ADAPTIVE DYNAMIC CLONE SELECTION NEURAL NETWORK ALGORITHM FOR MOTOR FAULT DIAGNOSIS

Wu Hongbing[1, 2] Lou Peihuang[1] Tang Dunbing[1]

1. College of Mechanical and Electrical Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, 210016, China

2. Huaian College of Information Technology, Huaian, Jiangsu, 223003, China

E-mail:whb3967957@163.com

Submitted: Jan.14，2013 ***Accepted: Mar.16，2013*** ***Published: Apr.10，2013***

Abstract- A fault diagnosis method based on adaptive dynamic clone selection neural network (ADCSNN) is proposed in this paper. In this method the weights of neural network is encoded as the antibody, and the network error is considered as the antigen. The algorithm is then applied to fault detection of motor equipment. The experiments results show that the fault diagnosis method based on ADCS neural network has the capability in escaping local minimum and improving the algorithm speed, this gives better performance.

Index terms*:* Adaptive, neural network, fault diagnosis, motor, clone selection algorithm

I. INTRODUCTION

With power system moving in the direction of super-high voltage, high capacity and automation, the safety operation of electrical equipments plays an increasingly important role in the safety and stability of power system. It has been a common concern for electrical engineers to monitor apparatus abnormity and foresee equipment faults according to information of equipment's situations provided by advanced state monitoring and fault diagnosing techniques. Repair schedule can be made in accordance with equipment situations and maintenance can be done before any potential faults should appear so as to avoid big economical losses. The possibility for motor fault is unavoidable. Many components of the induction motor are susceptible to failures, the stator windings are subject to insulation breakdown, the bars and end rings of the squirrel cage are subject to failures, machine bearings are subject to excessive wear and damage etc, all caused by a combination of thermal, electrical, mechanical, magnetic and environmental stresses [1,2].

Artificial neural network is a huge system which mimics human brain [3, 4]. Since Hopfield published his paper about neural networks with self-feedback connections [5], and Rumelhart et al. published their monograph on parallel distributed processing (PDP) [6], many neural networks have been developed for various applications. Neural networks have been successfully applied to fault detection [7], automatic control [8], combinatorial optimization [9], information prediction [10], and other fields [11]. Among a number of network models, the BP neural network is a basic model and widely applied in the fault diagnosis area.

Neural networks have provided new fault detection methods, especially for complex systems for which scientific models are difficult to develop [12]. The simple structure and strong nonlinear mapping ability has led to wide use of back-propagation neural networks in fault detection. But it has two disadvantages when used to resolve some complicated problems, which are the slow convergence speed and getting into local minimum easily. Artificial neural network method has some "black box" characteristics in study and problem solution. The knowledge-obtaining process under this approach is difficult to interpret. So, new approaches, particularly those based on biological artificial intelligence, become urgent to be put on the agenda.

Our natural immune system protects our body from foreign cells called antigens by recognizing and eliminating them. This process is called an immune response. Based on the development of

biological immune mechanism, through learning from natural defense mechanism, the artificial immune system provides self-studied, self-organized and clearly self-expressed knowledge. It has the advantage of distributed parallel processing, robustness and can solve complicated problems. It has presented itself a new approach towards fault diagnosis, which can be used to solve problems impossible to be solved by other methods. An immune response contains metaphors like pattern recognition, memory, and novelty detection. The fundamental of AIS is inspired by the theoretical immune system and the observed immune functions, principles, and models.

The artificial immune algorithm is a global optimization algorithm which has the natures of diversity, learning, memory and parallel search [13, 14 and 15].The objectives and constraints of IA are first expressed as antigen inputs. The IA antigen and antibody are equivalent to the objective and the feasible solution of a conventional optimization method [16]. The genetic operators including crossover and mutation are the sequencing processes for the production of antibodies in a feasible space. The algorithm operating on the feasible cells can quickly achieve convergence during the search process. Information entropy is also introduced as a measurement of the diversity of the population to avoid falling into a local optimal solution [17]. Based on an analysis of the mechanism of antigen-antibody recognition as well as patterns of the mechanism, and combined with matrix singular value decomposition, this paper studies the fault diagnosing method based on the minimum antigen-antibody binding energy so as to make a quick diagnosis for equipment faults.

This paper first describes an adaptive dynamic clone selection neural network (ADCSNN). The ADCSNN algorithm proposed in this paper optimizes the weights of the network globally using immune algorithm and searches the weights of network locally using BP algorithm. So it can escape getting into local minimum and improve the convergence speed of the algorithm. Finally, the optimized network is applied to fault detection of motor equipment with better performance [18, 19].

II. ADAPTIVE DYNAMIC CLONE SELECTION ALGORITHMS

The biology has a kind of immune system called the adaptive immune system, which uses two types of lymphocytes: T cells and B cells. Here T cells are ignored. Only B cells which can secrete antibodies are involved. In the immune system, the lymphocytes recognize an invading

antigen and produce antibodies to exclude the foreign antigen. The exact number of foreign molecules the immune system can recognize is unknown but has been estimated to be greater than 1×10^{16}. In spite of the diverse types of antibodies, there is a control mechanism that adjusts to produce the needed quantities in the immune system. The immune system produces the diverse antibodies by recognizing the idiot types between antigens and antibodies or between antibodies and antibodies. These combination intensities can be guessed by the affinity defined using the information entropy theory.

Based on the antibody-antibody affinity, antibody–antigen affinity and their dynamically allotted memory units along with the scale of antibody populations, Adaptive Dynamic Clone Selection Algorithm (ADCS) can adaptively regulate its evolution.

In an optimization problem, the antigen, the antibody, and the affinity between them correspond to the objective function, the solution, and the combination intensity of the solution [20]. Adaptive Dynamic Clone Selection Algorithms [21, 22 and 23] is as follows:

Step 1 Antigens recognition

In the initialization stage, fault samples of the induction motor are taken as a set of antigens (AG) in the shape space. The antibody set (AB) is randomly generated from AG. *Take a training sample* (ag_i): The training sample is defined as the antigen, and in our study the training sample is defined by the feature vector.

The max iteration, termination condition, mutation probability p_m, reconstruction probability p_c, clonal scale N_c, and other parameters are set. Iteration $k=0$, and the initial population are generated as follows:

$$A(0)=\{a_1(0),a_2(0),\cdots,a_n(0)\}\in I^n \tag{1}$$

The input data acts as the antigens for the IA. The objective function and constraints are given as input.

