Number of Coherent Nodes Estimation Based on Multiple Feature Extraction in Wireless Sensor Network

JIAQI ZHEN, (Member, IEEE), AND XIAOLI ZHANG
College of Electronic Engineering, Heilongjiang University, Harbin 150080, China
Corresponding author: Jiaqi Zhen (zhenjiaqi@hlju.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61501176, in part by the Natural Science Foundation of Heilongjiang Province under Grant F2018025, in part by the Postdoctoral Scientific Research Developmental Fund of Heilongjiang Province under Grant LBH-Q17149, in part by the University Nursing Program for Young Scholars with Creative Talents in Heilongjiang Province under Grant UNPYSCT-2017116, and in part by the Excellent Youth Project of Natural Science Foundation of Heilongjiang Province under Grant YQ2019015.

ABSTRACT Knowing number of nodes is the precondition of their locating and estimation of other parameters in wireless sensor network, actually it can be evaluated according to the signal received by the sensor array. The popular algorithms are only appropriate for the uncorrelated signals and circumstance of Gaussian white noise, the estimation precision will deteriorate at some harsh environments. Hence a new algorithm for determining coherent node number based on multiple feature extraction in wireless sensor network is presented in this article. First, perform spatial difference smoothing to the array data. Then the eigenvalues and eigenvectors can be acquired by eigen-decomposition, consequently multiple features which are used for training are constructed. Finally, back propagation neural network and particle swarm optimization are exploited for calculating node number. Simulation results demonstrate that the algorithms has good performances under the background of colored noise and small samples.

INDEX TERMS Number of signals, wireless sensor network, back propagation neural network, colored noise, coherent signals.

I. INTRODUCTION

With the development of sensor, embedded system, information processing and wireless communication, wireless sensor network (WSN) consisting of many nodes makes it possible to obtain effective information from land, sea, air, sky, and underground. WSN is a kind of net constructing by some sensors which have possibility of perception, computing, communication, and organizing through Ad Hoc, these nodes collaborate with one another [1]–[6], monitor, perceive, and collect every various environmental data in real-time, we can fuse these data to get detail information, then transmit them to the users [7], [8]. Comparing with traditional wireless communication network, WSN shows self-organization, low-cost, and dynamic, so that it can be deployed in the forest [9], [10], desert [11], battle field [12], [13], and other hostile environments, as well as intelligent home [14], [15], factory [16], [17], traffic [18], [19], and hospital [20], [21], their number can be a few to tens of thousands.

The core mission of WSN is detecting information of interest, one of the common characteristics is the demand for the node number [22]–[25], we can respond correctly only number of the nodes are known accurately, it is also a premise of estimating their directions of arrival, positions, powers, and other parameters, actually we can determine this information according to their signals acquiring by some array sensors. In practical applications, there are variety of disturbances, and complex noise, meanwhile, sampling times is always limited by the hardware and harsh environments, then the node number are often calculated inexacty, leading to the deviation of other estimations. Generally speaking, the nodes to be estimated are the cooperation targets of our side, but there are also lots of non-cooperative nodes, some of them are correlated, even coherent with one another, the most typical examples are radars, submarines, satellites, and their decoys set by enemy deliberately, they respectively form a
wireless network and bring great difficulty to our locating and other parameter estimations [26], [27]. Therefore, to study an effective node number estimation algorithm has a great theoretical significance and practical value.

The problem of signal number estimation can date back to the late 1950s, it is necessary to set a detecting threshold to compare with the test statistic at that time, so this algorithm is susceptible to the subjective influence. Since the minimum length description (MDL) [28] and other information theory criteria [29] were put forward, their good estimation performances have attracted wide attentions, but they are only suitable for the Gaussian white noise, concerning this issue, some scholars proposed Gerschgorin Disk estimation [30] which is appropriate for colored noise, then they are improved by spatial smoothing technique, so as to apply to coherent sources. Another algorithm is based on Bootstrap, which is a course of resampling to the received data, in the year 2000, Breich was the first to use Bootstrap and hypothesis testing to compute the signal number [31]. Based on Bootstrap resampling, Zhang exploited clustering to estimate number of sources in circumstances of low signal to noise ratio(SNR) and small samples, as a result of full use of eigenvalues and eigenvectors, the precision is improved greatly compared with previous similar algorithms [32]. But because Bootstrap needs resampling, recombination, and recalculation to the received data for hundreds of times, it is very time consuming.

