Cooperative Perception Optimization Based on Self-Checking Machine Learning

Haoxiang Sun[1,*], Changxing Chen[1], Yunfei Ling[1] and Mu Yang[1]

Abstract: In the process of spectrum perception, in order to realize accurate perception of the channel state, the method of multi-node cooperative perception can usually be used. However, the first problem to be considered is how to complete information fusion and obtain more accurate and reliable judgment results based on multi-node perception results. The ideas put forward in this paper are as follows: firstly, the perceived results of each node are obtained on the premise of limiting detection probability and false alarm probability. Then, on the one hand, the weighted fusion criterion of decision-making weight optimization of each node is realized based on a genetic algorithm, and the useless nodes also can be screened out to reduce energy loss; on the other hand, through the linear fitting ability of RBF neural network, the self-inspection of the perceptive nodes can be realized to ensure the normal operation of the perceptive work of each node. What's more, the real-time training data can be obtained by spectral segmentation technology to ensure the real-time accuracy of the optimization results. Finally, the simulation results show that this method can effectively improve the accuracy and stability of channel perception results, optimize the structure of the cooperative network and reduce energy consumption.

Keywords: Spectrum sensing, cooperative sensing, genetic algorithm, neural network, fusion criteria, self-checking.

1 Introduction

Spectrum perception refers to the perception of whether there is a master user signal in the channel of different frequency bands within a certain spatial range, so as to find out whether there is an unutilized or under-utilized radio "spectrum hole" in the current time and space. Spectrum sensing technology is not only an important guarantee to protect authorized primary users from being interfered, but also a prerequisite for unauthorized secondary users to make full use of spectrum resources to complete communication.

The traditional spectral sensing methods include energy detection, cyclic stationary feature detection and matched filter detection. In practice, in order to facilitate implementation, one simple application method energy detection is usually adopted, but it is not accurate and stable enough. It is easy to cause interference to primary users, and can only be limited to improve spectrum efficiency, for which most of the primary user

[1] Air Force Engineering University, Xi'an, 710051, China.
[*] Corresponding Author: Haoxiang Sun. Email: guardian_shx@sina.com.

can't accept such secondary users to use their authorized spectrum.

Therefore, cooperative spectrum sensing technology is proposed. Cooperative perception can improve the accuracy of perception results by eliminating the influence of path shadow and depth fading based on the fusion of perception results of multiple nodes in space. The difficulty of collaborative perception lies in the selection of information fusion criterion of perception results. In the beginning, “and” criterion, “or” criterion was used. Later, some literature proposed perception criterion based on d-s evidence theory and other methods. However, the fusion results of these methods all have some shortcomings.

As is known to all, the purpose of cognitive radio is through the use of artificial intelligence technology study from the external environment, thus the real-time change of transmission parameters such as power, carrier frequency and modulation technology, to realize the coexistence of primary users and secondary high reliable communication, and the heterogeneous network environment of limited radio spectrum resources efficient use. Therefore, artificial intelligence technology can be applied to every process of spectral perception.

Li et al. [Li and Yang (2010)] formalized their application to cognitive radio and developed a framework from within which they can be useful. Thilina et al. [Thilina, Choi, Saquib et al. (2013)] proposes that support vector machine (SVM) is applied to optimize the classification results of cooperative sensing in spectrum sensing. In Zografski et al. [Zografski, Bogoeva-Gaceva and Petrusevski (2006)], a machine learning algorithm is used to analyze the occupancy state of the primary user to evaluate the probability of the secondary user blocking the future slot, so that the system designer can use it to define the spectrum allocation and spectrum sharing strategy.

The idea proposed in this paper is to record the occupation of the main user in the cognitive channel in real time by spectral segmentation technology under the condition of limiting detection probability and false alarm probability, and combine the perception results of each node as the training data of genetic algorithm.

The overall learning process is to realize the information fusion of the results of each perception node based on genetic algorithm to calculate the individual fitness in the evolutionary process, and obtain the optimal mapping relationship between the perception results of each perception node and channel state, that is, the weight of the perception results of each node in the decision-making. In addition, considering that the perceptive devices of the perceptive nodes may fail, this paper also proposes to use the nonlinear fitting ability of RBF neural network to detect the fault nodes in the learning process, timely troubleshoot the faults and avoid affecting the judgment results.

2 Sensing model

2.1 Single perceptive node model

At each perception node, spectral perception is conducted by energy detection method, so a binary hypothesis test model is established as follows:

$$x(i) = \begin{cases} n(i), & H_0 \\ hs(i) + n(i), & H_1 \end{cases} \tag{1}$$

In the above formula, $x(i)$ represents the sampled signal received at the ith time, $n(i)$

represents the noise signal in space, $s(i)$ represents the main user signal emitted by the transmitter, and h represents the attenuation coefficient of the main user signal in the process of spatial propagation. Suppose that H_0 means that the primary user signal does not exist, that is, the received signal only contains noise signal, H_1 means that the primary user exists, that is, the received signal has both noise signal and primary user signal.