Step 2 Production of initial antibodies $A(0)$

Some antibodies are first selected from the group of memory cells, i.e., some antibodies are chosen from a data base containing superior previous antibodies. In this step, antibodies are created randomly on the feasible space.

Step 3 Calculation of affinities $f(\varphi(A(k)))$

The affinity between the antigen and antibody is then calculated along with the fitness and density of the antibodies. The density is the proportion of those antibodies with the same or similar affinities to all the antibodies. The normalized affinity of antibody $A_i(k)$ in population $A(k)$ is

$$A(k):\{f(\varphi(A(k)))\}=[f(\varphi(A_1(k))),f(\varphi(A_2(k))),\cdots,f(\varphi(A_n(k)))] \tag{2}$$

Step 4 Memory cells updates

The antibodies that have high affinity with the antigen are added to the memory cells. Since the number of memory cells is limited, antibodies in the memory cells are replaced by new ones with higher affinities. Allot adaptively the antibody populations, namely: according to the affinity, the antibody population is disparted to memory unit and generic antibody unit,

$$A(k)=\{M(k),A_b(k)\} \tag{3}$$

Where $M(k)=\{A_1(k),A_2(k),\cdots,A_t(k)\}=(a_1,a_2,\cdots,a_{Na})$, $A_b(k)=\{A_{t+1}(k),A_{t+2}(k),\cdots,A_n(k)\}$, $t=fix[n\times(s_c+D_{is})]$. $fix(*)$ is the integral function below, $fix(x)$ denotes the most integer less than x; s_c is a constant set to assure the size of memory units.

Otherwise:

$$D_{is}=\frac{\sqrt{\frac{1}{(n-1)\times n}\sum_{j=1}^{n}\sum_{i=1}^{n}D_{ij}}}{\max_i\{|u_i-d_i|\}} \tag{4}$$

Which is used to measure the diversity of antibody population, $0\le D_{is}\le 1$, the bigger D_{is} is, the better is the diversity.

Antibody clone operator R_C^P is defined as

$$\begin{aligned}A_c^{(1)}&=R_C^P(a_1,a_2,\cdots,a_{N_a})=R_C^P(a_1)+R_C^P(a_2)+\cdots+R_C^P(a_{N_a})\\&=\{a_1^1,a_1^2,\cdots,a_1^{q_1}\}+\{a_2^1,a_2^2,\cdots,a_2^{q_2}\}+\cdots+\{a_{N_a}^1,a_{N_a}^2,\cdots,a_{N_a}^{q_{N_a}}\}\end{aligned} \tag{5}$$

Where $R_C^P(a_i)=\{a_i^1,a_i^2,\cdots,a_i^{q_i}\}, i=1,2,\cdots,N_a, q_i=1,2,\cdots,n_i$

Step 5 regulate the mutate probability**:** according to the following equation, the corresponding mutate probability of each antibody can be calculated.

$$p_m^i(k)=p_m^c+\left[1+\exp\left(l\frac{f(\varphi(A_i(k)))}{\sum_{j=1}^{n}f(\varphi(A_j(k)))}\right)\right]^{-1}\qquad i=1,2,\cdots,n \tag{6}$$

A further amendment is made as follows:

$$p_m^i(k)=\begin{cases} p_m^M & p_m^i(k)>p_m^M \quad i=1,2,\cdots,t \\ p_m^{A_b} & p_m^i(k)<p_m^M \quad i=t+1,\cdots,n \end{cases} \tag{7}$$

Where p_m^M and $p_m^{A_b}$ are mutate threshold value of memory unit and generic antibody unit respectively, generally, $p_m^M << p_m^{A_b} < 1$.

Antibody mutate operator R_A^M is defined as

$$\begin{aligned} A_c^{(2)} &= R_A^M\left(A^{(1)}\right) \\ &= R_A^M\left(\left\{a_1^1,a_1^2,\cdots,a_1^{q_1}\right\}+\left\{a_2^1,a_2^2,\cdots,a_2^{q_2}\right\}+\cdots+\left\{a_{N_a}^1,a_{N_a}^2,\cdots,a_{N_a}^{q_{N_a}}\right\}\right) \\ &= \left\{R_A^M\left(a_1^1\right)+R_A^M\left(a_1^2\right)+\cdots+R_A^M\left(a_1^{q1}\right)\right\}+\cdots+\left\{R_A^M\left(a_{N_a}^1\right)+\cdots+R_A^M\left(a_{N_a}^{q_{N_a}}\right)\right\} \\ &= \left\{a_1^{11},a_1^{12},\cdots,a_1^{1q1}\right\}+\cdots+\left\{a_{N_a}^{11},a_{N_a}^{12},\cdots,a_{N_a}^{1q_{N_a}}\right\} \end{aligned} \tag{8}$$

Where $R_A^M\left(a_i^j\right)=a_i^{1j}, i=1,2,\cdots,N_a,\ j=1,2,\cdots,q_{N_a}$

Step 6: Adjust the clone scale according to the affinity**,** perform the clonal operator and get the new antibody population $A(k+1)$). In immunology, clone means asexual propagation so that a group of identical cells can be descended from a single common ancestor, such as a bacterial colony whose members arise from a single original cell as the result of mitosis [24].Antibody clone operator R_C^P is defined as

$$\begin{aligned} A_c^{(1)} &= R_C^P\left(a_1,a_2,\cdots,a_{N_a}\right)=R_C^P\left(a_1\right)+R_C^P\left(a_2\right)+\cdots+R_C^P\left(a_{N_a}\right) \\ &= \left\{a_1^1,a_1^2,\cdots,a_1^{q_1}\right\}+\left\{a_2^1,a_2^2,\cdots,a_2^{q_2}\right\}+\cdots+\left\{a_{N_a}^1,a_{N_a}^2,\cdots,a_{N_a}^{q_{N_a}}\right\} \end{aligned} \tag{9}$$

Where $R_C^P\left(a_i\right)=\left\{a_i^1,a_i^2,\cdots,a_i^{q_i}\right\}, i=1,2,\cdots,N_a, q_i=1,2,\cdots,n_i$

Step 7: Antibody reselect operator R_C^S is defined as

$$\begin{aligned} Ab' &= A^{(3)}=R_C^S\left(A^{(2)}\right) \\ &= R_C^S\left(\left\{a_1^{11},a_1^{12},\cdots,a_1^{1q1}\right\}+\left\{a_2^{11},a_2^{12},\cdots,a_2^{1q1}\right\}+\cdots+\left\{a_{N_a}^{11},a_{N_a}^{12},\cdots,a_{N_a}^{1q_{N_a}}\right\}\right) \\ &= R_C^S\left(a_1^{11}+a_1^{12}+\cdots+a_1^{1q1}+a_2^{11}+a_2^{12}+\cdots+a_2^{1q1}+\cdots+a_{N_a}^{11}+a_{N_a}^{12}+\cdots+a_{N_a}^{1q_{N_a}}\right) \\ &= R_C^S\left(a_1^{11}+a_1^{12}+\cdots+a_1^{1q1}\right)+R_C^S\left(a_2^{11}+a_2^{12}+\cdots+a_2^{1q1}\right)+\cdots \\ &+ R_C^S\left(a_{N_a}^{11}+a_{N_a}^{12}+\cdots+a_{N_a}^{1q_{N_a}}\right) \\ &= a_1'+a_2'+\cdots+a_{N_a}' \end{aligned} \tag{10}$$

$k = k + 1$; if satisfy the halt condition, end, or else return to step2.