In 1995, Vapnik proposed the idea of support vector machine(SVM) [33], it is a statistical theory aiming at small snapshots and mainly used for solving data classification. SVM transforms the problem of nonlinear classification into linear classification through increasing the data dimension, it shows good generalization ability, solves the common over-fitting in traditional algorithms, the efficiency and accuracy of the algorithm are relatively high. At present, SVM has been put into text classification, biomedicine, face recognition, and signal number estimation [34], [35], simultaneously more and more researchers solve these problems by neural network [36], [37], but they are merely suitable for uncorrelated sources and Gaussian white noise.

In general, coherent node sources which commonly exist in WSN means signals with the same frequency and fixed phase differences, they will lead to the rank deficit of the received data covariance, resulting in the number misjudgment. Besides, as long as background noise does not meet the white statistical property, it can be considered as colored noise, then the smaller eigenvalues are no longer approximately equal in this environment. So this article presents a new algorithm for determining coherent node number, and there are three contributions: First, differential smoothing is used for decorrelation to get the covariance matrix without the information of uncorrelated sources and noise. Second, multiple features are formed based on the eigenvalues and eigenvectors. Third, back propagation neural network(BPNN) is employed to evaluate signal number, in the processing of parameter initialization, particle swarm optimization(PSO) is
also utilized to solve the problem of unstable training or poor convergence.

The organization of the paper is described below: In section II, we model the signal composed by uncorrelated, coherent sources, and colored noise. Section III gives the process of node number estimation based on BPNN. Section IV demonstrates the algorithm performance by some simulations. Section V summarizes the full paper.

II. SIGNAL MODEL

As shown in Figure 3, a uniform linear array (ULA) formed by M sensors receives K far-field mixed narrow-band sources, where $K_u$ independent sources are from $\theta_u (u = 1, 2, \ldots, K_u)$, $K_c$ coherent ones are from $\theta_c (c = 1, 2, \ldots, K_c)$, and $K_u + K_c = K$, then array output at time $t$ is written

$$X(t) = \sum_{u=1}^{K_u} a(\theta_u) s_u(t) + \sum_{c=1}^{K_c} a(\theta_c) \rho_c s_c(t) + N(t)$$

$$= A\Sigma(t) + N(t).$$

(1)

the covariance matrix of $X(t)$ is

$$E[X(t)X^H(t)] = A\Sigma A^H + A\Sigma K + R_n$$

$$= R_N + R_{NT} + R_n.$$  

(2)

where $R_N$ is the covariance matrix of uncorrelated sources and performs Toeplitz characteristic, that is to say, for a Toeplitz matrix $E$, it satisfies $E = JE^T J$, here $J$ is a permutation matrix whose elements on back-diagonal equal one, and others are all zeros. $R_{NT}$ and $R_n$ are respectively the covariance matrix corresponding coherent and noise. As in Figure 3, we can acquire $M$ similar arrays by virtual sliding, each of them includes $M$ sensors, then refer to the idea of spatial smoothing, we can obtain the data after virtual smoothing

$$X_i(t) = [x_i^1(t), \ldots, x_i^M(t), x_1(t), \ldots, x_{M-i+1}(t)].$$

(3)

the covariance of every subarray is

$$R_i = E[X_i(t)X_i^H(t)].$$

(4)

III. DETERMINE NODE SIGNAL NUMBER

A. REMOVE UNCORRELATED SOURCES AND NOISE

we can eliminate the uncorrelated sources and noise through introducing spatial difference matrix

$$R_i^d = R_i - J_M R_i^T J_M.$$

(5)