Normally, the corresponding statistic T or decision threshold λ can be constructed based on the signal to noise ratio of the channel. When $\mathrm{T} < \lambda$, receiving hypothesis H_0, the primary user signal does not exist, conversely, when $\mathrm{T} > \lambda$ accepts a hypothesis H_1, the primary user exist. What's more, the threshold value is uncertain when T=lambda, or it needs to be judged in another way. In cooperative perception, the perception results of other nodes can be fused to make judgments.

Therefore, the detection probability and false alarm probability of node i are shown in Eq. (2).

$$\begin{cases} P_{di} = P\{\mathrm{T}_i > \lambda_i | H_1\} \\ P_{fi} = P\{\mathrm{T}_i > \lambda_i | H_0\} \end{cases} \quad n = 1,2,3, \dots \dots, n \tag{2}$$

2.2 Cooperative spectrum perception model

In the application of centralized spectrum sensing technology, the traditional fusion criteria of perception results include "and" criterion and "or" criterion. When the "and" criterion is used, the detection probability P_{d_AND} and false alarm probability P_{f_AND} of the collaborative perception model with n perception nodes are shown in Eq. (3):

$$\begin{cases} P_{d\_AND} = \prod_{i=1}^{n} P_{di} \\ P_{f\_AND} = \prod_{i=1}^{n} P_{fi} \end{cases} \quad n = 1,2,3, \dots \dots, n \tag{3}$$

When the "or" criterion is used, the detection probability P_{d_OR} and false alarm probability P_{f_OR} of the collaborative perception model with n perception nodes are shown in Eq. (3):

$$\begin{cases} P_{d\_OR} = 1 - \prod_{i=1}^{n}(1 - P_{di}) \\ P_{f\_OR} = 1 - \prod_{i=1}^{n}(1 - P_{fi}) \end{cases} \quad n = 1,2,3, \dots \dots, n \tag{4}$$

It can be seen that with the use of the "and" criterion, the false alarm probability of the perception model will decrease with the increase of the number of nodes, so the reliability will be improved, but the detection probability will also decrease, resulting in lower detection accuracy and lower spectrum utilization. At the same time, with the use of "or" criterion, although the detection probability is improved with the increase of the number of nodes, the false alarm probability is also increased, resulting in a decrease in reliability. Therefore, there are some deficiencies in both schemes.

In order to achieve high detection probability and false alarm probability, the weighted fusion criterion is used in this paper. In cooperative perception, based on fusion judgment with different weights, the detection probability P_{d_ω} and false alarm probability P_{f_ω} are shown in Eq. (5):

$$\begin{cases} P_{d\_\omega} = \sum_{i=1}^{n} \omega_i P_{di} \\ P_{f\_\omega} = \sum_{i=1}^{n} \omega_i P_{fi} \end{cases} \quad n = 1,2,3, \dots \dots, n \tag{5}$$

It can be known from formula (5) that $P_{d\_\omega} \in [\min(P_{di}), \max(P_{di})]$, $P_{f\_\omega} \in [\min(P_{fi}), \max(P_{fi})]$. That is, when the detection probability and false alarm probability of each node are not the same, the detection probability and false alarm probability of fusion judgment result are reduced. Therefore, this paper considers the value of limiting the detection probability of each node $P_{di} \geq P_{d0}$, the value of limiting the false alarm probability of each node $P_{fi} \leq P_{f0}$, then the global probability value after fusion can be obtained to ensure the arrival the scope of $P_{d\_\omega} \geq P_{d0}$, $P_{f\_\omega} \leq P_{f0}$.

In Lin [Lin (2010)], it is mentioned that when detection probability and false alarm probability are known, the relationship between the number of samples needed and SNR is shown in Eq. (6).

$$\mathrm{N} = 2[\mathrm{Q}^{-1}(P_{fa}) - \mathrm{Q}^{-1}(P_d)(1+\mathrm{snr})]^2/snr^2 \quad (6)$$

Therefore, given the appropriate sampling frequency f_s, the optimal detection time required for each node to achieve the rated performance under the condition of known SNR can be obtained. In this way, the waste of energy and time can be reduced as much as possible on the premise that the perception results of each node reach the standard detection probability and false alarm probability, so as to ensure the accuracy and reliability of fusion perception results.

3 Self-checking machine learning model

In order to ensure high detection probability and false alarm probability at the same time, this paper proposes a centralized spectrum sensing technology model based on self-checking machine learning, as shown in Fig. 1.

