IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 17, 2020, accepted November 8, 2020, date of publication November 11, 2020,
date of current version November 19, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3037337

Diffusion ACNSAF Algorithm for Secure Distributed Estimation Under Complex-Valued Attacks

PENGWEI WEN[1], **JIASHU ZHANG**[2], **SHENG ZHANG**[2], **(Member, IEEE), AND GAO CHEN**[3]
[1]School of Electronic and Information Engineering, Zhongyuan University of Technology, Zhengzhou 450007, China
[2]Sichuan Province Key Laboratory of Signal and Information Processing, Southwest Jiaotong University, Chengdu 610031, China
[3]School of Electrical Engineering and Intelligentization, Dongguan University of Technology, Dongguan 523808, China

Corresponding authors: Jiashu Zhang (jszhang@home.swjtu.edu.cn) and Sheng Zhang (dr.s.zhang@ieee.org)

This work was supported in part by the National Science Foundation of China under Grant 61671392, Grant 61801401, Grant 62001528, and Grant 62071396; in part by the National Science Foundation of Henan Province under Grant 202300410517, and in part by the Key Scientific Research Projects in Colleges and Universities of Henan Province under Grant 21A120011.

ABSTRACT In order to protect the distributed network from complex-valued attack, a secure diffusion augmented complex-valued normalized subband adaptive filter algorithm is proposed in this article, which is derived from a novel complex-valued detection method. This complex-value detection method can be considered to consist of two parts. First, a non-cooperative augmented complex-valued normalized subband adaptive filter algorithm is used to provide reliable reference estimations under complex-valued attacks, and then the threshold test is constructed by using the reliable reference estimations to detect the trustworthy neighbors of each node. The second part is to reassemble the information from the trustworthy neighbors by using the diffusion augmented complex-valued normalized subband adaptive filter algorithm. Then, the theoretical analyses of the mean and mean-square performance of the proposed algorithm are derived based on the energy conservation framework. Finally, some experiments are performed to show the effectiveness of the proposed algorithm under non-time-varying and time-varying complex-valued attacks, respectively.

INDEX TERMS Adaptive networks, noncircular attacks, widely linear model, non-cooperative, augmented complex-valued normalized subband adaptive filter, threshold test.

I. INTRODUCTION

Recent advancements in wireless sensor networks (WSNs) have shown that it is an attractive and challenging research area. One of the key technologies of WSN is distributed adaptation over networks. In adaptive networks, the interconnected nodes can continually learn and adapt independently or collaboratively estimate some parameters of interest from observations collected by the dispersed agents [1]–[3]. The distributed strategies for data processing over networks can be divided into incremental strategy, consensus strategy and diffusion strategy [4]. Since the incremental strategy requires a cyclic path that runs across the nodes to obtain the parameter estimation, it is sensitive to the link failures. Compared to the incremental strategy, the diffusion strategy is more flexible, robust and energy efficient [5]. Besides, diffusion strategy has been shown to have superior stability and performance improvement than the consensus-based implementations. Due to these merits, the diffusion strategy is widely used in many applications, such as target tracking, environment monitoring, wireless and sensor networks [6]–[10].

Based on the diffusion strategy, many diffusion adaptive algorithms have been proposed, such as diffusion least mean square (D-LMS) algorithm [11], diffusion recursive least square (D-RLS) algorithm [12], diffusion total least square (D-TLS) algorithm [13], diffusion affine projection (D-APA) algorithm, diffusion Kalman filter and so on [8]. All of these studies were carried out under the assumption that the network environment was secure (not attacked). That is to say, these diffusion algorithms achieve good performance in convergence rate and steady-state error under a secure environment. However, the wireless sensor itself and communication between wireless sensors are very vulnerable. Therefore, it is possible that the attack will cause the information

The associate editor coordinating the review of this manuscript and approving it for publication was Yingsong Li.

received by the sensor from the adjacent sensors to be biased, resulting in inaccurate estimation of the whole network [14]. If these attacks occur, the existing distributed algorithms will provide biased estimates because the network cannot effectively defend against attacks. Therefore, it is necessary to develop secure distributed algorithms with low computational complexity.

