Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.12785/amis/090340

Study on the Evaluation of UAV Disaster Monitoring System Architecture based on the RSBFNN Algorithmic Method

Zhifei Li∗*, Yifan Zhu, Feng Yang and Wenguang Wang*

College of Information Systems and Management, National University of Defense Technology, 410073 Changsha, Hunan, P. R. China

Received: 10 Aug. 2014, Revised: 11 Nov. 2014, Accepted: 12 Nov. 2014
Published online: 1 May 2015

Abstract: In disaster monitoring, Unmanned Aerial Vehicles (UAVs) are becoming more suitable than earth observing satellites and manned helicopters, especially in emergent situations. However, decisions must be made as to which attributes the UAV must possess when developing new models. Combined with NOLH experiment design and multi-agent simulation (MAS) technology, a new architecture evaluation method, termed Rough Set Based Fuzzy Neural Network (RSBFNN), is proposed to simulate and analyze both the UAV attributes and desired effectiveness. Experimental results show that this method is very dependable when comparing predicted and actual responses and yields better performance than other models. This method best fits the design of the disaster monitoring UAV, and can be used to perform a series of dynamic trade studies, in which various architecture alternatives are examined and compared.

Keywords: Disaster Monitoring, UAV, System Architecture, MAS, Rough Set, Fuzzy Neural Network

1 Introduction

In recent years, the outbreak frequency and scale of natural or accidental disasters have been significantly larger than usual [1]. In China, there have been more than one million casualties and up to hundreds of billions of dollars in economic losses caused by natural disasters, accidental disasters, and public emergencies every year. Disaster monitoring is of great importance in preparing for the deployment of rescue operations and reconstruction in stricken areas after a disaster occurs [2,3,4]. Unmanned Aerial Vehicles (UAVs) are more suitable for disaster monitoring than earth observing satellites and manned helicopters, particularly in emergent situations [5]. After decades of development, the UAV is relatively mature from a technical standpoint, with advantages such as low cost, ease of manipulation, high agility, and adaptability. Additionally, the UAV can carry important equipment to complete certain special tasks from the air, such as aerial surveillance, aerial propaganda, or emergency rescue. The UAV has already played a critical role in dealing with natural disasters, accidents, and public security events. This paper focuses on the use of Unmanned Aerial Vehicles for disaster monitoring. In the process of UAV concept design, finding the ideal methods to vary parameters of system architectures and compare competing solutions through modeling and simulation is an urgent problem that must be solved.

Multivariate regression has been used to express the relationships between a product's physical parameters and the subsequent responses [6,7,8], but it is inadequate to capture relationships, because the approach assumes linearity, making it inappropriate to analyze engineering data that contain large amounts of noise. Soft computing is probably the most appropriate way to identify nonlinear relationships between the design parameters of a product and the results. Recently, the use of soft computing techniques to map responses to design parameters has emerged as a substantial field of research. Soft computing is a collection of fuzzy logic, neurocomputing, evolutionary computing, and probabilistic computing; the aim is to exploit the tolerance for imprecision and uncertainty, in order to achieve tractable, robust, and low-cost solutions [9]. There have been many successful examples of the use of soft computing techniques in engineering design using fuzzy rule-based models [10,11,

∗ Corresponding author e-mail: lee.nudt@gmail.com

12], rough sets [13,14,15], neural networks [16,17,18], and association rule mining [19].

Among the classical simulation data analysis and alternative evaluation methods, fuzzy logic is easy to understand, and the neural network is better suited to adaptive learning [20]. Fuzzy methods are suitable under incomplete data conditions and require fewer observations than other models. Fuzzy theory [21,22] was originally developed to deal with problems involving linguistic terms and has been successfully applied to financial time series forecasting [23,24]. In contrast to traditional model-based methods, ANNs are data-driven, self-adaptive methods, in that there are few *a priori* assumptions made about the models for the problems being studied. They learn from examples and capture subtle functional relationships among the data, even if the underlying relationships are unknown or are hard to describe. Despite the advantages of ANNs, however, these models have some weaknesses. The most important of these is data limitation. No definite rule exists for the sample size requirement of a given problem. The amount of data for network training depends on the network structure, the training method, and the complexity of the particular problem or the amount of noise in the data at hand [25]. Recently, more hybrid models have been proposed using ARIMA, Artificial Neural Networks, and fuzzy logic with good prediction performance. Many studies have also reported the development of a number of hybrid models that integrate fuzzy techniques with forecasting methods, in order to improve accuracy. Chang *et al.* [26], developed a hybrid model by integrating the Self Organization Map neural network, GAS and a fuzzy rule base to forecast the future sales of a printed circuit board factory. Lin and Cobourn [27] combined the Takagi–Sugeno fuzzy system and a nonlinear regression model for time series forecasting. Pai [28] proposed a hybrid ellipsoidal fuzzy system for a time series forecasting model, in order to forecast regional electricity loads in Taiwan. Huarng and Yu [29] described a combined methodology using neural networks to forecast fuzzy time series.

Currently, the problem of how to combine the strengths of fuzzy and neural networks technology, in order to improve learning and expressing abilities, is a subject of great concern. The fuzzy neural network is a new technology in this context; it brings together the advantages of neural networks and fuzzy systems. The basic point of the fuzzy neural network is to deal with uncertain fuzzy systems, artificial neural network connective structure, and learning methods to make a fuzzy neural network with the combined capabilities of fuzzy expression, adaptive learning, and distributed information processing. However, the fuzzy neural network has not yet been formed into a unified theoretical system and requires further study. For one thing, the learning process may bring a large computation load, owing to the overly complex network structure. Additionally, the construction of network structure demands a vast expert knowledge of the field. In data mining, the source for knowledge acquisition is important, as well as its representation, its method for reasoning, and its decision-making method for large numbers of observations and experimental simulation data. This is particularly true for inaccurate and incomplete data or data with no prior knowledge. Rough set theory and the fuzzy neural network have become important research tools in this field [30].