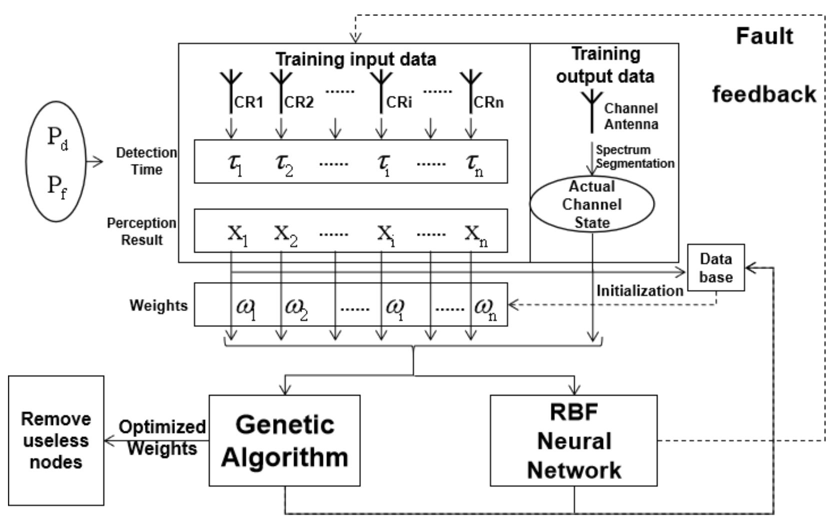

Figure 1: Centralized spectrum sensing model flowchart based on self-checking machine learning

As can be seen from Fig. 1, the perception results obtained by n perception nodes are input data of the machine learning training data set, and the actual channel states obtained by the cognitive channel perception antenna and spectrum segmentation technology are output data of the training data set. These data are used for machine learning training on the one hand, on the other hand, transmitted to the database for recording, and the initialized input weight can be determined by statistics. Machine learning is divided into two parts. One part is to determine the optimal weight of information fusion of multi-node perception results and remove useless nodes based on the genetic algorithm; the other part is to locate fault nodes through fault detection neural network.

3.1 Data acquisition based on spectrum segmentation

Spectrum segmentation technology refers to the frequency band bandwidth is divided into several parts based on different purposes, such as frequency division multiplexing technology, a typical application, the total bandwidth used for transmission channel is divided into a number of sub-bands (or called sub-channels), each sub-channel to transmit one signal. In this paper, we consider applying spectrum segmentation technology to the training data collection of spectrum perception, and divide the channel used by authorized users into two sub-bands, as shown in Fig. 2. The total bandwidth of the channel is W, in which the sub-band with bandwidth of W_s is used to receive channel signals for spectral perception. Based on the energy detection method, the authorized users and cognitive users' use state in the channel can be counted as the training data of machine learning. The sub-band with the remaining the bandwidth of W_c is used for the data transmission of cognitive users. In addition, in order to reduce the impact on the data transmission rate of cognitive users to assume $W_c \gg W_s$.

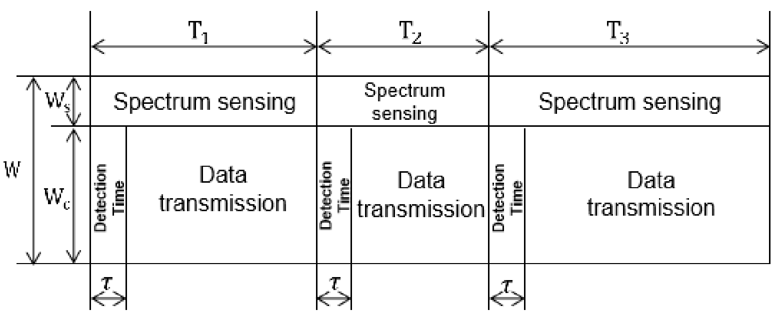

Figure 2: Sketch Map of spectrum segmentation technology

3.2 Weight optimization based on genetic algorithm

The decision-making model based on information fusion of perception results is actually a classifier of perception results. By referring to Han [Han (2005)], we can know that common classification algorithms include decision tree, Bayesian network, neural network and so on. In order to observe the optimization weight more conveniently, this paper uses the weighted naive Bayes classification algorithm based on genetic algorithm mentioned in Bao et al. [Bao, Zhou and Duan (2018)] to realize the weight optimization of channel state judgment results of each perception node.

Set the attribute vector $X = \{x_1, x_2, \cdots, x_m\}$ to represent m perceptive nodes, each node collects n sets of data for training. The actual state of the channel obtained by spectral sensing technology is taken as the category variable C. The data was collected n times and the result C was obtained that $C = \{c_1, c_2, \cdots, c_n\}$, so it can be obtained that the training sample set is $\text{Train} = \{X, C\}$ and the test sample set is $\text{Test} = \{X_j\}$.

The weighted naive Bayesian classification model is shown in Eq. (7).

$$(Test) = argmaxP(C_k)\prod_{i=1}^{m} p(x_i|C_k)^{\omega_i}, 1 \leq k \leq n \tag{7}$$

In the formula, ω_i represents the weight of the perceptive node. The greater the attribute weight is, the more accurate the detection of the actual channel state is, and the greater the influence on the judgment result is.

In order to obtain the optimal weight, genetic algorithm can be used to search for the optimal combination of weights. Genetic algorithm is a heuristic algorithm that solves the optimization of search by simulating the natural selection of Darwinian biological evolution and the biological evolution process of genetic mechanism. It can operate directly on structural objects without the limitation of derivative and continuity of functions, and it has inherent implicit parallelism and better global optimization ability. Moreover, the probabilistic optimization method can be automatically acquired and guide the optimization search space, adjust the search direction adaptively without a definite rule.