Combining the enactment iterative times with hunting condition, here the algorithm is halted at the following criterion:

$$\left| f^{*} - f^{best} \right| < \varepsilon \tag{11}$$

Where f^{*} is the global optimum, f^{best} is the current best function value.

The optimization routine iterates from Step 3 until the termination criterion is satisfied.

III. ADCS NEURAL NETWORK DIAGNOSIS METHOD

By analogy we can show that the immune system and the fault identification problem share some common features. In fact, we can state that the system (the body) is exposed to a set of faults (the antigen). Moreover, in diagnosable systems each fault set can be uniquely identified by one of its consistent syndromes (the antigen's molecular structure). In addition, the production of high "affinity" antibodies, pre-selected for the specific antigen, can be considered as fault identification. It follows that AIS can be used as a basis for identifying the set of faulty processors. In addition, our AIS-based diagnosis algorithm should be based on an affinity function that measures the resemblance between the input syndrome and the one generated for a given potential solution.

The main idea of the ADCS neural network diagnosis method is to optimize the weights of the neural network globally using immune algorithm firstly utilizing its characteristics of preservation of diversity and global convergence, after global search to optimize the weights locally using BP algorithm with momentum to get a network which reaches a satisfactory level of performance, finally, input the test data to calculate the output of the network, and diagnosis the fault type based on the output. The principle involves encoding the number of hidden layer neurons, activation functions, and network training method as a binary string. The optimal solution is obtained by the ADCSA as the mean square error (MSE) of the antibody fitness.

a. Fitness probability calculation

The MSE of each antibody is computed in each ADCSA generation. Let MSE of the $i-th$ antibody be MSE_i. The fitness of the antibody is then defined as its reciprocal:

$$f_i = 1/MSE_i \tag{12}$$

Thus, the fitness probability of antibody i is defined as the ratio of single antibody fitness to the sum of the fitness of all antibodies,

$$p_{fi} = f_i / \sum_{j=1}^{S} f_j , \qquad i = 1,2,\cdots,S \tag{13}$$

Where S is the number of antibodies.

b. Density probability calculation

The number of antibodies with the highest density is counted in the current population. The density probability of these antibodies is defined as:

$$p_d = \begin{cases} \dfrac{1}{S}\left(1-\dfrac{t}{S}\right), & \text{t antibodies with highest density} \\ \dfrac{1}{S}\left(1+\dfrac{t^2}{S^2-St}\right), & \textit{other S - t antibodies} \end{cases} \tag{14}$$

The sum of the density probabilities of all antibodies is equal to 1. The density probability of antibodies with the high density is smaller than that of antibodies with lower densities.

c. Antibody promotion and suppression

The antibody selection probability is composed of its fitness probability and its density probability,

$$p_i = \alpha p_{fi} + (1-\alpha) p_d \tag{15}$$

Where $0<\alpha<1$. From Eq. (4) larger antibody fitness will result in a larger selection probability while a higher antibody density will result in a smaller selection probability. Thus, high fitness antibodies are saved, while the antibody diversity is also guaranteed by the promotion and suppression between antibodies based on their densities.

d. Crossover and mutation

During the cross operation, the individual is divided into several groups, and each part is composed of a group of two individuals to complete the single point crossover and multipoint crossover operation with the cross probability.

Mutation can help to increase the diversity of the population; we need to implement single point mutation and multipoint mutation with the rate of the variation. Mutation probability is an important factor to increase population diversity. In the genetic algorithm based on binary encoding, a lower mutation rate is enough to prevent the entire group of genes in any position to change. However, the rate is too small to produce new individuals, and too large probability

makes the genetic algorithm a random search algorithm.

The crossover probability was set to 0.9 while the mutation probability was set to 0.01. The best antibody was directly selected into the next generation with no crossover or mutation operations to guarantee the ADCSA convergence [25].The specific steps for the ADCSA optimization of the BPNN were as follows:

(1) Set the initial parameters, including the population size, S, the crossover probability, P_c, the mutation probability, P_m, the maximum number of network training epochs, Gen, and the minimum MSE goal.

(2) Randomly create S antibodies.

(3)Decode the antibody to get the BPNN parameters and structure, train the network, and compute the MSE of each individual. Let $MSE_{\min} = \min_i(MSE_i),\ i = 1,2,\cdots,S$. Then, calculate the antibody's fitness probability, density probability, and selection probability using Eqs. (3)-(5).

(4)Update memory cells.

(5) Select S antibodies according to their selection probability.

(6)Generate new antibodies using crossovers between antibodies and mutations. The best antibody is directly selected into the next generation with no crossover and mutation operations.

(7) If $MSE_{\min}$ meets the design limit or the number of iterations reaches the predefined limit Gen, then terminate the algorithm, otherwise, go to Step 3.

IV. EXPERIMENT SETUP

The relationship between fault modes and input parameters in motor equipment is a very complicated nonlinear function. Neural networks provide a suitable tool to approximate these nonlinear functions. Neural networks have several advantages. First, they can relate multiple analog variables without making intricate assumptions about the input variables. Second, they do not rely on expert experience but use only the observed data to approximate the implicit nonlinear input/output relationship during the training process. This study uses an adaptive dynamic clone selection algorithm to optimize the structure and parameters of the feed forward neural network to improve the fault detection accuracy.

Induction motors are the most used electro-mechanical devices in industrial applications to convert energy from the electrical form to the mechanical form. They are exposed to many loading and environmental conditions such as corrosive and dusty places. When a motor is continually operated at these conditions, aging of it accelerates and this may lead to many failures. Our method for diagnosis of faults in three-phase induction motors is based on three steps. Figure1 presents the scheme of our approach for induction motor fault detection and diagnosis. We performed experiments on an actual induction motor to test the effectiveness of our algorithm.