In this paper, we combine rough set theory and fuzzy neural network technology, use the indiscernibility relation and knowledge reduction of rough set to streamline a simplified rule from a large number of original data, establish the fuzzy neural network model and determine the connection between the hidden layer nodes, which can give the network a good topology from the outset and thus greatly reduce the complexity of the network. This **R**ough **S**et **B**ased **F**uzzy **N**eural **N**etwork (RSBFNN) displays quick learning and strong fault tolerance abilities, and, furthermore, the model is interpretable. The remainder of this paper is organized as follows. Section 2 provides the general framework of the RSBFNN, along with the algorithm of the proposed method. In Section 3, a case study for UAV system architecture evaluation is presented. Finally, Section 4 concludes the paper and presents future work.

2 RSBFNN Algorithmic Method

2.1 The general framework of the method

The basic idea of UAV system architecture evaluation based on the RSBFNN method is specified as follows. The architecture representation is used to structure the modeling and simulation environment, which consists of agent-based models created in a multi-agent simulation (MAS) tool. A design of experiments (DoE) is then wrapped around the MAS model, in order to obtain a representative sampling of the multi-dimensional design space. This DoE and MAS tool provides the necessary data to create a series of decision rules for each of the key responses, with respect to the varied inputs in the actual model.

Firstly, the simulation output data stored in the database is preprocessed, and the missing data is filled in or the invalid data is removed to obtain the original data decision table. Next, the original data is discretized and normalized to obtain the minimum set of rules covering the original decision sample characteristics with the greatest degree of completeness, using rough set attribute reduction. Finally, the initial fuzzy neural network topology is determined and the network structure with the test data is trained and adjusted to get a comprehensive evaluation model with the optimal structure. The overall flow chart of the RSBFNN to UAV architecture evaluation is shown in Fig. 1.

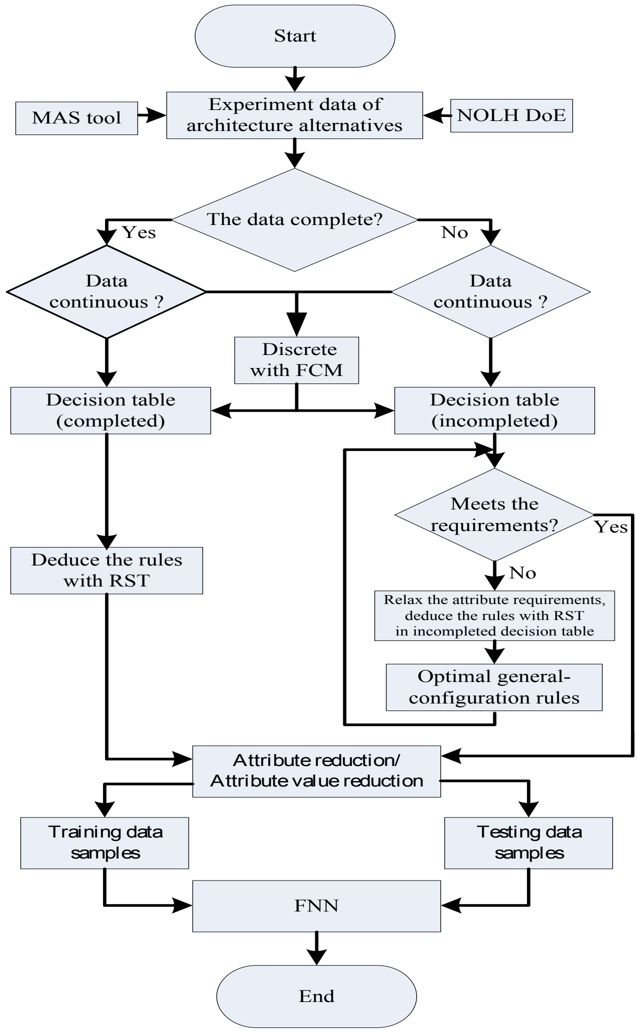

Fig. 1: The RSBFNN overall flow chart

2.2 Rough set representation

2.2.1 UAV architecture definition

UAV architecture can be defined as:

$$S(U,A,V,f) \tag{1}$$

where U is a non-empty set of alternatives, A is a set of non-empty attributes of a selected configuration, V is the range of α, $\alpha \in A$, and f is an information function, $f : U \to V_\alpha$, giving each attribute of each object an information value, where $\alpha \in A$, $x \in U$, $f(x,a) \in V_\alpha$.

The decision table for UAV system architecture is defined as follows:

$$S = (U, A \cup \{d\}, V, f) \tag{2}$$

where U, A, V, and f have the same meaning as the UAV architecture model, and $\{d\}$ is a decision attribute. Therefore, designers can get $\{d\}$ attribute values from the DoE result of the interested attribute range.

2.2.2 The Upper and Lower Approximation of UAV architecture

In the UAV architecture model, each attribute subset $M \subseteq A$, $IND(M)$ expresses the meta-relationship between any two alternatives, called indiscernible relations, which are defined as follows:

$$IND(M) = \{(x,y) \in U \times U | \forall \alpha \in M, \alpha(x) = \alpha(y)\} \tag{3}$$

where, $M \subseteq A$ (M is a subset of the entire attribute A), and $X \subseteq U$ (X is a subset of all optional alternatives, U).

For X, the upper and lower approximation of M is defined as:

$$\underline{M}X = \cup\{Y \in U/IND(M) | Y \subseteq X\} \tag{4}$$

$$\bar{M}X = \cup\{Y \in U/IND(M)|YIX \neq \emptyset\} \quad (5)$$

As seen from the definitions, for the selected architecture alternatives M, the lower approximation represents the minimum optional architecture alternatives set similar to M and the upper approximation represents the maximum optional architecture alternatives set similar to X.