In order to process distributed network information safely, several secure distributed algorithms have been proposed. Under the Byzantine attacks, distributed detection over sensor networks has been studied in [15], [16]. Due to the Bayesian detection method, the Byzantine attacks are well detected. When attacks occur in cognitive radio networks, secure collaborative spectrum sensing was proposed in [17] by using cooperative diversity between cognitive radio spectrum sensors. By using information from neighbors to create adaptive combination weights that are inversely proportional to the distance between the instantaneous local estimation and the reference estimation, a robust reputation-based dLMS (R-dLMS) algorithm was proposed to achieve secure distributed estimation under malicious attacks in [18]. Because the reference estimation comes from instantaneous local estimation, it will be unreliable when the number of attacks is large. To solve this problem, a novel detection method using the non-cooperative LMS (nc-LMS) to provide reference estimation was developed and the secure D-LMS (SD-LMS) algorithm was proposed by Liu and Li in [19]. The SD-LMS algorithm is well solved, no matter whether the attacks are on the sensor or the transmission path. However, The SD-LMS algorithm is aimed at attacks in real domain. The definition of complex-valued attacks is proposed in smart grid [20]. A Kalman filter (KF) estimation/monitoring solution was proposed by using Euclidean detector when the attacks are assumed to be false data injection [21]. In real life, complex-valued signals are mostly noncircular. In [22], it is impractical to ignore the correlation between the real part and the imaginary part of the complex-valued attack signal. Subsequently, a noncircular attack model was proposed and the corresponding state estimation method was presented in [23].

In this article, to give a good estimation performance over WSNs in the presence of circular or noncircular complex-valued attacks, a novel detection method is proposed, and a secure diffusion augmented complex-valued normalized subband adaptive filter (SD-ACNSAF) algorithm is derived. The SD-ACNSAF can be considered to consist of two parts. First, it needs a non-cooperative ACNSAF (nc-ACNSAF) algorithm to provide reliable reference estimation, which is further used for constructing the threshold test to detect the trustworthy neighbors of each node. The second part is to reassemble the information from the trustworthy neighbors by using the D-ACNSAF algorithm. The main contributions of this article are summarized below. 1) By using a novel threshold test, the SD-ACNSAF algorithm is proposed for protecting distributed network from complex-valued attacks. Besides, an adaptive way to select the threshold is suggested. 2) The theoretical analyses of the mean and mean-square performance of the proposed algorithm are presented.

This article is organized as follows. Section II reviews some attack models. In Section III, a SD-ACNSAF algorithm is proposed to achieve secure distributed estimation under attacks. Section IV illustrates the stability and steady-state performance analysis of the proposed algorithm. Section V illustrates the simulation results obtained by using the proposed algorithm, and Section VI presents some conclusions.

Notation: Throughout this article, matrices and column vectors are defined by boldface capital and small boldface, respectively. $\Re(\cdot)$ and $\Im(\cdot)$ represent the real and imaginary parts, respectively. $(\cdot)^{(m)}$ denotes the *m-th* element of a vector. $(\cdot)^{*}$ stands for the complex conjugate. The superscript $(\cdot)^{T}$ and $(\cdot)^{H}$ stands for transposition and Hermitian transposition. $E[\cdot]$ represents the expectation of $[\cdot]$.

II. PROBLEM FORMULATION

Fig. 1 shows a topology of the wireless sensor network that has not been attacked. It contains N sensors. A set Γ_k is defined as including the sensor node k and its adjacent sensors. In the set, each node can interchange information with their neighbor nodes, the information may be real-value or complex-valued measurement vectors. Assuming all the measurements are complex-valued in the article, and each node k obtains an observed complex-valued output signal $d_k(n)$ and a regression complex-valued vector $u_k(n)$ at time instant n, where $u_k(n)$ is an M-dimensional column vector. The complex-valued output signal $d_k(n)$ can be represented by the following widely linear (WL) model

$$d_k(n) = u_k^T(n)h_o + u_k^H(n)g_o + \upsilon_k(n) \tag{1}$$

where h_o and g_o are the M-dimensional unknown column vectors to be estimated, $\upsilon_k(n)$ is the additive background noise with the variance $\sigma_{\upsilon,k}^2$ at agent k. When the M-dimensional unknown column vector g_o is 0, (1) becomes a general linear model.

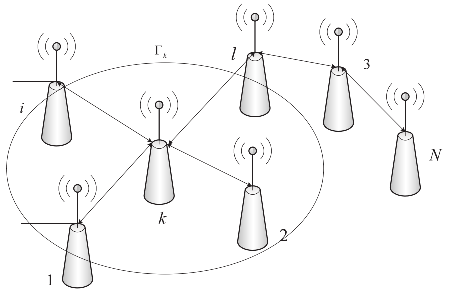


FIGURE 1. Example of distributed sensor network topology. Each agent *k* exchanges data with neighbor nodes.