Therefore, the corresponding genetic algorithm model is established as shown in Fig. 3.

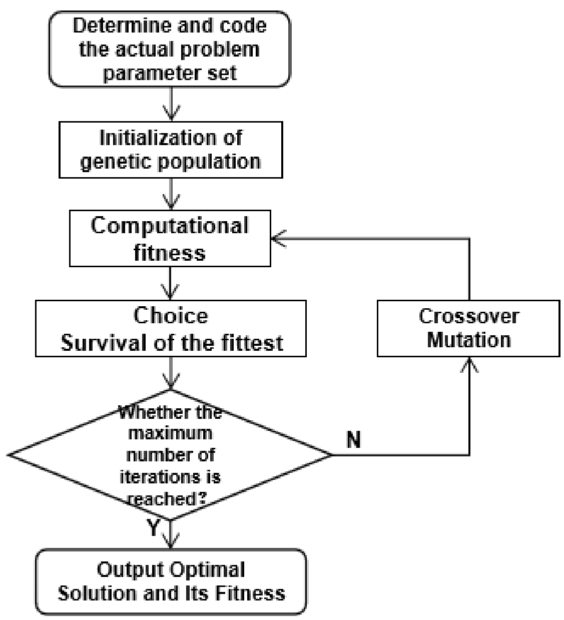

Figure 3: Genetic algorithm structure diagram

In order to establish the model, it is assumed that there are n perceptive nodes in the system, so the number of attributes is n, and the length of each chromosome is n. The number of data collected by each node is m, so the number of individuals in the population of the genetic algorithm is set as m, and set the maximum number of iterations to d_{max}.

The genetic algorithm mainly includes three basic genetic operators: selection, crossover and mutation. Selection is survival of the fittest, is to select and retain chromosomes that are more in line with the requirements by setting the fitness calculation method. In this paper, the detection probability of perceptual results is used as the fitness of the selection operator. Crossover is the genetic process of replacing and recombining part of the attribute structure of two parent chromosomes to generate new chromosomes. The probability of crossover operator p_{jc} needs to be set in the model. Mutation is a process in which the gene value of some attribute loci of some chromosomes in a population changes, and the probability of mutation operator p_{by} needs to be set in the model.

Fig. 3 shows that the parameters of the model of weight optimization genetic algorithm have been determined, so that the genetic population can be initialized, the fitness of chromosomes in each generation can be calculated, and the optimal solution can be finally searched.

According to the obtained weights, nodes with extremely small weights can be screened out. It can be seen that they are useless in the current channel perception, and this node is no longer used for perception, which not only saves energy consumption, but also optimizes the cooperative perception structure and reduces training data.

3.3 Fault detection based on neural network

RBF neural network was selected as the fault detection neural network based on the characteristics of high requirements of spectral perception for the accuracy of results and fast implementation of the training process. Compared with BP neural network used in some references, the RBF neural network has a very high approximation accuracy for the mapping relation between input and output. At the same time, as a feedforward neural network, RBF neural network has the ability of global approximation, which avoids the disadvantages of BP neural network falling into local minimization. With a compact topology, its structural parameters can be separated for learning, and its convergence speed is fast.

To achieve weight optimization by applying RBF neural network, it is necessary to establish a three-layer forward network with a hidden layer, as shown in Fig. 4. The perception result of each perception node is taken as the input vector of the first layer. The second layer is the hidden layer, which is used to complete the transformation of input vectors from the spatial layer to the nonlinear high-dimensional layer. The transformation function of the hidden layer is a non-negative nonlinear function that is radially symmetric to the center point and attenuates. The third layer is the output layer, which is used to find out the mapping relationship with the hidden layer, and to output the linear weighted sum of hidden elements.

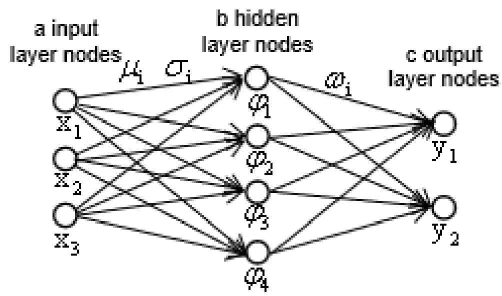

Figure 4: RBF neural network structure diagram

In the process from the input layer to the hidden layer, the k-means algorithm is used to calculate the center vector μ_i, and the KNN (K Nearest Neighbor) algorithm is used to calculate the width vector σ_i. And the connection weight from the hidden layer to the output layer is ω_i.

Therefore, the relationship between input and output can be expressed as:

$$y_i = \sum_{i=1}^{a} \omega_{ij} \phi(||x - \mu_i||), (j = 1,2, \dots, c) \tag{8}$$

In the above formula, $||\mathrm{x}||^2 = \sum_{k=1}^{b} {x_k}^2$, $\phi(\cdot)$ is the activation function, and gaussian is commonly used as the activation function.

The process of fault detection based on RBF neural network is shown in Fig. 5.

