. Jang, "ANFIS: Adaptive-network-based fuzzy inference systems," *IEEE Transactions on Systems, Man and Cybernetics*, vol.23, no. 3, pp.665–685, May 1993.

[4] N. Kasabov and Q. Song, "DENFIS: Dynamic evolving neural-fuzzy inference system and its application for time-series prediction," *IEEE Transactions on Fuzzy Systems*, vol.10, no.2, pp.144–154, Apr. 2002.

[5] E. H. Mamdani, "Application of fuzzy logic to approximate reasoning using linguistic systems," *IEEE Transactions on Computers*, vol.26, no.12, pp.1182–1191, Dec. 1977.

[6] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE Transactions on Systems, Man and Cybernetics*, vol.15, no.1, pp.116–132, 1985.

[7] M. Sugeno and G. T. Kang, "Structure identification of fuzzy model," *Fuzzy Sets and Systems*, vol.28, no.1, pp.15–33, Oct. 1988.

[8] K. K. Ang and C. Quek, "RSPOP: Rough set-based pseudo outer-product fuzzy rule identification algorithm," *Neural Computations*, vol.17, no.1, pp.205–243, Jan. 2005.

[9] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol.3, pp.1157–1182, Jan. 2003.
[10] N. R. Pal, "Soft computing for feature analysis," *Fuzzy Sets and Systems*, vol.103, no.2, pp.201–221, Apr. 1999.
[11] C. Quek and R. W. Zhou, "The POP learning algorithms: Reducing work in identifying fuzzy rules," *Neural Networks*, vol.14, no. 10, pp.1431–1445, Dec. 2001.
[12] S. Haykin, *Neural networks: A comprehensive foundation.* Prentice Hall, 1998.
[13] C. T. Lin and C. S. G. Lee, *Neural fuzzy systems: A neuro-fuzzy synergism to intelligent systems.* Prentice Hall, 1996.
[14] S.W. Tung, C. Quek and C. Guan, "SaFIN: A self-adaptive fuzzy inference network," In preparation.
[15] D. Chakraborty and N. R. Pal, "Integrated feature analysis and fuzzy rule-based system identification in a neuro-fuzzy paradigm," *IEEE Transactions on Systems, Man and Cybernetics–Part B*, vol.31, no.3, pp. 391–400, Jun. 2001.
[16] H. Nakanishi, I. Turksen and M. Sugeno, "A review and comparison of six reasoning methods," *Fuzzy Sets and Systems*, vol.57, no.3, pp.257–294, Aug. 1993.
[17] M. Sugeno and T. Yasukawa, "A fuzzy-logic-based approach to qualitative modeling," *IEEE Transactions on Fuzzy Systems*, vol.1, no.1, pp.7–31, Feb. 1993.
[18] M. Mackey and L. Glass, "Oscillation and chaos in physiological control systems," *Science*, vol.197, no. 4300, pp.287–289, Jul. 1977.
[19] C. Quek and R. W. Zhou, "POPFNN: A pseudo outer-product based fuzzy neural network," *Neural Networks*, vol.9, no.9, pp.1569–1581, Dec. 1996.
[20] N. Kasabov, "Evolving fuzzy neural networks for supervised/ unsupervised online knowledge-based learning," *IEEE Transactions on Systems, Man and Cybernetics–Part B*, vol.31, no.6, pp.902–918, Dec. 2001.
[21] K. H. Quah and C. Quek, "FITSK: Online local learning with generic fuzzy input Takagi-Sugeno-Kang fuzzy framework for nonlinear system estimation," *IEEE Transactions on Systems, Man and Cybernetics–Part B*, vol.36, no.1, pp.166–178, Feb. 2006.
[22] W. L. Tung and C. Quek, "GenSoFNN: A generic self-organizing fuzzy neural network," *IEEE Transactions on Neural Networks*, vol.13, no.5, pp. 1075–1086, Sep. 2002.