2.2.3 The division matrix and division function in UAV architecture alternatives

The division matrix of selected attributes M in the configuration decision tables is defined as follows:

$$(C_{ij}) = \{\alpha \in M|\alpha(x_i) \neq |\alpha(x_j)\} \quad \text{for } i,j = 1,2,\ldots,n \quad (6)$$

The division function is defined as follows:

$$f(M) = \prod \sum_{(x,y)\in U\times U} \alpha(x,y) \quad (7)$$

The division matrix and division function are used to infer the smallest reduction, which is a small subset of the attributes that can reflect implicit relationships in the selected configuration decision tables.

2.3 Discretization with FCM

We use the *Fuzzy C-Means* method to discrete the continuous data. The definition of *FCM* is summarized as follows [31]:

$X = \{x_1,x_2,\ldots,x_n\}$, sampling set of an attribute;
$x_j = (x_{j1},x_{j2},\ldots,x_{jk})$, j-th k-dimensional vector of each attribute;
c, the number of clusters that are specified;
v_i, the center of the i-th cluster;

$$v_i = \frac{\sum_{j=1}^{n}(u_{ij})^q x_j}{\sum_{j=1}^{n}(u_{ij})^q} \quad (8)$$

$V = (v_1,v_2,\ldots,v_c)$, center vector composed of a cluster center;
q real number greater than 1;
u_{ij}, weight index, which controls the fuzziness of the attribute clustering;
ε, termination condition determined by the engineering staff;
$||x_j - v_i||^2$, Euler distance of j-th attribute and the cluster center

The definition of the membership function of each attribute vector to each attribute cluster:

$$u_{ij} = \frac{\left[\frac{1}{||x_j-x_i||^2}\right]^{1/(q-1)}}{\sum_{k=1}^{c}\left[\frac{1}{||x_j-x_k||^2}\right]^{1/(q-1)}} \quad (9)$$

In the process of discretization of continuous data, the minimal value of the following objective function is required:

$$J_q(u_{ij},v_k) = \sum_{j=1}^{n}\sum_{i=1}^{c}(u_{ij})^q||x_j - v_i||^2; c \leq n \quad (10)$$

The application procedures are summarized as follows:

Step 1: Determine the target that needs to be analyzed and the related attributes that need to be discretized.
Step 2: Determine a set of sampling point of the attributes $X = \{x_1,x_2,\ldots,x_n\}$ and j-th k-dimensional vector of each attribute sampling point.
Step 3: After discretization of the configuration attributes, allocate the values of c, q and ε.
Step 4: Initialize the membership function matrix u_{ij}^0, which represents the distance of each configuration attribute point to the initial cluster center.
Step 5: Use u_{ij}^0 and v_i to upgrade the center of each configuration property cluster.
Step 6: Calculate $u_{ij}^{(L+1)}$, which represents the relationship of each configuration attribute point to its center.
Step 7: If $\max\left[||u_{ij}^{(L)} - u_{ij}^{(L+1)}\right] \leq \varepsilon$, then stop iteration, otherwise return to step 5.

2.4 T-S modeling

Let the inputs $x = [x_1,x_2,\ldots,x_n]^T$. x_i is the set of fuzzy linguistic variables, set as $T(x_i) = \{A_i^1,A_i^2,\ldots,A_i^{m_i}\}$, $i = 1,2,\ldots,n$. Where $A_i^{s_i}$ $(s_i = 1,2,\ldots,m_i)$ is the s_i-th linguistic variable values of x_i, which is a fuzzy set defined on the universe of discourse, the corresponding membership is $u_{A_I^m}$. Let the output vector, $y = [y_1,y_2,\ldots,y_n]^T$; then the Takagi–Sugeno model [32] is:

$$\begin{aligned} R_j :& \text{ if } \left(x_1 \text{ is } A_1^{s_{1j}}\right) \text{ and } \left(x_2 \text{ is } A_2^{s_{2j}}\right) \text{ and } \ldots \\ & \text{and } \left(x_n \text{ is } A_n^{s_{nj}}\right) \\ & \text{then } \begin{cases} y_{1j} = p_{j0}^1 + p_{j1}^1 x_1 + \ldots + p_{jn}^1 x_n \\ y_{kj} = p_{j0}^r + p_{j1}^r x_1 + \ldots + p_{jn}^r x_n \end{cases} \end{aligned} \quad (11)$$

$$y_k = \sum_{j=1}^{m} a_j y_{kj} \Big/ \sum_{j=1}^{m} a_j = \sum_{j=1}^{m} \bar{a}_j y_{kj} \quad (12)$$

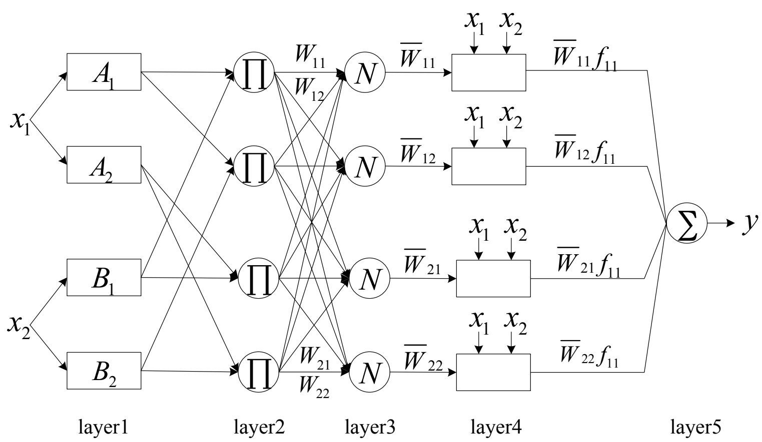

Fig. 2: The structure of FNN

where, R_j is the j-th fuzzy rules;

$a_j = \mu_{A_1}^{s_{1j}}(x_1) \wedge \mu_{A_2}^{s_{2j}}(x_2) \wedge \ldots \wedge \mu_{A_n}^{s_{nj}}(x_n)$, $\wedge$ is fuzzy computing which requires a small operation: $\bar{a}_j = a_j / \sum_{j=1}^{m} a_j$, $j = 1,2,\ldots,m$, $m \leq \prod_{i=1}^{n} m_i$, $k = 1,2,\ldots,r$.