In order to deal with circular and non-circular complex-valued signals effectively, the D-ACNSAF[1] algorithm is

[1] A more detailed description of D-ACNSAF algorithm can be found in [24].

introduced according to the adaptive-then-combine (ATC) diffusion strategy [24]

Adaptive step:

$$\underline{h}_k(n+1) = h_k(n) + \mu_k \sum_{m=1}^{N_s} \frac{e_{k,m,D}(n)}{\|u_{k,m}(n)\|^2} u^*_{k,m}(n) \quad (2)$$

$$\underline{g}_k(n+1) = g_k(n) + \mu_k \sum_{m=1}^{N_s} \frac{e_{k,m,D}(n)}{\|u_{k,m}(n)\|^2} u_{k,m}(n) \quad (3)$$

Combination step:

$$h_k(n+1) = \sum_{l \in \Gamma_k} b_{k,l} \underline{h}_l(n+1) \quad (4)$$

$$g_k(n+1) = \sum_{l \in \Gamma_k} b_{k,l} \underline{g}_l(n+1) \quad (5)$$

where $\underline{h}_k(n)$ and $\underline{g}_k(n)$ are the estimates of intermediate weights, μ_k is the step-size at agent k. $b_{k,l}$ is the combination parameter and satisfies $\sum_{l \in \Gamma_k} b_{k,l} = 1$ and $b_{k,l} = 0$ if $l \notin \Gamma_k$. N_s is the number of subbands, $e_{k,m,D}(n)$ is the subband error signal. $u_k(n)$ is partitioned into N_s subband signal $u_{k,m}(n)$ through the analysis filters for $m = 1, 2, \ldots, N_s$. D is the down sampling.

The D-ACNSAF algorithm is derived without attack. In this article, we consider the existence of complex attacks in wireless sensor network.

Generally, malicious attacks include two ways, bad data attack and false data injection attack. As the bad data attack replaces $d_k(n)$ with the random data $d_k^+(n)$, it can be easily detected by verifying the consistency of the l_2 norm of the attacked error signal [25]. However, this detection method cannot be used for false data injection attack[2] [26]. In this article, the complex-valued false data injection attack is studied. The complex-valued false data injection attack can be modeled as follows [23]:

$$z_k^a(n) = H_k(n) z_k(n) + \tilde{H}_k^H(n) z_k^* \quad (6)$$

where $z_k(n)$ represents the complex-valued measurement vector, $H_k(n)$ and $\tilde{H}_k(n)$ are attack matrices. When $H_k(n)$ and $\tilde{H}_k(n)$ are not equal to zero, $z_k^a(n)$ denotes the noncircular complex-valued attack signal. Otherwise $z_k^a(n)$ is the circular complex-valued attack signal.

Considering the general situation, attacks generally occur in transmission paths (called compromised communications) and sensors (called compromised sensors). The compromised sensors and compromised communications are defined as follows:

1) *Compromised sensors:* The sensor that is transmitted fictitious data by an intruder is considered to be attacked. For example, if node k is attacked, it can be modelled by the following equation

$$d_k'(n) = d_k(n) + d_k''(n) \quad (7)$$

where $d_k(n)$ is the true desired output signal. $d_k''(n)$ is the additional desired signal after attack, which is defined as a linear combination of input signal and M-dimensional error vector $q_k(n)$. The formula of $q_k(n)$ is similar to that of $z_k^a(n)$.

2) *Compromised communications:* The communication is attacked between the node and its adjacent node when the transmitted information between them is disturbed. For example, when the communication between k and l is compromised, real information from l received by node k can be represented as

$$\underline{h}_k^r(n) = \underline{h}_l(n) + z_{k,l}^a(n) \quad (8)$$

$$\underline{g}_k^r(n) = \underline{g}_l(n) + z_{k,l}^{a'}(n) \quad (9)$$

where $z_{k,l}^a(n)$ and $z_{k,l}^{a'}(n)$ are the mutual independent interference complex-valued signals with the same mean and variance $\sigma^2_{z_{k,l}}$.

To make the attack model more similar to the actual situation, the following assumptions are presented.