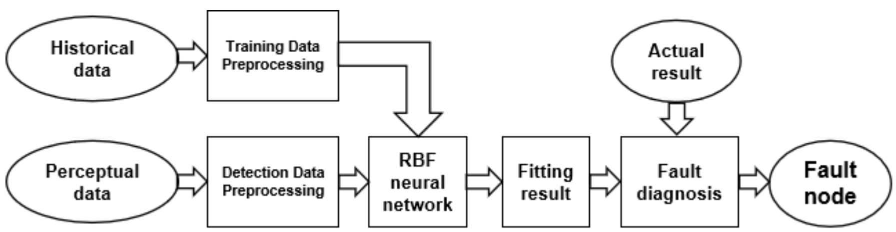

Figure 5: Fault detection flow chart based on RBF neural network

Firstly, the historical perception data of each perception node are preprocessed to obtain training data, and the RBF neural network group is established. The number of perceptive nodes is m, and the number of data collected by each node is n groups. Therefore, data preprocessing is m groups of data to construct m RBF neural networks. The input data of each group of data is shown in Eq. (9):

$$\begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_{j-1} \\ X_{j+1} \\ \vdots \\ X_m \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{(j-1)1} & x_{(j-1)2} & \cdots & x_{(j-1)n} \\ x_{(j+1)1} & x_{(j+1)2} & \cdots & x_{(j+1)n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix}, \mathrm{j} = 1,2, \dots, \mathrm{m} \tag{9}$$

The output data is shown in Eq. (10):

$$Y_j = X_j = [x_{j1}\ x_{j2}\ \cdots x_{jn}], j = 1,2,\dots,m \tag{10}$$

Therefore, each perception result of each perception node is preprocessed into m groups through data preprocessing, and corresponding output y_j can be obtained through the nonlinear fitting of the corresponding RBF neural network respectively.

Fault diagnosis is to compare the y_j output by the neural network with the perception result x_j of the actual perception node j, as shown in Eq. (11):

$$\begin{cases} P(y_j \neq x_j) \leq \varepsilon, & H_0 \\ P(y_j \neq x_j) > \varepsilon, & H_1 \end{cases} \tag{11}$$

In the above formula, ε is the threshold to determine whether a fault occurs. In order to avoid contingency, the output value of the neural network is compared with the perception result many times. The fault probability with inconsistent statistical results is compared with the threshold value. When the fault probability of a node is greater than the threshold value, the node is judged to have a fault.

4 Experimental simulation and analysis

By referring to IEEE802.22 standard and other references, it is learned that spectrum sensing in an ideal state requires detection probability of more than 99% and false alarm probability of less than 10%. Therefore, in the simulation, the detection probability value of each perception node is set as $P_{di} \geq 99\%$, and the false alarm probability value is set as $P_{fi} \leq 10\%$. The change relationship between the number of single-perception samples of each node and the SNR is shown in Fig. 6 according to Eq. (6).

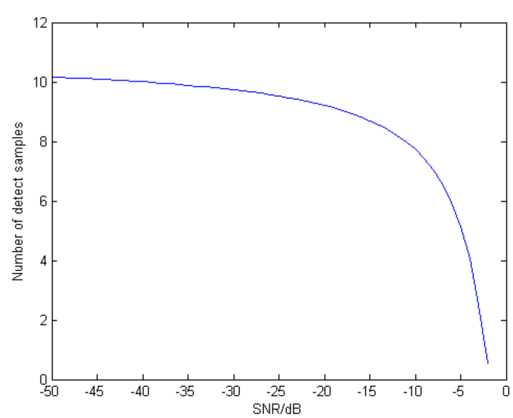

Figure 6: Curve of the relationship between the number of single perception samples and SNR

Therefore, the detection time of each node was obtained, and the number of perceptive nodes was set as 10, so the corresponding perception results could be obtained. 1000 times of data were collected for the training of weight optimization by genetic algorithm.

In order to verify the theory given above, the actual environment is simulated in the laboratory, with larger obstacles representing buildings, and smaller obstacles representing trees, the transceiver antenna and obstacles are arranged as shown in Fig. 7.

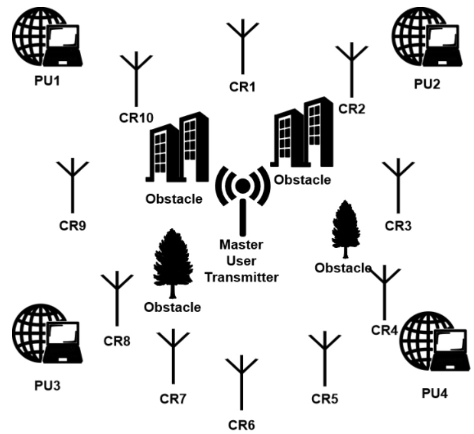

Figure 7: Structure of experimental simulation environment

In the simulation experiment, there are 10 sensing nodes in the system, so the number of attributes is 10 and the length of each chromosome established is 10. The number of data collected by each node is 1000, so the number of individuals in the population of the genetic algorithm is set as 1000. The maximum number of iterations was set as 500, the probability of chromosome crossing was 0.2, and the probability of mutation was 0.2. The accuracy of fusion results was taken as the fitness of the genetic algorithm. Finally, the relationship between fitness and iteration times in the process of genetic algorithm searching for the optimal solution can be obtained by importing data simulation, as shown in Fig. 8.