2.5 Construction of the FNN

According to the Takagi–Sugeno model, the network structure can be designed as shown in Fig. 2. The network has five layers.

Layer 1: Each node I in this layer is adaptive with a note function, which is the membership function $\mu_i^{s_i}$, through which input components belong to each linguistic variable value fuzzy set. Membership functions can be any type of appropriate parameterized membership function, such as when to the Gaussian function, the output of this layer, is

$$O_{is_i}^1 = \mu_i^{s_i}(x_i) = e^{-\frac{1}{2}\left(\frac{x_i - c_{is_i}}{\sigma_{is_i}}\right)^2} \tag{13}$$

where, x_i input variables;

$s_1 = 1,2,\ldots,m_i$, $i = 1,2,\ldots,n$, m_i is fuzzy partition number of x_i; $\{c_{is_i}, \sigma_{is_i}\}$ is a parameter set. When the values of these parameters change, the Gaussian function will change; this shows the different forms of membership functions of fuzzy sets. The parameter in this layer is the premise parameter.

Layer 2: Each node output of this layer represents the incentive intensity of the rule, the rule node will perform a fuzzy *AND* operation, the output is:

$$\begin{aligned} O_j^2 &= \min\{O_{1s_{1j}}^1, O_{2s_{2j}}^1, \ldots, O_{ns_{nj}}^1\} \\ &= \min\{\mu_1^{s_{1j}}, \mu_2^{s_{2j}}, \ldots, \mu_n^{s_{nj}}\} \end{aligned} \tag{14}$$

$j = 1,2,\ldots,m$, $m = \prod_{i=1}^{n} m_i$.

Layer 3: The j-th node of this layer calculates ratios of incentive intensity of the j-th rule divided by the whole incentive intensity of all the rules, which is the output of this layer, also known as the normalized incentive intensity.

$$O_j^3 = O_j^2 / \sum_{j=1}^{m} O_j^2, j = 1,2,\ldots,m \tag{15}$$

Layer 4: Each node in this layer is an adaptive node that has node function. The output is:

$$O_j^4 = \sum_{j=1}^{m} y_{kj} O_j^3, k = 1,2,\ldots,r \tag{16}$$

The parameter $\{p_{ji}^k\}$ in y_{kj} is a parameter set of this node. Parameters in the layer are conclusion parameters.

Layer 5: This layer calculates the sum of all the signals as the total output.

$$O_j^5 = \sum_{k=1}^{r} O_k^4, k = 1,2,\ldots,r \tag{17}$$

2.6 Learning algorithm of the neural network

In this paper, the learning algorithm of neural networks is the gradient descent of the BP algorithm [33]. Let the residual of cost function be:

$$E = \frac{1}{2} \sum_{r=1}^{R} (Y - y)^2 \tag{18}$$

where, R is the number of learning samples, Y is system expected output value, and y is the actual output value of the network.

Train the connection power W_{1m} between the forth and the fifth layer through the BP algorithm, and the center C_{ij} of the membership function and the width σ_{ij}^2, and the specific learning algorithm is:

$$\frac{\partial E}{\partial w_{lm}} = \frac{\partial E}{\partial y}\frac{\partial y}{\partial w_{lm}} = -(Y-y)g_m^{(4)} \tag{19}$$

$$\delta_m^{(4)} = -\sum_{r=1}^{R}\frac{\partial E}{\partial f_l^{(5)}}\frac{\partial f_l^{(5)}}{\partial g_m^{(4)}}\frac{\partial g_m^{(4)}}{\partial f_m^{(4)}} = \sum_{r=1}^{R}(Y-y)w_{lm} \tag{20}$$

$$\begin{aligned}\delta_k^{(3)} &= -\sum_{r=1}^{R}\frac{\partial E}{\partial f_m^{(4)}}\frac{\partial f_m^{(4)}}{\partial g_k^{(3)}}\frac{\partial g_k^{(3)}}{\partial f_k^{(3)}}\\ &=\delta_m^{(4)}\sum_{m=1}^{M}\alpha_{lm}\Big/\left(\sum_{m=1}^{M}\alpha_{lm}\right)^2\end{aligned} \tag{21}$$

$$\delta_j^{(2)} = -\sum_{k=1}^{K}\delta_k^{(3)}\frac{\partial f_k^{(3)}}{\partial g_j^{(2)}}g_j^{(2)} \tag{22}$$

$$\frac{\partial E}{\partial c_{ij}} = -\delta_j^{(2)}\frac{2(x_i-c_{ij})}{\sigma_{ij}^2} \tag{23}$$

$$\frac{\partial E}{\partial \sigma_{ij}} = -\delta_j^{(2)}\frac{2(x_i-c_{ij})^2}{\sigma_{ij}^2} \tag{24}$$

The new learning algorithm with the adjusted parameter is:

$$W_{lm}(k+1) = w_{lm}(k) - \beta\frac{\partial E}{\partial w_{lm}} + \alpha\Delta w_{lm}(k) \tag{25}$$

$$c_{ij}(k+1) = c_{ij}(k) - \beta\frac{\partial E}{\partial c_{ij}} + \alpha\Delta C_{ij}(k) \tag{26}$$

$$\sigma_{ij}(k+1) = \sigma_{ij}(k) - \beta\frac{\partial E}{\partial \sigma_{ij}} + \alpha\Delta\sigma_k(k) \tag{27}$$

In the above formula, β represents the learning efficiency, and α represents momentum coefficient.

3 Case study

3.1 Problem description

In order to illustrate this method, a representative notional example was created and used to execute a proof-of-concept on the proposed process. The scenario is summarized as follows.