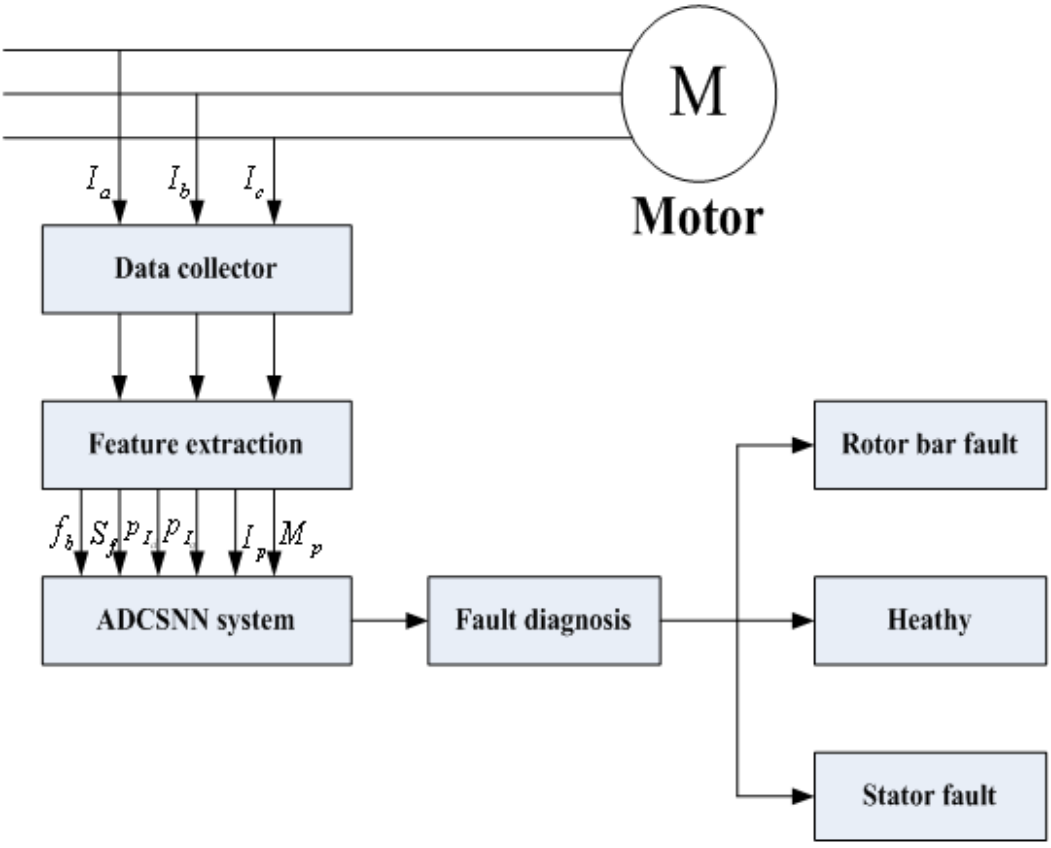

Figure 1. The scheme of fault diagnosis

In Figure1, six features are obtained from three-phase currents of the induction motor, and then these features are given to the adaptive artificial immune system. After the motor condition has been learned by the adaptive immune system, the memory set obtained in the training stage can be used to detect any fault.

The aim of the feature extraction stage is to obtain the fault related features using current signals. The motor condition can be classified by comparing with reference features gathered into classes. Motor current signals can provide sufficient information on stator and broken rotor bars faults [26]. The extracted features are given in the next sub-sections.

(1)Broken rotor bar related feature

The broken rotor bar related feature is expressed as

$$f_b = 2sf_s \tag{16}$$

Where f_b and f_s are the broken rotor bar related sideband frequency and line frequency, respectively. The s represents the slip value.

(2)Stator fault related feature

The feature vector for stator faults are formed as follows:

$$S_f = 1 - \frac{\max(I_{d0})}{\max(I_{q0})} \tag{17}$$

The preceding value converges to one when a fault occurs. In a healthy condition, this value is near to zero.

(3)The mean power of the one phase current (M_p)

$$M_p = \frac{1}{N}\sum_{k=1}^{N}\left\|I_a(k)\right\|^2 \tag{18}$$

(4)The peak-to-peak values of I_d

$$p_{Id} = \left|\max(I_d) - \min(I_d)\right| \tag{19}$$

(5) The peak-to-peak values of I_q

$$p_{Iq} = \left|\max(I_q) - \min(I_q)\right| \tag{20}$$

(6)The last feature is the mean of the complex positive sequence component modulus:

$$I_p = mean\left|\frac{1}{3}\left(I_a + \alpha I_b + \alpha^2 I_c\right)\right|, \qquad \alpha = e^{j(2\pi/3)} \tag{21}$$

We performed experiments on an actual induction motor to test the effectiveness of our algorithm. The characteristics of the three-phase induction motor used in the experiments are listed in Table 1. Three current sensors and a data acquisition card are used to acquire current signals from the induction motor. The motor was tested with a healthy rotor, with a stator open phase fault, and

with a faulty rotor that had one broken rotor bar. Three supply voltages, 300, 340, and 380V, were used for tests of each motor condition.

Table 1. Induction motor characteristic used in the experiment

Description	**Value**
Power	0.5kW
Input voltage	380V
Full load current	1.5A
Supply frequency	50Hz
Number of poles	4
Number of rotor bars	22
Full load speed	1390rpm

The induction motor is operated under three conditions: one for healthy condition, one for the rotor having one broken bar, and one for faulty stator. The number of samples for classification is given in Table 2 for each motor condition.

Table 2. The properties or training and testing data

Class	**Operating mode**	**Supply voltage**	**Number of training samples**
Fault1	Healthy condition	380V	15
		340V	15
		300V	15
Fault2	One broken rotor bar	380V	15
		340V	15
		300V	15
Fault3	Stator fault	380V	15
		340V	15
		300V	15

V. EXPERIMENTAL RESULTS AND DISCUSSION

An induction motor was tested in the laboratory under 3 fault conditions. These tests were performed when the motor was energized from three different power supplies. The motor were supplied on different load in these tests. The diagnosis results based on six fault features are all given. The experimental results can be shown in Tables 3–11. Full load condition and 380V supply voltage under 3 type faults, see table 3.