It can be seen from the Fig. 8 that the optimal solution is close to the optimal solution after about 80 iterations. Therefore, in order to save energy consumption and reduce operation time, the maximum number of genetic iterations can be set to 100 when updating the model with real-time data in the later stage to meet the requirements.

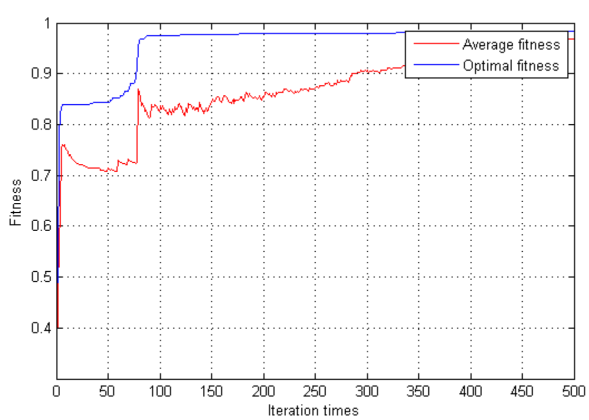

Figure 8: Graph of the relationship between genetic fitness and the number of iterations

The optimized weight obtained from the training data through genetic algorithm is shown in Tab. 1, and the optimal fitness corresponding to the following weight is 0.984.

Table 1: Optimal weights corresponding to each sensing node

Sensing Node Number	1	2	3	4	5
Optimal weight	0.086496	0.0006773	0.16392	0.097439	0.15046
Sensing Node Number	6	7	8	9	10
Optimal weight	0.16541	0.15943	0.01937	0.15045	0.0061805

As can be seen from Tab. 1, the weights of nodes 2 and 10 are extremely small, which is consistent with the situation that nodes 2 and 10 in Fig. 7 are affected by large obstacles. So they can be regarded as useless nodes. Remove the data of useless nodes, set the maximum iteration number to 100, and use genetic algorithm to train again to get the relationship between fitness and iteration number, as shown in Fig. 9.

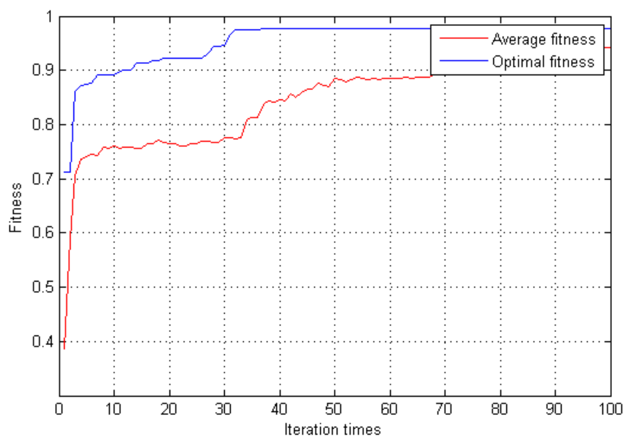

Figure 9: Graph of the relationship between genetic fitness and iteration times after removing useless nodes

The optimized weight obtained by training the data with the useless nodes removed is shown in Tab. 2, and the optimal fitness corresponding to the following weight is 0.977.

Table 2: Optimal Weights of Sensing Nodes after Removing Useless Nodes

Sensing Node Number	1	3	4	5
Optimal weight	0.091103	0.14578	0.12842	0.12907
Sensing Node Number	6	7	8	9
Optimal weight	0.1778	0.16169	0.013502	0.14792

It can be seen that although the optimal fitness decreases slightly at this time, the training results are close to the optimal solution after 32 iterations, and the solving speed is greatly improved. At the same time, there is no useless node, and the corresponding weight is roughly the same as the original model, so it can be used as the optimal weight for multi-node weighted fusion judgment of collaborative perception.

Similarly, the collected data are transferred to the neural network for training. 900 groups of data are randomly selected as the training data to establish the neural network, and the remaining 100 groups are taken as the test data. The detection results are shown in Fig. 10.

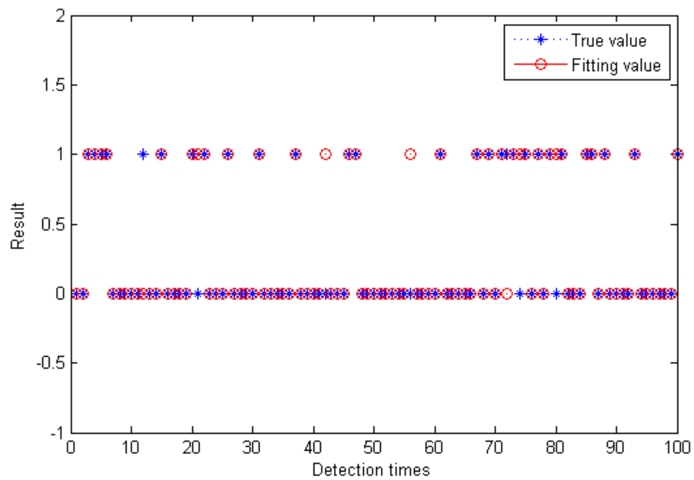

Figure 10: Schematic diagram of test results for 100 times

Another 100 times of data were collected. The statistical results of 100 times of detection were taken as a fault judgment, and the 10 nodes were judged respectively. The judgment results are shown in Fig. 11.