$R_N$ is Toeplitz matrix, so $R_N = J_M R_N^T J_M$, take (2) into (5), we have

$$R_d = (R_N + R_{NT} + R_n) - J_M (R_N + R_{NT} + R_n)^T J_M$$

$$= R_{NT} - J_M R_{NT} J_M.$$  

(6)

thus, the covariance no longer has the information of uncorrelated sources and noise. We need to dispose $M$ arrays and average them, then covariance containing only coherent sources can be acquired

$$R_{ds} = \frac{1}{M} \sum_{i=1}^{M} R_i^d.$$  

(7)

then eigen-decomposition is performed, we have

$$R_{ds} = U_c \Sigma_c U_c^H = U_{sc} \Sigma_{sc} U_{sc}^H + U_{nc} \Sigma_{nc} U_{nc}^H.$$  

(8)

B. FEATURE EXTRACTION

After decomposing $R_{ds}$, we can extract some features containing signal information. Eigenvalues represent powers of node signal, so they are selected as the first feature parameter expressing as $r_1 = \lambda = [\lambda_1, \lambda_2, \ldots, \lambda_M]$, and corresponding eigenvector is $e = [e_1, e_2, \ldots, e_M]$.

Besides, the ratio of adjacent eigenvalues reflects their variation degree as well, so it is chosen as the second feature, that is

$$r_2 = \frac{\lambda_i}{\lambda_{i+1}}, i = 1, 2, \ldots, M - 1.$$  

(9)

In order to differentiate signal subspace and noise subspace, corresponding information in eigenvector can be utilized likewise, because mean square error of the eigenvector envelope manifests the regularity of difference between signal and noise, it is written

$$r_3 = \xi_i = \sqrt{\frac{\sum_{k=1}^{M} (|e_{ik}| - |\tilde{e}_i|)^2}{M}}, i = 1, 2, \ldots, M.$$  

(10)

where $|\tilde{e}_i| = \frac{|e_{i1}| + |e_{i2}| + \ldots + |e_{iM}|}{M}$, $i = 1, 2, \ldots, M$. Therefore, three kinds of parameters are selected as input features. In fact, we can extract more characteristics by fusing $\lambda$ and $e$.

C. BPNN TECHNIQUE

BPNN plays an irreplaceable role in intelligent algorithm, which greatly improves the height of our analyzing problems, it is also the promotion of neural network algorithm and widely used. BPNN mainly includes the forward propagation of input data and the back propagation of error, the main principle is to establish the relationship between the input data and the ideal output by constantly changing the weights
among each layer, its purpose is to keep the output data close to the ideal output so as to minimize the error. The number of hidden layers can be set according to the requirement of the project, the more the number of layers, the smaller the error, and the better the generalization ability of the network, but it will increase the training time. Therefore, the number of hidden layers should be reasonably determined by the complexity of data.

It can be seen from Figure 4, BPNN has three layers: input layer, hidden layer, and output layer, they respectively include $n$, $p$, and $m$ neurons, the connection weights are $w_{ij}$ and $w_{kj}$ limited in $(-1, 1)$.

The input and output of hidden layer are calculated according to training samples

$$net_j = h_{ij} = \sum_{i=1}^{n} w_{ij}x_i + b_j, \quad j = 1, 2, \ldots, p. \quad (11)$$

$$z_j = h_{oj} = f(net_j) = f(\sum_{i=1}^{n} w_{ij}x_i + b_j), \quad j = 1, 2, \ldots, p. \quad (12)$$

where the weight vector is $W_1 = [w_{11} \ldots w_{1n} \ w_{21} \ldots w_{2n} \ w_{p1} \ldots w_{pm}]_{1 \times (np)}$, intercept vector is $B_1 = [b_1 \ b_2 \ldots \ b_p]_{1 \times p}$. Here $net_j$ and $z_j$ denote input and output of $j$-th neuron in hidden layer, $f()$ is the transfer function, Sigmoid can be chosen here, it is