Table 3. The experimental results (Full load, 380V supply voltage)

Diagnosis results		**Fault features**					
		f_b	S_f	M_p	p_{Id}	p_{Iq}	I_p
	1	50.011	0.011	1.511	0.061	0.031	1.500
	2	50.012	0.010	1.501	0.062	0.030	1.501
Fault1	3	49.512	0.012	1.502	0.059	0.031	1.502
	4	49.871	0.011	1.502	0.063	0.029	1.500
	5	49.111	0.013	1.521	0.061	0.021	1.501
	1	36.616	0.010	1.001	0.061	0.023	0.991
	2	36.291	0.010	0.996	0.063	0.021	0.993
Fault2	3	35.332	0.016	0.989	0.059	0.031	0.987
	4	36.221	0.010	0.991	0.058	0.030	0.997
	5	35.911	0.009	1.010	0.057	0.032	0.989
	1	50.001	0.861	0.712	0.060	0.031	0.801
	2	49.972	0.870	0.723	0.061	0.029	0.802
Fault3	3	50.017	0.862	0.773	0.063	0.032	0.801
	4	49.881	0.866	0.761	0.062	0.031	0.811
	5	50.021	0.831	0.691	0.061	0.030	0.809

Full load and 340V supply voltage condition under 3 type faults, see table 4. Full load and 300V supply voltage condition under 3 type faults, see table 5,

Table4. Experimental fault result of motor with full load and 340V supply voltage

Diagnosis results	**Fault features**					
	f_b	S_f	M_p	p_{Id}	p_{Iq}	I_p
Fault1	50.003	0.010	1.301	0.059	0.030	1.200
Fault2	36.616	0.010	0.801	0.057	0.023	0.791
Fault3	50.001	0.861	0.612	0.057	0.029	0.761

Table5. Experimental fault result of motor with full load and 300V supply voltage

Diagnosis results	**Fault features**					
	f_b	S_f	M_p	p_{Id}	p_{Iq}	I_p
Fault1	50.003	0.010	1.110	0.057	0.029	1.100
Fault2	36.616	0.010	0.711	0.056	0.021	0.711
Fault3	50.001	0.861	0.523	0.056	0.020	0.701

Half full-load and 380V supply voltage condition under 3 type faults, see table 6. Half full-load and 340V supply voltage condition under 3 type faults, see table 7. Half full-load and 300V supply voltage condition under 3 type faults, see table 8.

Table6. Experimental fault result of motor with half full-load and 380V supply voltage

Diagnosis results	**Fault features**					
	f_b	S_f	M_p	p_{Id}	p_{Iq}	I_p
Fault1	50.011	0.012	1.211	0.052	0.030	1.300
Fault2	36.291	0.011	0.886	0.053	0.019	0.981
Fault3	50.017	0.762	0.771	0.059	0.030	0.800

Table7. Experimental fault result of motor with half full-load and 340V supply voltage

Diagnosis results	**Fault features**					
	f_b	S_f	M_p	p_{Id}	p_{Iq}	I_p
Fault1	50.011	0.010	1.111	0.052	0.029	1.100
Fault2	36.291	0.010	0.835	0.053	0.017	0.821
Fault3	50.017	0.731	0.767	0.059	0.028	0.711

Table8. Experimental fault result of motor with half full-load and 300V supply voltage

Diagnosis results	**Fault features**					
	f_b	S_f	M_p	p_{Id}	p_{Iq}	I_p
Fault1	50.011	0.009	1.001	0.048	0.021	0.902
Fault2	36.291	0.008	0.725	0.050	0.015	0.711
Fault3	50.017	0.711	0.667	0.050	0.021	0.667

No load and 380V supply voltage condition under 3 type faults, see table 9. No load and 340V supply voltage condition under 3 type faults, see table 10. No load and 300V supply voltage condition under 3 type faults, see table 11.

Table9. Experimental fault result of motor with no load and 380V supply voltage

Diagnosis results	**Fault features**					
	f_b	S_f	M_p	p_{Id}	p_{Iq}	I_p
Fault1	50.011	0.010	1.011	0.042	0.028	1.001
Fault2	36.291	0.010	0.826	0.043	0.011	0.881
Fault3	50.017	0.732	0.711	0.052	0.023	0.708

Table10. Experimental fault result of motor with no load and 340V supply voltage

Diagnosis results	**Fault features**					
	f_b	S_f	M_p	p_{Id}	p_{Iq}	I_p
Fault1	50.011	0.009	1.001	0.032	0.026	0.890
Fault2	36.291	0.008	0.802	0.043	0.010	0.781
Fault3	50.017	0.632	0.701	0.050	0.021	0.607

Table11. Experimental fault result of motor with no load and 300V supply voltage

Diagnosis results	**Fault features**					
	f_b	S_f	M_p	p_{Id}	p_{Iq}	I_p
Fault1	50.011	0.008	1.000	0.030	0.020	0.790
Fault2	36.291	0.008	0.799	0.033	0.010	0.618
Fault3	50.017	0.612	0.601	0.048	0.020	0.577

From these tables, it can be seen that we can correctly diagnose fault based on the fault features. For the diagnosis of fault1, fault2 and fault3, the methods can all make correct decisions. Experimental performance results are shown in Figure 2. It gives the best, average, and worst convergence speeds of the classifier. The training data are taken as antigens. For these antigens, an antibody population is generated. All training data are normalized before the training process starts. The performance of ADCSNN becomes constant after 30 iterations.