A1: For each node k, attacks cannot occur at an adjacent node l and its transmission path to node k at the same time. Besides, the number of the attacks that occur in the set Γ_k is less than $\lceil \frac{n_k}{2} \rceil$, where n_k is the cardinality of the set Γ_k.

A2: The attacker does not have the complete knowledge of the data model.

A3: The background noise and the input signal are complex-valued vectors and spatially mutual independent [27], [28].

In order to detect attacks to the greatest extent, the assumption $A1$ is necessary for the proposed algorithm on the secure network topology. The detailed explanation is given in the following part. Since the transmitted information is time-varying and the attacker may not be so efficient, the assumption $A2$ is resonable and has been used in attack hypotheses [29], [30]. In the following part, all the attacks occur under the assumption $A2$. The assumption $A3$ is widely used in the analysis of complex adaptive filtering, and it is reasonable.

According to the correlation between real part and imaginary part, complex-valued signals are divided into circular or non-circular signals [31]. Therefore, to make full use of the available second-order information, a novel detection method suitable for complex-valued signals is proposed to detect the circular and noncircular attacks in the following part.

III. DERIVATION OF SECURE D-ACNSAF ALGORITHM

In this section, a secure D-ACNSAF algorithm is proposed to ensure the reliability of information exchange between sensors in an attacked environment. The SD-ACNSAF algorithm contains nc-ACNSAF and D-ACNSAF algorithm. The nc-ACNSAF algorithm is D-ACNSAF algorithm without information exchange, which provides reliable reference estimation. The detailed explanation is given in Remark 1. Then, the D-ACNSAF algorithm uses reliable estimations from self-update.

[2] The false data injection attack can send some malicious data satisfying the model (1). When l_2 norm detection of error signal is used, the false data injection attack is robust to the detection method.

A. NC-ACNSAF ALGORITHM

In order to accurately detect the trustworthy adjacent nodes of each node, similar to the method of dealing with real-valued attacks in [19], an nc-ACNSAF subsystem is used to provide reliable reference estimates. The nc-ACNSAF[3] subsystem that each node can construct an ACNSAF local adaptive rule is used to provide reliable reference estimates and given as follows:

$$h_k(n+1) = h_k(n) + \mu_k \sum_{m=1}^{N_s} \frac{e_{k,m,D}(n)}{\|u_{k,m}(n)\|^2} u_{k,m}^*(n) \tag{10}$$

$$g_k(n+1) = g_k(n) + \mu_k \sum_{m=1}^{N_s} \frac{e_{k,m,D}(n)}{\|u_{k,m}(n)\|^2} u_{k,m}(n) \tag{11}$$

The D-ACNSAF subsystem is used to achieve secure distributed estimation over network. Assuming that attacks on D-ACNSAF subsystem also occur on nc-ACNSAF subsystem.

Remark1: Since the nc-ACNSAF only uses its own information and does not need a combination strategy to update the weights, other nodes are unaffected when a node is attacked. The nc-ACNSAF is suitable for providing the reliable real-time reference estimates to monitor system changes.

B. SD-ACNSAF ALGORITHM

Since the weight estimation of the D-ACNSAF is determined by combining the intermediate weights of adjacent secure nodes, it cannot be used as reference estimation. For each node k, the reliable reference estimates $\overline{w}_k(n)$ and $\overline{w}'_k(n)$ (which are given below) can be obtained based on the received estimates $h_l^r(n)$ and $g_l^r(n)$ from the neighbors of the nc-ACNSAF subsystem. Once $\overline{w}_k(n)$ and $\overline{w}'_k(n)$ are available, the reliable neighbors for each node are detected via a threshold test constructed on the reference estimates. Since the intermediate weights are in the form of complex values, the real and the imaginary parts of the reference estimates should be obtained separately and then recombined. Firstly, each node k receives the estimates from its neighbors of the nc-ACNSAF subsystem, the real and imaginary parts of each element j of these intermediate weights including itself are sorted, respectively. The progress is given as follows (see the diagram given in Fig. 2):