$$f(x) = \frac{1}{1 + e^{-x}}. \quad (13)$$

then reckon the input and output of the output layer

$$y_{ik} = \sum_{j=1}^{p} w_{kj}z_j + b_k, \quad k = 1, 2, \ldots, m. \quad (14)$$

$$y_k = f(y_{ik}) = f(\sum_{j=1}^{p} w_{kj}z_j + b_k), \quad k = 1, 2, \ldots, m. \quad (15)$$

where the weight vector is $W_2 = [w_{11}' \ldots w_{1p}' \ w_{21}' \ldots w_{2m}' \ w_{p1}' \ldots w_{pm}']_{1 \times (mp)}$, intercept vector is $B_2 = [b_{1}' \ b_{2}' \ldots \ b_{m}']_{1 \times m}$. here, $y_{ik}$ and $y_k$ represent input and output data of the $k$-th neuron in output layer. When we import the $p$-th group of samples, the error is

$$E_p = \frac{1}{2} \sum_{k=1}^{m} (d_{pk} - y_{pk})^2. \quad (16)$$

combining (12), (15), and (16), we have

$$E_p = \frac{1}{2} \sum_{k=1}^{m} (d_{pk} - f(\sum_{j=0}^{p} w_{kj}z_j + b_k))^2$$

$$= \frac{1}{2} \sum_{k=1}^{m} (d_{pk} - f(\sum_{j=0}^{n} w_{ij}x_i + b_j + b_k))^2. \quad (17)$$

We know from (17), $E_p$ can be improved by modifying weighting $w_{ij}$ and $w_{kj}$, when $E_p$ is obtained, the course of back propagation is implemented to decrease the error:

1. Define input partial derivative of output layer as $\delta_0$, then

$$\frac{\partial E_p}{\partial w_{kj}} = \frac{\partial E_p}{\partial y_{ik}} \frac{\partial y_{ik}}{\partial h_{ij}} \frac{\partial h_{ij}}{\partial w_{kj}}. \quad (18)$$

$$\frac{\partial E_p}{\partial y_{ik}} = \frac{\partial [\frac{1}{2} \sum_{k=1}^{m} (d_{pk} - y_{pk})^2]}{\partial y_{ik}}$$

$$= -(d_{pk} - y_{pk}) f'(y_{ik})$$

$$= -\delta_k. \quad (19)$$

$$\frac{\partial y_{ik}}{\partial h_{ij}} = \frac{\partial [\sum_{j=1}^{p} w_{kj}z_j + b_k]}{\partial h_{ij}} = z_j. \quad (20)$$

$$\frac{\partial E_p}{\partial o_k} = -\delta_0 z_j. \quad (21)$$

2. Calculate the input partial derivative $\delta_h$ of hidden layer

$$\frac{\partial E_p}{\partial w_{ij}} = \frac{\partial E_p}{\partial h_{ij}} \frac{\partial h_{ij}}{\partial w_{ij}}. \quad (22)$$

$$\frac{\partial E_p}{\partial h_{ij}} = \frac{\partial [\frac{1}{2} \sum_{k=1}^{m} (d_{pk} - y_{pk})^2]}{\partial h_{ij}}$$

$$= -(\sum_{k=1}^{m} \delta_0 w_{kj}) f'(y_{ik})$$

$$= -\delta_h. \quad (23)$$

$$\frac{\partial h_{ij}}{\partial w_{ij}} = \frac{\partial (\sum_{j=1}^{n} w_{ij}x_i + b_j)}{\partial w_{ij}} = x_i. \quad (24)$$

$$\frac{\partial E_p}{\partial w_{ij}} = -\delta_h x_i. \quad (25)$$

3. Modify the weighting $w_{kj}$ with partial derivative $\delta_0$

$$\Delta w_{kj} = -\eta \frac{\partial E_p}{\partial w_{kj}} = -\eta \delta_0 z_j. \quad (26)$$

$$w_{kj}(N + 1) = w_{kj}(N) + \eta \delta_0 z_j. \quad (27)$$

where $\eta$ means learning rate, $w_{kj}(N + 1)$ is the connection weighting after adjusting $N + 1$ times.
D. PARAMETER INITIALIZATION

Though BPNN has a strong learning ability, it is easy to fall into local optimum value, resulting in obtaining a good training result but a bad testing effect. Besides, a pair of random initial weighting and threshold will lead to a poor convergence. Therefore, PSO is employed for optimizing initial parameters of BPNN here, all the weightings in the network are encoded to vectors representing particle positions, then connection weightings are reconstituted by iterative optimization, consequently the problems of local optimum and complex computation can be processed effectively.