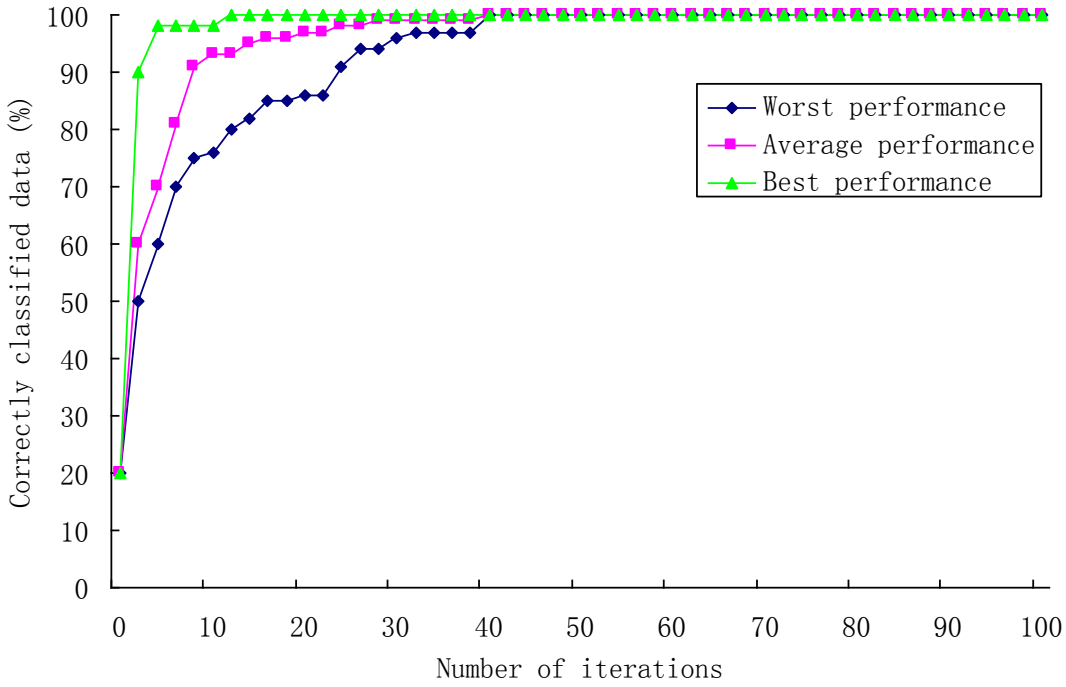

Figure 2. Experimental performance results

Features are used to determine faults under different supply voltages to increase the accuracy of our method. The 3D representation of training data is given in Figure3 for three fault types.

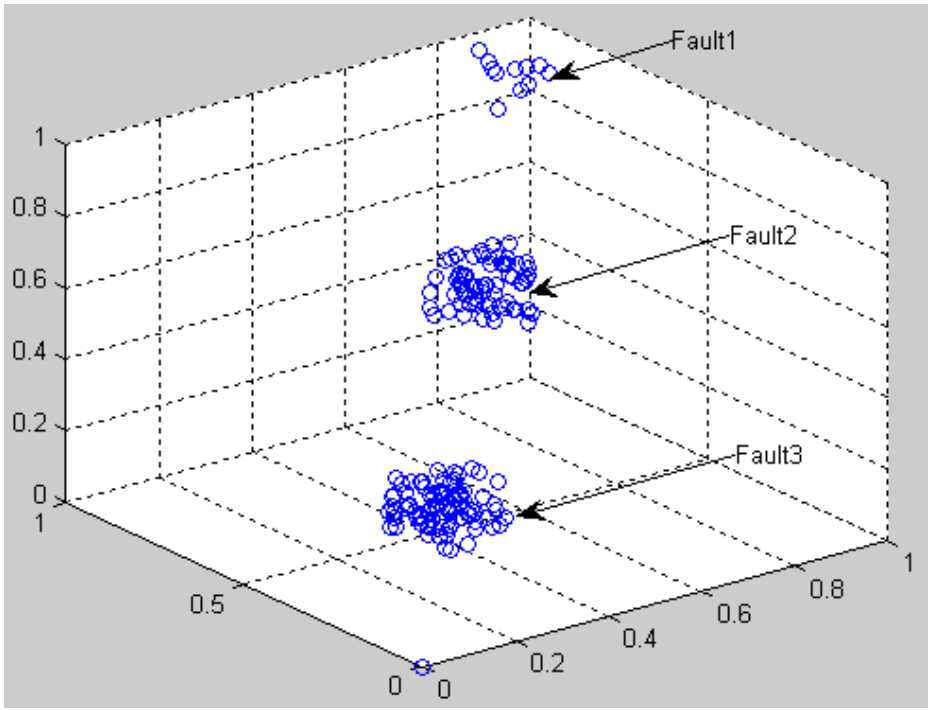

Figure 3. 3D representation of three fault type

A typical evolution of the ADCSNN-based classifier is shown in Figure4, where the classification mean square error is measured over the training set, as a function of the algorithm's number of iterations.

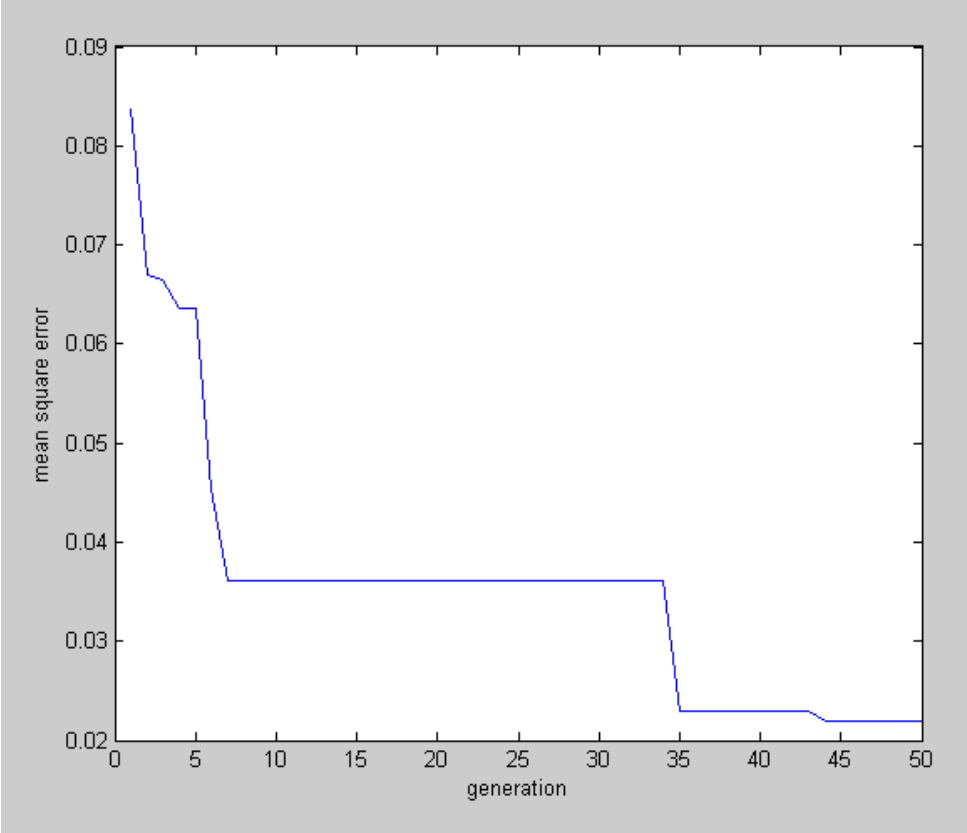

Figure 4. Evolution of classification during a typical ADCSNN execution

Artificial neural network (ANN), K-NN, and fuzzy K-NN methods were also applied to the same fault types, respectively. Artificial neural network (ANN), support vector machines (SVM) [27], K-NN, and fuzzy K-NN methods were also applied to the same fault types, respectively. The structure of the neural network was 6-10-3. The neural network had six inputs and three outputs. The output states of the neural networks were set to the following:

[1; 0; 0] Healthy condition (H)

[0; 1; 0] One broken bar (B1)

[0; 0; 1] Stator fault (S)

The k parameter of fuzzy and standard K-NN was selected as 3. The total accuracy rate for our method and other classifiers are given in Table 12.