$$w_{k1}^{(j)}(n) = \left[\Re(h_{l_1}^{r(j)}), \ldots, \Re(h_{l_s}^{r(j)}), \Re(h_{l_t}^{r(j)}), \ldots, \Re(h_{l_{n_k}}^{r(j)})\right]$$
$$w_{k2}^{(j)}(n) = \left[\Im(h_{l_1}^{r(j)}), \ldots, \Im(h_{l_s}^{r(j)}), \Im(h_{l_t}^{r(j)}), \ldots, \Im(h_{l_{n_k}}^{r(j)})\right]$$
$$w_k^{(j)}(n) = w_{k1}^{(j)}(n) + iw_{k2}^{(j)}(n)$$
$$w_{k1}^{'(j)}(n) = \left[\Re(g_{l_1}^{r(j)}), \ldots, \Re(g_{l_s}^{r(j)}), \Re(g_{l_t}^{r(j)}), \ldots, \Re(g_{l_{n_k}}^{r(j)})\right]$$
$$w_{k2}^{'(j)}(n) = \left[\Im(g_{l_1}^{r(j)}), \ldots, \Im(g_{l_s}^{r(j)}), \Im(g_{l_t}^{r(j)}), \ldots, \Im(g_{l_{n_k}}^{r(j)})\right]$$
$$w_k^{'(j)}(n) = w_{k1}^{'(j)}(n) + iw_{k2}^{'(j)}(n)$$

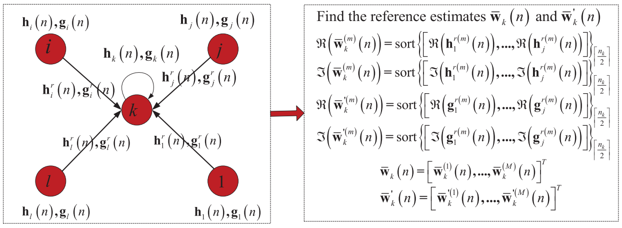

$\mathbf{h}_i(n), \mathbf{g}_i(n)$
$\mathbf{h}_j(n), \mathbf{g}_j(n)$
i
j
$\mathbf{h}_k(n), \mathbf{g}_k(n)$
$\mathbf{h}_i^r(n), \mathbf{g}_i^r(n)$
$\mathbf{h}_j^r(n), \mathbf{g}_j^r(n)$
k
$\mathbf{h}_l^r(n), \mathbf{g}_l^r(n)$
$\mathbf{h}_1^r(n), \mathbf{g}_1^r(n)$
l
1
$\mathbf{h}_l(n), \mathbf{g}_l(n)$
$\mathbf{h}_1(n), \mathbf{g}_1(n)$
Find the reference estimates $\overline{\mathbf{w}}_k(n)$ and $\overline{\mathbf{w}}'_k(n)$
$\Re\left(\overline{\mathbf{w}}_k^{(m)}(n)\right) = \text{sort}\left\{\left[\Re\left(\mathbf{h}_1^{r(m)}(n)\right), \ldots, \Re\left(\mathbf{h}_j^{r(m)}(n)\right)\right]\right\}_{\lceil \frac{n_k}{2} \rceil}$
$\Im\left(\overline{\mathbf{w}}_k^{(m)}(n)\right) = \text{sort}\left\{\left[\Im\left(\mathbf{h}_1^{r(m)}(n)\right), \ldots, \Im\left(\mathbf{h}_j^{r(m)}(n)\right)\right]\right\}_{\lceil \frac{n_k}{2} \rceil}$
$\Re\left(\overline{\mathbf{w}}_k^{'(m)}(n)\right) = \text{sort}\left\{\left[\Re\left(\mathbf{g}_1^{r(m)}(n)\right), \ldots, \Re\left(\mathbf{g}_j^{r(m)}(n)\right)\right]\right\}_{\lceil \frac{n_k}{2} \rceil}$
$\Im\left(\overline{\mathbf{w}}_k^{'(m)}(n)\right) = \text{sort}\left\{\left[\Im\left(\mathbf{g}_1^{r(m)}(n)\right), \ldots, \Im\left(\mathbf{g}_j^{r(m)}(n)\right)\right]\right\}_{\lceil \frac{n_k}{2} \rceil}$
$\overline{\mathbf{w}}_k(n) = \left[\overline{\mathbf{w}}_k^{(1)}(n), \ldots, \overline{\mathbf{w}}_k^{(M)}(n)\right]^T$
$\overline{\mathbf{w}}'_k(n) = \left[\overline{\mathbf{w}}_k^{'(1)}(n), \ldots, \overline{\mathbf{w}}_k^{'(M)}(n)\right]^T$

FIGURE 2. **Example of finding the reference estimates of node *k*.**

where $\Re(h_{l_s}^{r(j)}) < \Re(h_{l_t}^{r(j)})$, $\Im(h_{l_s}^{r(j)}) < \Im(h_{l_t}^{r(j)})$, $\Re(g_{l_s}^{r(j)}) < \Re(g_{l_t}^{r(j)})$, $\Im(g_{l_s}^{r(j)}) < \Im(g_{l_t}^{r(j)})$ and $l_1, l_s, l_t, l_{n_k} \in \Gamma_k$.