The realization process of PSO algorithm is mainly to simulate the foraging of birds, during the foraging process, a bird usually finds out the optimal path to the food and then attracts other birds to gather together. Thus, each bird is deemed as a particle in the vast space, and any location is accurately described by the fitness function, so that the optimal location of each particle with a certain velocity can be recorded. At this point, the moving direction is determined based on the optimal position, and the flight speed is adjusted according to the rules of the algorithm. In the process of finding the best position, each particle constantly updates its speed and location, and the final output result is a population optimal value selected from all the individuals.

Supposing there are \( N \) particles in the space, \( D \)-dimensional objective function is selected as fitness, then the optimal solution of objective function is solved through iterative updating each particle. Define the position vector and flight speed are respectively \( X_i \) and \( V_i \), that is

\[
X_i = (x_{i1}, x_{i2}, \ldots, x_{iD}), \quad i = 1, 2, \ldots, N. \tag{30}
\]

\[
V_i = (v_{i1}, v_{i2}, \ldots, v_{iD}), \quad i = 1, 2, \ldots, N. \tag{31}
\]

search the optimal individual and population positions

\[
P_i = (p_{i1}, p_{i2}, \ldots, p_{iD}), \quad i = 1, 2, \ldots, N. \tag{32}
\]

\[
P_g = (p_{g1}, p_{g2}, \ldots, p_{gD}), \quad i = 1, 2, \ldots, N. \tag{33}
\]

These particles update their speeds and positions to generate new populations

\[
v_i^d(t + 1) = v_i^d(t)w(t) + c_1r_1(p_i^d(t) - x_i^d(t)) + c_2r_2[p_g^d(t) - x_i^d(t)]. \tag{34}
\]

\[
x_i^d(t + 1) = x_i^d(t) + v_i^d(t + 1). \tag{35}
\]

where \( v_i^d(t) \) and \( x_i^d(t) \) are respectively the \( d \)-dimensional velocity and position vectors of the \( i \)-th particle after \( t \) iterations, \( w(t) \) denotes corresponding inertia weighting, \( c_1 \) and \( c_2 \) are learning factors on behalf of steps modifying best position of the particle itself and the whole population, \( r_1 \) and \( r_2 \) are randomly selected in \([0,1]\).

Figure 6 gives the flow chart of PSO, the steps are summarized below:

(1) Initialize the parameters of the particles \( c_1, c_2, \) and \( T_{\text{max}} \), original speed and position in velocity space and searching space.
(2) Evaluate the fitness of each particle according to the defined fitness function.
(3) Set the individual optimal values, then find out the global optimum location from them.
(4) Update speed and position of each particle according to (3) and (5).
(5) Compare the current value with the former one to obtain the individual optimal value \( p_{opt} \), then compare the present global optimum with the historical optimum to get global optimal position.
(6) If the iteration times meets the pre-set maximum, the result will be given, or return to (4).

To sum up, we can generalize the proposed algorithm as follows:

Step 1: Remove the uncorrelated source and noise from the received data to obtain the coherent source item;
Step 2: Decompose the covariance matrix of coherent sources, then extract multiple feature parameters from eigenvalues and eigenvectors;
Step 3: Initialize the network parameters through PSO;
Step 4: Train the network with corresponding data;
Step 5: Test the data with the trained network;
Step 6: Modify the parameters according to the testing data properly;

As the proposed algorithm employ multiple feature extraction, BPNN, and PSO for coherent node signals, we can abbreviate it MBPC. It mainly includes solving covariance, extracting coherent features, and training, so the complexity is about \( M^2Z + 3M + MDT_{max}T_{LM} + M \), where \( Z \) is the snapshots number. The proposed algorithm uses virtual smoothing technique, so no matter how many coherent and uncorrelated signals there are, it only needs to satisfy \( M > K \) and has no special requirements to the modulation mode.