Table 12. Experimental results of method with other artificial intelligence techniques

Method	**Accuracy rate (%)**
KNN	86.75
Fuzzy K-NN	90.33
ANN	96.10
SVM	98.14
ADCSNN	98.96

In Table12, The accuracy ADCSNN is approximate to 98.96, ADCSNN is better than K-NN, Fuzzy K-NN, ANN and SVM but difference is much more significant for K-NN and Fuzzy K-NN methods. A typical evolution of the ADCSNN-based classifier is shown in Figure 5, where the classification accuracy is measured over the training set, as a function of the algorithm's number of iterations.

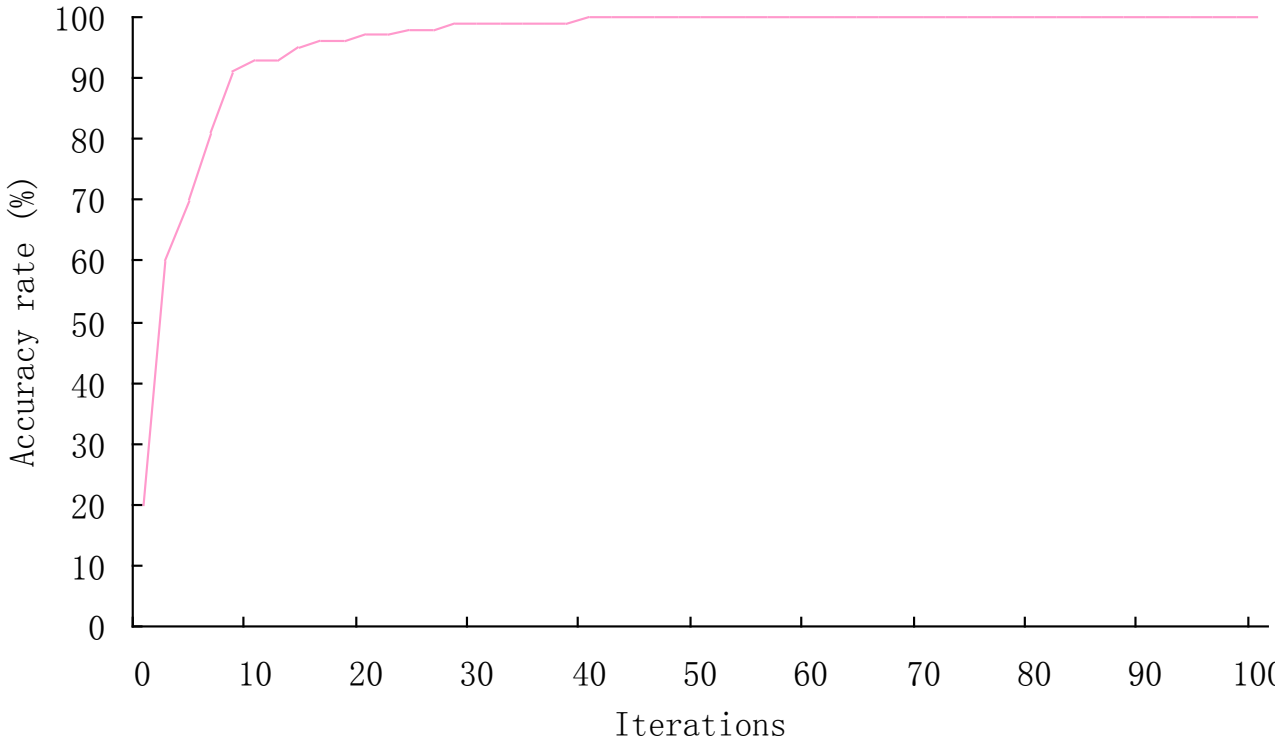

Figure 5. Evolution of classification during a typical ADCSNN execution

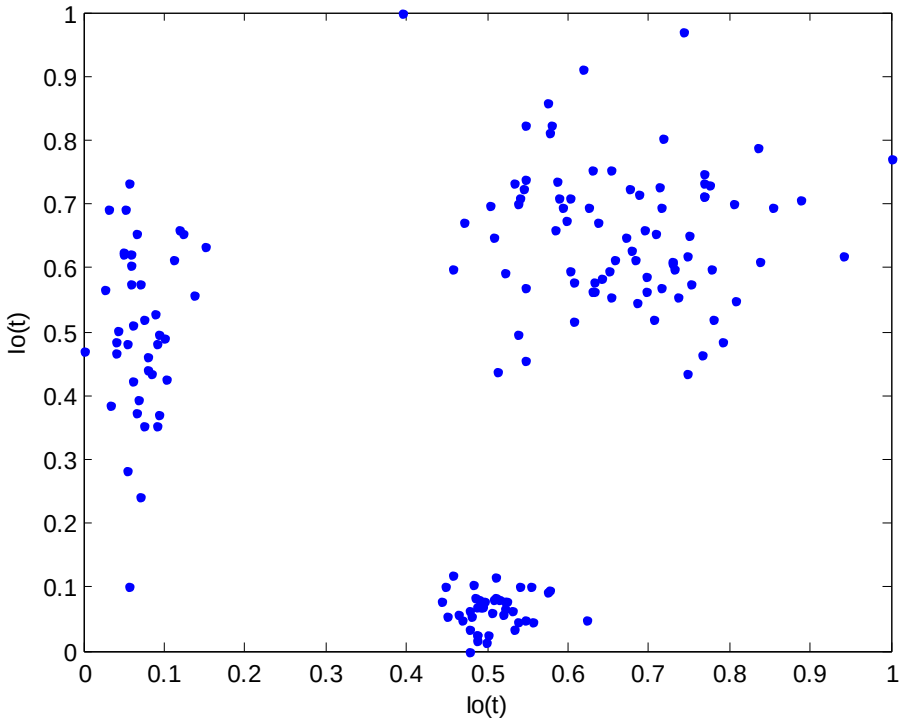

Figure 6. Spaces of each class

The experiments were made on a real induction motor. New features were obtained from three-phase motor to diagnose and classify the faults. The spaces are given in Figure 6. When the results were compared to other intelligent classifiers, ADCSNN produced competitive results for all the problems. The proposed method has possibilities to detect multi faults by ADCSNN. The experimental results show that the proposed ADCSNN has high accuracy rate of fault diagnosis. The average performance of ADCSNN is better than other classifier methods to which it was compared.