The state has identified a capability gap in detecting and monitoring widely distributed ground disasters. An initial analysis of alternatives shows that UAVs are a potential solution for closing this gap. However, there is some debate over which attributes the UAV must possess. Some believe that placing existing sensors on an improved platform (greater speed, range, and RCS than existing platforms) will close the capability gap. Others believe that placing improved sensors on an existing platform will be more effective. There is not available funding to do both and the risk associated with developing a new platform and new sensors has been deemed unacceptable. A decision will have to be made as to which technical approach is best.

3.2 Data preparation

In this typical disaster-monitoring scenario, after comprehensively considering the technical characteristics of the UAV monitoring system, some interesting attributes are identified:

– Number of UAVs (*Num-U*)
– Sensor Angle (*Sen-Ang*)
– Sensor Range (*Sen-Ran*)
– Sensor width (*Sen-Wid*)
– Radar Power (*Rad-Pow*)
– Radar cross section (*RCS*)
– Speed (*Spe*)
– Max endurance (*M-Endu*)
– Maximum range (*Max fly range FR*)
– Maximum altitude (*M-Fl-Hei*)
– Weight (*Weight*)
– Probability of reliability (*Pro-re*)

Note that these simplified variables of UAVs and their ranges were developed to create a representative example, and were not fully developed, as they would be for an actual program.

Table 1: Variables of UAVs and their ranges

Variable	Min	Max	Variable	Min	Max
Num-U	1	12	*Spe*	150km/h	550km/h
Sen-Ang	90°	360°	*M-Endu*	35min	60min
Sen-Ran	25km	90km	*M-Fl-Ran*	200km	750km
Sen-Wid	40°	120°	*M-Fl-Hei*	500m	2000m
Rad-Pow	3	8	*Weight*	80kg	200kg
RCS	1	7	*Pro-re*	0.8	1

We used a multi-agent simulation tool and ensured that the model was working correctly; it was then time to run the experiment. In the case of this paper, we examined 65 different combinations by implementing the NOLH design [34], using the ranges of factor levels specified in Table 1. The scatterplot matrix of Fig. 3 shows the orthogonality and space-filling properties of the NOLH design used for the 12 variables. Although some factors took on a limited number of discrete levels, which tends to degrade the orthogonality, the maximum pair wise correlation between any two columns was less than 0.1.

Based on the multi-agent modeling and simulation, we developed software that supported the agent-based