E. DATA PREPROCESSING AND PARAMETER SETTING

1) DATA PREPROCESSING
In order to transform the data with different orders of magnitude into a same order which can be computed directly, as well as excluding the interferences, we need to convert the original data in the domain \([-1, +1]\) by

\[
y = \frac{(y_{max} - y_{min}) \times (x - x_{min})}{(x_{max} - x_{min})} + y_{min}.
\]

2) DETERMINE THE NUMBER OF NEURONS IN EACH LAYER
Three classical layers are used here, we will estimate node number according to the differences between signal eigenvalues, noise eigenvalues, and their corresponding eigenvectors. In this article, 23 sets of data including eigenvalues, the ratio of adjacent eigenvalues and mean square error of the eigenvector envelope are picked, that means number of neurons in input layer \( n = 23 \). The output is the node number, namely neuron number in output layer \( m = 1 \). For the sake of decrease the complexity of the network, only one hidden layer is set. Then we should choose a reasonable neuron number, if they are too small, we can not acquire a good result, else, a long time will be taken, so the formula for neuron number in hidden layer is

\[
p = \sqrt{n + m + \alpha}.
\]

where \( p \), \( n \), and \( m \) are respectively neuron numbers of the three layers. According to the reality, \( \alpha \) is chosen in \([1, 10]\), \( p \) is set to be 10.

3) TRANSFER FUNCTION
In the course of constructing BPNN, the non-linear activation function Sigmoid is used here, which maps the data in \([-1, +1]\) and makes input variable reflect the relation between input data and output data.

4) PARAMETER SETTING FOR BPNN
For the sake of ensuring the stability of the network and rate of convergence, learning rate is suggested to be 0.4, the maximal learning times \( T_{LM} = 2000 \), minimum desired error is 0.001.

5) PARAMETER SETTING FOR PSO
On the basis of actual sample scale, define the size of particle population as \( N = 30 \), and set the connection weighting,
Threshold, particle position, speed in $[-1, +1]$. Acceleration factor $c_1 = 2, c_2 = 1.8$, the maximal iteration times $T_{\text{max}} = 100$.

6) COLORED NOISE MODEL

Colored noise widely exists in reality, corresponding covariance matrix of received data is

$$R = ARSA^H + R_N. \quad (38)$$

where $R_N$ is that of colored noise, the elements on its diagonal are no longer equal any more. Referring to [38], we express their elements as

$$\eta_{ik} = \sigma_n^2 \rho_{|i-k|} \exp(j(i-k)0.77\pi), \quad (i, k = 1, 2, \ldots, M). \quad (39)$$

here $\sigma_n^2$ is the colored noise power, $\rho$ is the spatial correlation coefficient in $[0,1]$.

IV. SIMULATION

Supposing that our ULA is composed of eight omnidirectional sensors with spacing $d = \lambda/2$, $\lambda$ is the wavelength of the node signal, matlab environment, 80% of the snapshots is used for training and others is served for testing, 500 Monte-carlo trials, Modified MDL criteria[39] based on Forward-Backward spatial smoothing(MFBS)[40] and MBPC are separately employed for the estimation.

Example 1 Estimation success probability for the node signals with large angle intervals

Assuming that three far-field narrow-band node signals arrive at the array from $(20^\circ, 40^\circ, 60^\circ)$, the former two are coherent with each other, the other one is uncorrelated with them, Gaussian white noise(WN) environment, the success probabilities of the two algorithms versus SNR are shown in Figure 7 when snapshots is 200, Figure 8 gives that versus snapshots number when SNR is 10dB. Meanwhile, the same simulations under additive colored noise(CN) environment are exhibited in Figure 9 and Figure 10.