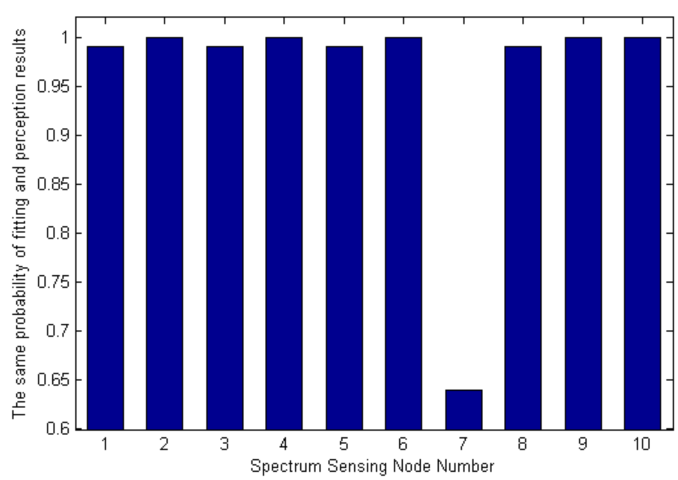

Figure 11: Schematic diagram of fault judgment results of 10 perceptive nodes

It can be seen from Fig. 10 that in the 100 detection attempts under normal circumstances, only 7 times that the fitting result is different from the perceived result. In the fault judgment in Fig. 11, it can be seen that the probability of 10 perception nodes except node 7 that the fitting result is the same as the perceived result is more than 90%. For the perceptive node 7, the detection results are shown in Fig. 12.

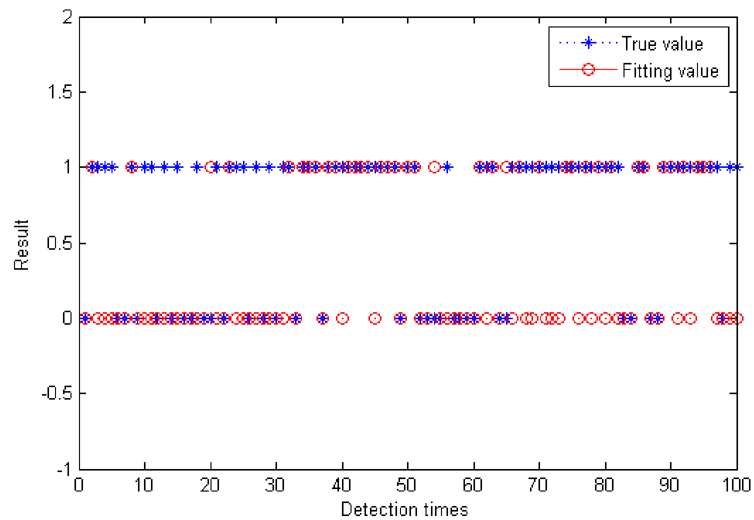

Figure 12: Schematic diagram of detection results of spectrum sensing node 7

It can be seen that the number of times that the fitting results differ from the perception results in this detection reaches 36 times. There is no doubt that this error rate is large, so it can be determined that node 7 is faulty.

5 Conclusions and prospects

In this paper, the innovation points in the first place is different from the previous paper first put forward the method to calculate the performance, but in the limit of detection probability and false alarm probability of all nodes under the premise of the standard for the perception of the result, so as to ensure the probability and false alarm probability of collaborative perception results based on weighted fusion detection can be up to standard, realize the global optimization result. Secondly, after using genetic algorithm to train the historical data to obtain the optimal weights of each node, this paper uses the optimized weights to remove the useless nodes, optimize the cooperative sensing network structure, reduce energy consumption and algorithm complexity. At the same time, in order to ensure higher reliability, this paper also proposed that the training data could be applied to RBF neural network again to realize fault detection of the sensing node, relying on the efficient and reliable nonlinear fitting ability of RBF neural network to timely and accurately detect and locate fault nodes. In addition, this paper also applies spectrum segmentation technology to ensure the real-time accuracy of training data. From the experimental results, it can be seen that the idea proposed in this paper effectively guarantees the global performance of the fusion of cooperative perception nodes to reach more than 97%. In the case of the existence of useless nodes, it can be accurately detected and hardly affects the performance of cooperative perception after removal. And the detection of node faults can also ensure its timeliness and accuracy.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

Azmat, F.; Chen, Y.; Stocks, N. (2016): Analysis of spectrum occupancy using machine learning algorithms. *IEEE Transactions on Vehicular Technology*, vol. 65, no. 9, pp. 6853-6860.