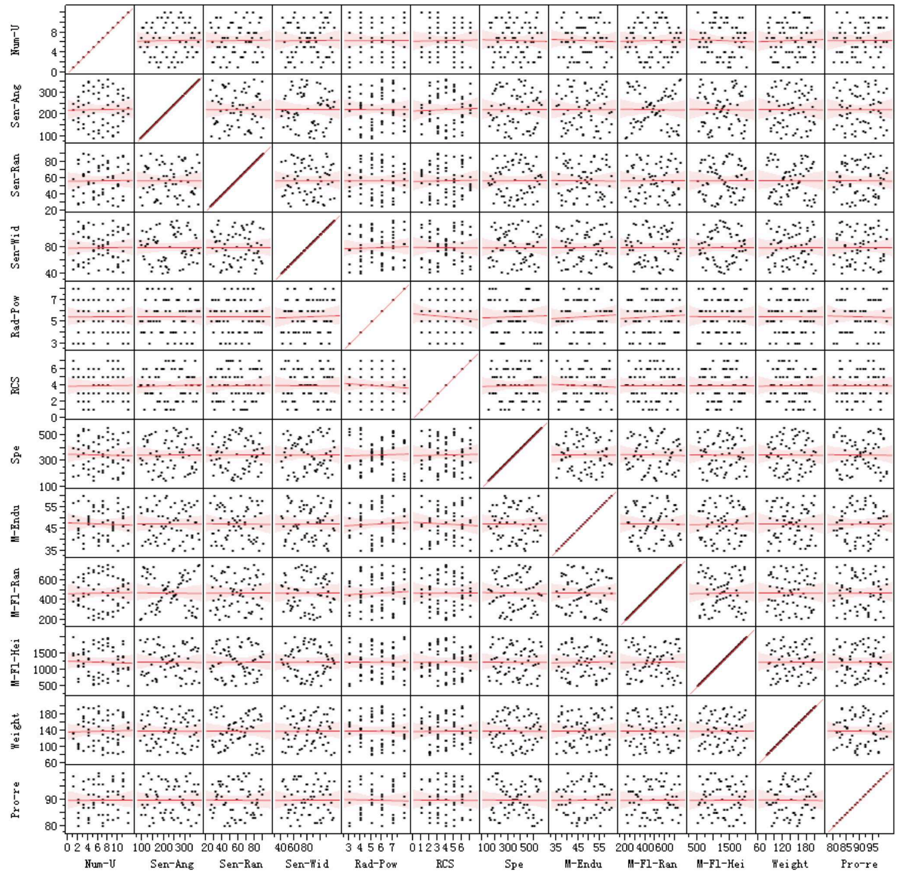

Fig. 3: The Scatterplot Matrix of the NOLH DoE result

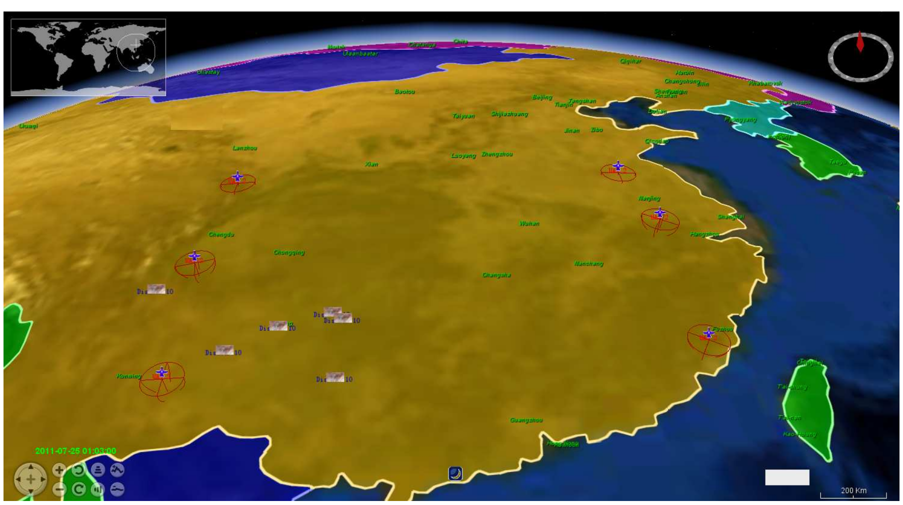

Fig. 4: Screen Shot of the Scenario in the MAS Tool

simulation. These 65 combinations were then entered into the multi-agent simulation tool to get the response effectiveness, measured by the number of disaster districts that the UAVs have monitored and detected. Fig. 4 is a screen shot of the scenario in the MAS tool.

Table 2: Data samples

	Num-U	*Sen-Ang*	*Sen-Ran*	*Sen-Wid*	*Rad-Pow*	*RCS*	*Spe*	*M-Endu*	*M-Fl-Ran*	*M-Fl-Hei*	*Weight*	*Pro-re*	*Num-Det-Disa*
1	9	103	48	66	4	6	469	47	733	1578	146	99	1
2	11	284	32	74	5	3	369	54	595	1883	172	90	4
3	11	187	87	58	4	6	213	46	423	1367	176	98	7
4	8	330	72	76	3	4	256	42	269	1977	189	93	8
5	11	217	37	41	3	2	250	50	535	1227	88	97	4
...	...	...	...	...	...	...	...	...	...	...	...	...	...
61	2	132	49	54	3	5	500	44	346	1461	155	80	2
62	2	280	82	71	5	6	388	58	630	711	95	89	1
63	4	107	68	61	4	1	413	47	716	828	127	87	8
64	5	326	44	75	3	5	244	41	243	547	103	89	8
65	2	183	29	51	4	3	188	43	389	1086	101	85	1

Table 3: Information about the decision table after FCM

	Num-U	*Sen-Ang*	*Sen-Ran*	*Sen-Wid*	*Rad-Pow*	*RCS*	*Spe*	*M-Endu*	*M-Fl-Ran*	*M-Fl-Hei*	*Weight*	*Pro-re*	*Num-Det-Disa*
1	4	1	2	2	2	5	4	3	5	4	3	5	1
2	5	4	1	3	3	2	3	4	4	5	4	3	3
3	5	2	5	2	2	5	1	3	3	3	5	5	5
4	4	5	4	3	1	3	2	2	1	5	5	4	5
5	5	3	1	1	1	1	2	4	4	3	1	5	3
...	...	...	...	...	...	...	...	...	...	...	...	...	...
61	1	1	2	1	1	4	5	2	2	4	4	1	1
62	1	4	5	2	3	5	3	5	4	1	1	3	1
63	2	1	4	2	2	1	4	3	5	2	2	2	5
64	2	5	2	3	1	4	2	2	1	1	1	3	5
65	1	2	1	1	2	2	1	2	2	2	1	2	1

Table 4: Information about the decision table after reduction

	Num-U	*Sen-Ang*	*Sen-Ran*	*Sen-Wid*	*Rad-Pow*	*RCS*	*Spe*	*M-Endu*	*M-Fl-Ran*	*M-Fl-Hei*	*Weight*	*Pro-re*	*Num-Det-Disa*
1	1	2	*	2	5	4	*	5	4	*	5	1	1
2	*	5	2	2	*	1	3	3	3	5	*	5	3
3	3	*	1	1	1	2	*	4	3	*	5	3	5
4	5	2	*	2	4	1	*	5	*	3	5	2	5
5	*	3	1	2	*	5	3	*	3	2	5	4	3
...	...	...	...	...	...	...	...	...	...	...	...	...	...
46	1	5	2	*	3	1	4	1	2	4	4	3	5
47	*	1	2	1	1	4	*	2	2	4	4	*	1
48	1	4	*	2	3	5	3	5	4	1	1	3	1
49	2	5	2	3	1	4	2	2	*	1	1	3	5
50	1	*	1	1	2	2	*	2	2	2	*	2	1

3.3 Data processing

After the simulation ran of a total of 150 cases (30 repetitions for each case), the initial decision table was configured and is shown in Table 2 (only a portion of all of the data are shown, owing to limits of paper length).

Before attribute reduction, the continuous attribute data had to be discretized. Using the method in section 2, the data samples were discretized into 5 clusters. The information about the decision table after discretization is shown in Table 3.

3.4 Attribute and value reduction

The degree of attribute dependability between A, and $B \subseteq U$ is defined as follows:

$$r_B(A) = \frac{card(POS_B(A))}{card(U)} \tag{28}$$

$$POS_B(A) = \cup_{X \in U/IND(A)} \underline{B}(X) \tag{29}$$

Table 5: Comparison of the Out Value and Out Error between different models

	Actual Value	RSBFNN		BPNN		FCEM		SRM	
		Out Value	Out Error	Out Value	Out Error	Out Value	Out Error	Out Value	Out Error
1	4	4.300	0.075	4.500	0.125	4.450	0.113	5.050	0.263
2	6	5.700	0.050	5.860	0.023	6.320	0.053	6.330	0.055
3	2	2.500	0.250	2.630	0.315	2.800	0.400	2.940	0.470
4	8	7.500	0.063	7.520	0.060	7.750	0.031	8.710	0.089
5	4	4.400	0.100	5.010	0.253	5.500	0.375	4.650	0.163
6	8	8.400	0.050	8.620	0.077	8.540	0.067	8.780	0.097
7	4	3.700	0.075	3.950	0.013	4.400	0.100	4.050	0.013
8	2	2.400	0.200	3.220	0.610	2.500	0.250	2.650	0.325
9	6	5.800	0.033	6.220	0.037	6.300	0.050	5.830	0.028
10	8	7.900	0.013	8.240	0.030	8.410	0.051	8.340	0.043
11	2	2.600	0.300	2.510	0.255	2.200	0.100	2.650	0.325
12	1	1.200	0.200	1.120	0.120	1.050	0.050	1.120	0.120
13	8	7.500	0.063	7.530	0.059	8.320	0.040	8.370	0.046
14	8	8.200	0.025	8.670	0.084	8.230	0.029	8.540	0.067
15	1	1.300	0.300	1.550	0.550	1.450	0.450	1.330	0.330