It is seen from Figure 7, MFBS can not estimate the node signal completely when SNR is lower than 2dB, and it reaches 100% after 6dB. While the proposed MBPC algorithm eliminates the noise, corresponding to enhance the SNR,
and PSO is exploited for the BPNN initialization, so it has a higher success probability from beginning to end. Figure 8 demonstrates that MBPC performs more better than MFBS under circumstance of small snapshots. We also know from Figure 9 and Figure 10, as a result of removing noise item, they are nearly the same with the results in WN. But the eliminating effect to the CN is inferior to the WN in low SNR or small snapshots, so the estimation results in WN is a little better than that in CN.

Example 2 Estimation success probability for the node signals with small angle intervals

Similarly, assuming that three far-field narrow-band node signals arrive at the array from \((19^\circ, 20^\circ, 40^\circ)\), the latter two nodes are coherent with each other, the first one is uncorrelated with them, other conditions are the same with example 1, Figure 11 and Figure 12 display the simulation results under WN, while Figure 13 and Figure 14 show that under CN environment.

We know from Figure 11 and Figure 12, as the latter two nodes are near to each other, their steering vectors of MFBS are correlated with one another more or less, then the covariance matrix after forward-backward smoothing still exists rank deficiency, thus their number and locations can not be distinguished. However, MBPC has removed the uncorrelated part, so it is not affected by the angle interval caused by uncorrelated node, in addition, its performance is still excellent at the background of lower SNR or small snapshots. It also can be observed from Figure 13 and Figure 14, on account of removing noise item, they are nearly the same with the results in WN.

V. CONCLUSION

For the sake of determine the coherent node number of wireless sensor network in the environment of colored noise and uncorrelated interferences, a novel algorithm based on BPNN is proposed, it gets rid of uncorrelated information and noise, then utilizes the strong learning ability to make a decision. Meanwhile, PSO is employed for the parameter initialization of BPNN, avoiding the problem of falling into local extremum. The new algorithm still performs well in the
surroundings of low SNR, small snapshots, and near uncorrelated interferences to some extent. Moreover, the trained network is adopted here so as to reduce the complexity and improve the estimation efficiency, promoting its practical applications in future.

ACKNOWLEDGMENT

The authors thank the editors and reviewers for their advices.

REFERENCES

[1] A. Raja Basha, “Energy efficient aggregation technique-based realisable secure aware routing protocol for wireless sensor network,” IET Wireless Sensor Syst., vol. 10, no. 4, pp. 166–174, Jul. 2020.

[2] N. Saeed, H. Nam, T. Y. Al-Naffouri, and M.-S. Alouini, “A state-of-the-art survey on multidimensional scaling-based localization techniques,” IEEE Commun. Surveys Tuts., vol. 21, no. 4, pp. 3565–3583, Jun. 2019.

[3] K. He, Z. Wang, D. Li, F. Zhu, and L. Fan, “Ultra-reliable mu-MIMO detector based on deep learning for 5G/BSG-enabled IoT,” Phys. Commun., Aug. 2020, Art. no. 101180.

[4] L. S. Fan, J. X. Xia, N. Yang, Y. S. Deng, T. Q. Duong, G. K. Karagiannidis, and A. Nallagatla, “Opportunistic access point selection for mobile edge computing networks,” IEEE Trans. Wireless Commun., to be published.

[5] A. Bekerche, Q. Wu, and P. Sun, “Efficient green protocols for sustainable wireless sensor networks,” IEEE Trans. Sustain. Comput., vol. 5, no. 1, pp. 61–80, Jan. 2020.

[6] J. M. C. Geoffrime and V. Geetha, “Energy optimization with higher information quality for SHM application in wireless sensor networks,” IEEE Sensors J., vol. 19, no. 9, pp. 3513–3520, May 2019.