VI. CONCLUSIONS

A novel adaptive dynamic clone selection neural networks system, ADCSNN, was designed and implemented. A method for fault diagnosis of motor sets using neural networks and adaptive dynamic clone selection optimization has been proposed. The optimized network has been successfully applied to a real induction motor. New features were obtained from three-phase

motor currents to diagnose and classify the broken rotor bars and stator faults. The experimental results show that the proposed ADCSNN has high classification precision. The average performance of ADCSNN is better than other classifier methods to which it was compared.

ACKNOWLEDGEMENTS

This work was supported by National Natural Science Foundation of China (Grant No. 61105114) and the Key Technology R&D Program of Jiangsu Province (No.BE2010189).

REFERENCES

[1]Nandi S, Toliyat H.A, LI X. Condition monitoring and fault diagnosis of electrical motors–a review [J]. IEEE Trans. on Energy Conversion, 2005, 20(4): 719-729

[2]Diallo D, Benbouzid M.E.H, Makouf A, A fault tolerant control architecture for induction motor drives in automotive applications [J]. IEEE Trans. on Vehicular Technology, 2004, 53(6): 1847-1855

[3] Denker, J.S, Neural Network Models of Learning and Adaptation, Physica 22D, 1986.

[4] Lu, H., Setiono, R, and Liu, H, “Effective Data Mining Using Neural Networks”, IEEE Trans. On Knowledge and Data Engineering, 1996, 8(6), pp. 957-961.

[5] Hopfield J J. Neural networks and physical systems with emergent collective computational abilities. Proc. Natl.Acad. Sci. USA, 1982, 79(8): 2554-2558.

[6] Rumelhart D E, McClell J L. Parallel Distributed Processing, Vol. 1-2. Cambridge, MA, USA: MIT Press, 1986.

[7] Dong Mingchui, Cheng Takson, Chan Sileong. On-line fast motor fault diagnostics based on fuzzy neural networks.Tsinghua Science and Technology, 2009, 14(2): 225-233.

[8] Psillakis H E. Further results on the use of Nussbaum gains in adaptive neural network control. IEEE Transactions on Automatic Control, 2010, 55(12): 2841-2846.

[9] Sun Ming, Zhao Lin, Cao Wei, et al. Novel hysteretic noisy chaotic neural network for broadcast scheduling problems in packet radio networks. IEEE Transactions on Neural Networks, 2010, 21(9): 1422-1433.

[10] Liang Yao, Liang Xu. Improving signal prediction performance of neural networks through multi resolution learning approach. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 2006, 36(2):341-352.

[11] Guo W W, Li M, Li Z, et al. Approximating nonlinear relations between susceptibility and magnetic contents in rocks using neural networks. Tsinghua Science and Technology, 2010, 15(3): 281-287.

[12] Khomfoi S, Tolbert L M. Fault diagnostic system for a multilevel inverter using a neural network. IEEE Transactions on Power Electronics, 2007, 22(3): 1062-1069.

[13] Gui Liang Yin and Li Ping Xiao, "Squirrel-Cage Motors Fault Diagnosis Using Immunology Principles", Proceedings of the Chinese Society for Electrical Engineering, Beijing, 2003, 6, pp. 132-136.

[14]Dasgupta, D., "An Overview of Artificial Immune System and Their Applications", Artificial Immune System and Their Applications, Spring-Verlag, 1998b, pp. 3-21.

[15]Mehdi Neshat, Ali Adeli. A Review of artificial fish swarm optimization methods and applications. INTERNATIONAL JOURNAL ON SMART SENSING AND INTELLIGENT SYSTEMS, VOL. 5, NO. 1, MARCH 2012

[16] Farmer J D, Packard N H, Perelson A A. The immune system adaptation and machine learning. Physica, 1986, 22D:187-204.

[17] Castro P A D, Von Zuben F J. Learning ensembles of neural networks by means of a Bayesian artificial immune system. IEEE Transactions on Neural Networks, 2011, 22(2): 304-316.

[18] Yuan H C, Xiong F L, Huai X Y. A method for estimating the number of hidden neurons in feed-forward neural networks based on information entropy. Computers and Electronics in Agriculture, 2003, 40(1-3): 57-64.

[19] A. F. Salami, H. Bello-Salau. A novel biased energy distribution (BED) technique for cluster-based routing in wireless sensor networks [J]. International journal on smart sensing and intelligent systems vol. 4, NO. 2, June 2011

[20]Chun J S, Lim J P, Jung H K. Optimal design of synchronous motor with parameter correction using immune algorithm. IEEE Trans. on Energy Conversion, 1999, 14(3):610-615.

[21] De Castro L N, von Zuben F J. The clonal selection algorithm with engineering applications. In: Whitley L D, Goldberg D E, et al, eds. Proc. of the GECCO 2000. San Fransisco: Morgan Kaufman Publishers, 2000. 36-37

[22] GONG Mao-Guo, HAO Lin. Data reduction based on artificial immune system [J]. Journal of Software, Vol.20, No.4, April 2009, pp.804-814 (in Chinese)

[23]Surya S,Mack G W,Powers E J et al. Characterization of distribution power quality events with flourier and wavelet transforms [J].IEEE Transactions on Power Delivery,2000,15 (1):247-254

[24] M. G. Gong, H. F. Du, and L. C. Jiao, "Optimal approximation of linear systems by artificial immune response," Sci. China Series F: Inf. Sci., vol. 49, no. 1, pp. 63–79, Jan. 2006.

[25]Jiao Licheng, Li Yangyang, Gong Maoguo, et al. Quantum-inspired immune clonal algorithm for global optimization. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 2008, 38(5): 1234-1253.

[26] Thomson, W. T., & Fenger, M. Current signature analysis to detect induction motor faults. IEEE Industrial Applications Magazine, 2001, 7, 26-34.

[27] Aydin, I., et al. A multi-objective artificial immune algorithm for parameter optimization in support vector machine. Applied Soft Computing Journal. doi:10.1016/j.asoc.2009.11.003.