Different attributes play different roles in the interdependencies between condition and decision attributes.

When a is added into B, the attribute importance of classification $U/IND(A)$ is defined as:

$$SGF(a,B,A) = \gamma_B(A) - \gamma_{B-\{a\}}(A) \tag{30}$$

Based on attribute and value reductions, the new decision data are created after reduction, which is according to the Best Probability Rule, as in Table 2. The information about the decision table after reduction is shown in Table 4, which has 50 rules.

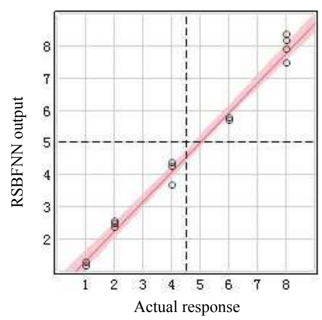

Fig. 5: Scatter diagrams showing actual and predicted responses

3.5 Experimental results and discussion

The data from the 50 group after reduction are divided into two parts, one for network training (35 groups) and one used to test the performance of the network (the last 15 groups).

Figs. 5 and 6 show the predicted outputs of the RSBFNN models against their corresponding actual responses. It was found that best fit lines (red lines) were situated close to unit lines. High correlations between actual and predicted values were observed for the models for the number of disaster districts that the UAVs detected. This shows that predicted responses of the RSBFNN models have a good match with actual responses.

We used different methods to analyze the same simulation out data, resulting in the conclusion that the RSBFNN algorithmic method yielded better performance than the Stepwise Regression Model (SRM), Back Propagation Neural Network (BPNN), and the Fuzzy Comprehensive Evaluation Model (FCEM). The contrast results are listed in Table 5.

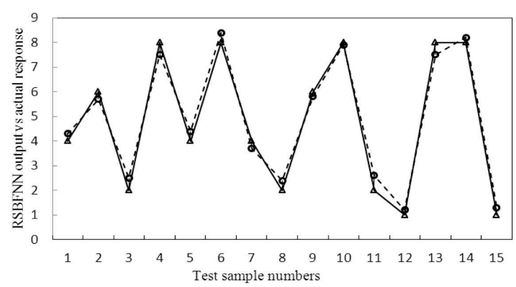

Fig. 6: Predictions of the 15 samples from the RSBFNN model in comparison to actual responses

Once the RSBFNN was verified to the greatest degree possible, the inputs to the neural network could be varied and the resulting outputs could be instantly obtained. This allowed the attributes of disaster monitoring UAVs to be varied to any value within the design space covered by the RSBFNN; the results that would typically be predicted by the agent-based model could be obtained without having to actually perform the time-consuming model execution, as shown in Fig. 7. Thus, the UAV's developers and designers can do a series of dynamic trade studies in which various architecture alternatives are examined and compared.

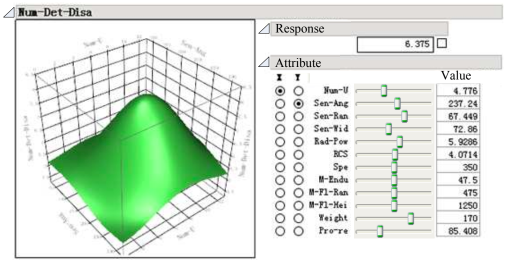

Fig. 7: Three Dimensional RSBFNN Model

4 Conclusion

This paper proposed a new fuzzy neural network based on rough set theory, which combines the advantages of the rough set theory and fuzzy neural networks. Taken together with NOLH experimental design and multi-agent simulation technology, the simulation output data can be effectively processed. Attribute reduction and rule extraction using rough set theory can take full advantage of the characteristics of the sample data, so that the network structure has good initial topology. This method can greatly reduce the size of the network and provides a great deal of *a priori* knowledge based on rough set theory to the initial weights set of fuzzy neural network, making the system error very small at the beginning. Simulation experiments proved the feasibility of this approach and the high practical value in UAV architecture evaluation and simulation data mining.

Acknowledgement

This research was supported, in part, by the National Natural Science Foundation of China, and a project supported by scientific research fund of Hunan provincial education department. We are grateful to the anonymous reviewers for their valuable comments and suggestions to improve our work.

References

[1] Özdamar Linet., Ekinci Ediz., Küçükyazic Beste., Emergency logistics planning in natural disasters, Annals of operations research, 129(1-4), 217-245. (2004)

[2] Li, J.F., Xing, L.N., Integrative Forest Fire Monitoring System Framework, Disaster Advances, 5(4), 726-729. (2012)

[3] Bai, G.Q., Xing, L.N., Chen, Y.W., Scheduling Multi-platforms Collaborative Disasters Monitoring Based on Coevolution Algorithm, Research Journal of Chemistry and Environment, 16(S2), 43-50. (2012)

[4] Lian, Z.Y., Tan, Y.J., Xu, Y.F., Mission Planning for Space-based Disaster Monitoring, Disaster Advances, 5(4), 1346-1350. (2012)