[7] Z. Na, Y. Liu, J. Shi, C. Liu, and Z. Gao, “UAV-supported clustered NOMA for 6G-enabled Internet of Things: Trajectory planning and resource allocation,” IEEE Internet Things J., early access, Jun. 23, 2020, doi: 10.1109/IOT.2020.3004432.

[8] Z. Na, J. Wang, C. Liu, M. Guan, and Z. Gao, “Join trajectory optimization and communication design for UAV-enabled OFDM networks,” Ad Hoc Netw., vol. 98, Mar. 2020, Art. no. 102031.

[9] C. R. Anderson, H. I. Volos, and R. M. Bachehr, “Characterization of low antenna ultrawideband propagation in a forest environment,” IEEE Trans. Veh. Technol., vol. 62, no. 7, pp. 2878–2895, Sep. 2013.

[10] D. P. Smith, G. G. Messier, and M. W. Wasson, “Boreal forest low antenna height propagation measurements,” IEEE Trans. Antennas Propag., vol. 64, no. 9, pp. 4004–4011, Sep. 2016.

[11] A. H. Dehval, M. Mouss, and C. G. Claudel, “Lessons learned on solar powered wireless sensor network deployments in urban, desert environments,” Ad Hoc Netw., vol. 28, pp. 52–67, May 2015.

[12] U. Jain and M. Hussain, “Securing wireless sensors in military applications through resilient authentication mechanism,” Procedia Comput. Sci., vol. 171, pp. 719–728, Jan. 2020.

[13] D. Orfanus, E. P. de Freitas, and F. Eliassen, “Self-organization as a supporting paradigm for military UAV relay networks,” IEEE Commun. Lett., vol. 20, no. 4, pp. 804–807, Apr. 2016.

[14] T.-H. Tsai, C.-C. Huang, C.-H. Chang, and M. A. Hussain, “Design of wireless vision sensor network for smart home,” IEEE Access, vol. 8, pp. 60455–60467, Mar. 2020.

[15] V. Bianchi, P. Ciampolini, and I. De Munari, “RSSI-based indoor localization and identification for ZigBee wireless sensor networks in smart homes,” IEEE Trans. Instrum. Meas., vol. 68, no. 2, pp. 566–575, Feb. 2019.

[16] H. Kapil and C. S. R. Murthy, “A pragmatic relay placement approach in 3-D space and Q-Learning-Based transmission scheme for reliable factory automation applications,” IEEE Syst. J., vol. 12, no. 1, pp. 823–833, Mar. 2018.

[17] A. Aijaz, “ENCLOSE: An enhanced wireless interface for communication in factory automation networks,” IEEE Trans. Ind. Informat., vol. 14, no. 12, pp. 5346–5358, Dec. 2018.

[18] D. Sharma and A. P. Bhondekar, “Traffic and energy aware routing for heterogeneous wireless sensor networks,” IEEE Commun. Lett., vol. 22, no. 8, pp. 1608–1611, Aug. 2018.
JIAQI ZHEN (Member, IEEE) was born in Harbin, Heilongjiang, China, in 1981. He received the master’s and Ph.D. degrees from the College of Information and Communication Engineering, Harbin Engineering University, Harbin, in 2007 and 2011, respectively.

From 2011 to 2015, he was a Lecturer in the major of communication engineering with Heilongjiang University, Harbin. Since 2015, he has been an Associate Professor in the major of the Internet of Things engineering with Heilongjiang University, and the Director of the Institute of Instrumentation. He has published an academic monograph, two textbooks, more than 30 articles, and 16 inventive patents. His research interests include spatial spectrum estimation, indoor positioning, array signal parameter estimation, and embedded system design. He received the Best Paper in the 2nd EAI International Conference on Advanced Hybrid Information Processing in 2018.

XIAOLI ZHANG was born in Linyi, Shandong, China, in 1995. She received the B.S. degree from the Department of Electronic Information Science and Technology, Qufu Normal University, Qufu, Shandong, in 2017. She is currently pursuing the master’s degree with Heilongjiang University. Her research interests include super-resolution direction finding technology and spatial spectrum estimation.

* * *