Bao, Y.; Zhou, L.; Duan, P. (2018): A weighted naive bayes classification algorithm based on a genetic algorithm. *Journal of Yunnan University for Nationalities: Natural Science Edition*, vol. 27, no. 6, pp. 525-529.

Barnes, S. D.; Maharaj, B. T.; Alfa, A. S. (2016): Cooperative prediction for cognitive radio networks. *Wireless Personal Communications*, vol. 89, no. 4, pp. 1177-1202.

Chen, Y.; Zhang, H.; Hu, H.; Wang, Q. (2014): An efficient cooperative spectrum sensing algorithm based on BP neural network. *International Conference on Wireless Communication & Sensor Network*, pp. 297-301.

Chu, Y. Z.; Zheng, B. Y.; Ji, W. (2010): Date fusion schemes based on cooperative spectrum sensing. *Journal of Nanjing University of Posts & Telecommunications*, vol. 30, no. 3, pp. 39-45.

Ganesan, G.; Ye, L. (2005): Cooperative spectrum sensing in cognitive radio networks. *IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks*, vol. 57, no. 1, pp. 62-67.

Han, J. (2005): *Data mining: Concepts and Techniques*. Morgan Kaufmann Publishers Inc.

Jaglan, R. R.; Mustafa, R.; Agrawal, S. (2018): Scalable and robust ANN based cooperative spectrum sensing for cognitive radio networks. *Wireless Personal Communications*, vol. 99, no. 3, pp. 1141-1157.

Jian, Y.; Long, L. Y.; Hua, W. Y.; Jie, W. R.; Sen, Y. S. (2012): Cooperative spectrum-sensing algorithm based on weight adaptive optimization. *Computer Engineering*, vol. 6, no. 14, pp. 345-352.

Kaushik, A.; Sharma, S. K.; Chatzinotas, S.; Björn, O.; Jondral, F. (2015): Sensing-throughput tradeoff for cognitive radio systems with unknown received power. *International Conference on Cognitive Radio Oriented Wireless Networks*, pp. 308-320.

Letaief, K. B.; Wei, Z. (2007): Cooperative spectrum sensing. *Cognitive Wireless Communication Networks*, pp. 115-138.

Li, L.; Yang, S. (2010): Snr-based weighted cooperative spectrum sensing in cognitive radio. *Journal of Huazhong Normal University*, vol. 6, no. 6, pp. 2204-2213.

Li, Z.; Wu, W.; Liu, X.; Qi, P. (2018): Improved cooperative spectrum sensing model based on machine learning for cognitive radio networks. *IET Communications*, vol. 12, no. 19, pp. 2485-2492.

Li, Z.; Wu, W.; Liu, X.; Qi, P. (2018): Improved cooperative spectrum sensing model based on machine learning for cognitive radio networks. *IET Communications*, vol. 12, no. 19, pp. 2485-2472.

Liang, X. X.; Cao, L.; Wei, C. G.; Yue, Y. G. (2014): Research on wireless sensor networks data fusion algorithm based on COA-BP. *Applied Mechanics and Materials*, vol. 539, no. 2, pp. 247-250.

Liang, Y. C.; Zeng, Y.; Peh, E. C. Y.; Hoang, A. T. (2008): Sensing-throughput tradeoff for cognitive radio networks. *IEEE Transactions on Wireless Communications*, vol. 7, no. 4, pp. 1326-1337.

Lin, P. (2010): An adaptive energy detection method with less time delay. *Science & Technology Vision*, vol. 25, no. 1, pp. 5-6.

Lu, X.; Wu, Q.; Wu, L. (2014): Research on cooperative spectrum sensing algorithm based on data fusion. *International Conference on Measurement*.

Macaluso, I.; Ahmadi, H.; Dasilva, L. A.; Doyle, L. (2014): Channel usage patterns and their impact on the effectiveness of machine learning for dynamic channel selection. *Cognitive Communication and Cooperative HetNet Coexistence*.

Thilina; K. M.; Choi, K. W.; Saquib, N.; Hossain, E. (2013): Machine learning techniques for cooperative spectrum sensing in cognitive radio networks. *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 11, pp. 2209-2221.

Zhao, X. L.; Zhao, H. S.; Cao, L.; Xu, J. Y. (2013): Weighted cooperative spectrum sensing algorithm based on improved energy detection. *Computer Engineering & Applications*, vol. 49, no. 24, pp. 61-60.

Zhou, X.; Sun, M.; Li, G. Y.; Juang, B. H. F. (2018): Intelligent wireless communications enabled by cognitive radio and machine learning. *China Communications*, vol. 15, no. 12, pp. 16-48.

Zografski, Z.; Bogoeva-Gaceva, G.; Petrusevski, V. (2006): A machine learning method with hybrid neural networks for spectrum analysis. *Proceedings of the 44st Annual Southeast Regional Conference*, pp. 296-299.