[5] Chen, C., Tan, Y.J., Xing, L.N., Study on Application of Unmanned Aerial Vehicle for Disaster Monitoring, Research Journal of Chemistry and Environment, 16(S2), 51-55. (2012)

[6] Chuang, M. C, & Ma, Y. C. Expressing the expected product images in product design of microelectronic products. International Journal of Industrial Ergonomics, 27, 233-245. (2001)

[7] Dor'e, R., Pailhes, J., Fischer, X., & Nadeau, J. P. Identification of sensory variables towards the integration of user requirements into preliminary design. International Journal of Industrial Ergonomics, 37,1-11. (2007)

[8] Han, S.H., Kim, K. J., Yun, M. H., Hong, S.W., & Kim, J. Identifying mobile phone design features critical to user satisfaction. Human Factors and Ergonomics in Manufacturing, 14(1), 15-29. (2004)

[9] Zadeh, L. A. Fuzzy logic, neural networks, and soft computing. Communications of the ACM, 37, 77-84. (1994)

[10] Akay, D., & Kurt, M. A neuro-fuzzy based approach to affective design. The International Journal of Advanced Manufacturing Technology, 40(5-6), 425-437. (2009)

[11] Hotto, H., & Hagiwara, M. A fuzzy rule based personal kansei modeling system. In IEEE International Conference on Fuzzy Systems (pp. 1031-1037). Vancouver, Canada. (2006)

[12] Lin, Y. C., Lai, H. H., & Yeh, C. H. Consumer oriented product form design based on fuzzy logic: A case study of mobile phones. International Journal of Industrial Ergonomics, 37, 531-543. (2007)

[13] Nagamachi, M. Kansei engineering and rough sets model. Lecture Notes in Computer Science, 4259, 27-37. (2006)

[14] Nishino, T., Nagamachi, M., & Sakawa, M. Acquisition of kansei decision rules of coffee flavor using rough set method. Kansei Engineering International, 5(4), 41-50. (2006)

[15] Zhai, L. Y., Khoo, L. P., & Zhong, Z. W. A dominance-based rough set approach to Kansei engineering in product development. Expert Systems with Applications, 36, 393-402. (2009)

[16] Chen, C. H., Khoo, L. P., & Yan, W. An investigation into affective design using sorting technique and Kohonen self-organizing map. Advances in Engineering Software, 37, 334-349. (2006)

[17] Hsiao, S. W., & Huang, H. C. A neural network based approach for product form design. Design Studies, 23, 67-84. (2002)

[18] Lai, H. H., Lin, Y. C., Yeh, C. H., & Wei, C. H. User-oriented design for the optimal combination on product design. International Journal of Production Economics, 100(2), 253-267. (2006)

[19] Jiao, J. R., Zhang, Y., & Helander, M. A kansei mining system for affective design. Expert Systems with Applications, 30(4), 658-673. (2006)

[20] Vinodhs,Balajisr., Fuzzy logic based leanness assessment and its decision support system, International Journal of Production Research, 49(13), 4027-4041. (2011)
[21] L. A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning II, Inf. Sci.8 (1975)301-357.
[22] L. A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning III, Inf. Sci.9 (1976)43-80.
[23] C. H. Leon, A. Liu, W. S. Chen, Pattern discovery of fuzzy time series for financial prediction, IEEE Trans Knowl. DataEng. 18(2006)613-625.
[24] H. K. Yu, Are fined fuzzy time-series model for forecasting, Physica A 346(3/4) (2005)657-681.
[25] G. Zhang, B. E. Patuwo, M. Y. Hu, Forecasting with artificial neural networks: the state of the art, Int. J. Forecast.14 (1998)35-62.
[26] P. Chang, C. Liu, Y. Wang, A hybrid model by clustering and evolving fuzzy rules for sales decision supports in printed circuit board industry, Decis. Support Syst. 42(2006) 1254-1269.
[27] Y. Lin, W. G. Cobourn, Fuzzy system models combined with nonlinear regression for daily ground-level ozone predictions, Atm. Environ. 41(2007) 3502-3513.
[28] P. F. Pai, Hybrid ellipsoidal fuzzy systems in forecasting regional electricity loads, Energ. Convers. Manage. 47(15/16)(2006) 2283-2289.
[29] K. Huarng, T. H. K. Yu, The application of neural networks to forecast fuzzy time series, Physica A 336 (2006) 481-491.
[30] Pawalk, Z., Rough Sets, International Journal of Computer and Information Science, 11(5), 341-356. (1982)
[31] Bezdek, J. C., Pattern recognition with fuzzy objective function algorithms, New York, Plenum Press. (1981)
[32] Tomohiro, Takagi, Michio Sugeno Fuzzy Identification of System and Its Application to Modeling and Control. (1985)
[33] Wang, X. Q., Yu, G., Study of construction project bidding based on the BP neural network improved by GA, China Civil Engineering Journal, 40(7),93-98. (2007)
[34] Sanchez, S. M., NOLH designs spreadsheet. Available online via http://harvest.nps.edu/ [accessed 06/01/2014]. (2011)

Zhifei Li received a M.Sc. degree at The Second Artillery Command College, Wuhan, P. R. China, in the College of Information Systems and Management at NUDT. His main areas of research include complex systems analysis, agent-based modeling, and simulation.

Yifan Zhu received a B.S. in mechanics from Peking University, Beijing, China, an M.E. in mechanics and a Ph.D. in systems engineering from the National University of Defense Technology, Changsha, China. His research area covers complex systems analysis, complex-network-based and agent-based modeling, and simulation.

Feng Yang received a Ph.D. in systems simulation from the National University of Defense Technology, Changsha, China, in 2002. He is currently a Professor with the College of Information Systems and Management, National University of Defense Technology. His research areas include system simulation, multi-agent modeling and complex systems analysis.

Wenguang Wang received the Ph.D. degree in system simulation from the National University of Defense Technology, Changsha, China, in 2011. His research interests include system simulation, SOA, and complex systems analysis.