Under complex-valued attack, the estimator of the unknown parameter vector will deviate from its true value. In other words, it is very possible that the attacked estimators should be on the left or right side of the sets. Under the assumption $A1$, the central $\lceil \frac{n_k}{2} \rceil$th estimates of the sets are reliable with a high probability.

Based on the above statement, the references $\overline{w}_k(n)$ and $\overline{w}'_k(n)$ of the *j-th* component at node k are chosen as

$$\overline{w}_k^{(j)}(n) = w_{\lceil \frac{n_k}{2} \rceil}^{(j)}(n) \tag{12}$$

$$\overline{w}_k^{'(j)}(n) = w_{\lceil \frac{n_k}{2} \rceil}^{'(j)}(n) \tag{13}$$

Repeating the above process for each entity m, the reliable reference vectors $\overline{w}_k(n) = [\overline{w}_k^{(1)}(n), \cdots, \overline{w}_k^{(M)}(n)]^T$ and $\overline{w}'_k(n) = [\overline{w}_k^{'(1)}(n), \cdots, \overline{w}_k^{'(M)}(n)]^T$ can be obtained. It is obvious that the reference estimates are only determined by the nc-ACNSAF subsystem, and they can be considered as weight estimation at time n without attack. Therefore, the reference vectors can be considered as reliable value for detecting whether the network is attacked or not (The detailed explanation is given in Appendix).

Considering that the estimated weights will converge to the deviation of the optimum weights under attacks, by using the principle of "*consistency*" [24], attacks that occur at nodes or communications can be detected. Firstly, defining the random error variable $s_{k,l}(n)$ as follows

$$s_{k,l}(n) = h_l^{r(j_1)}(n) - \overline{w}_k^{(j_1)}(n) \tag{14}$$

where j_1 is the index of the maximum element-wised l_1-norm distance between $h_l^{r(j_1)}(n)$ and $\overline{w}_k^{(j_1)}$ among M elements at time n, i.e.

$$\max_m \left\{ \left| h_l^{r(j)}(n) - \overline{w}_k^{(j)}(n) \right| \right\} = \left| h_l^{r(j_1)}(n) - \overline{w}_k^{(j_1)}(n) \right| \tag{15}$$

Similarly, $s'_{k,l}(n)$, $s''_{k,l}(n)$ and $s'''_{k,l}(n)$ can be defined as follows:

$$s'_{k,l}(n) = g_l^{r(j_2)}(n) - \overline{w}_k^{'(j_2)}(n) \tag{16}$$

$$s''_{k,l}(n) = \underline{h}_l^{r(j_3)}(n) - \overline{w}_k^{(j_3)}(n) \tag{17}$$

$$s'''_{k,l}(n) = \underline{g}_l^{r(j_4)}(n) - \overline{w}_k^{'(j_4)}(n) \tag{18}$$

where j_2 is the index of the maximum element-wised l_1-norm distance between $g_l^{r(j)}(n)$ and $\overline{w}_k^{'(j)}(n)$ among M elements, j_3

[3] The nc-ACNSAF algorithm can be considered as several independent ACNSAF algorithms updating simultaneously. A more detailed description of ACNSAF algorithm can be found in [32].

is the index of the maximum element-wised l_1-norm distance between $\underline{h}_l^{r(j)}(n)$ and $\overline{w}_k^{(j)}(n)$ among M elements at time n, j_4 is the index of the maximum element-wised l_1-norm distance between $\underline{g}_l^{r(j)}(n)$ and $\overline{w}_k^{'(j)}(n)$ among M elements at time n.

If an effective attack occurs on node k or the transmission path between node k and node l, these values $s_{k,l}(n)$, $s'_{k,l}(n)$, $s''_{k,l}(n)$ and $s'''_{k,l}(n)$ will become very large. Taking both ACNSAF and D-ACNSAF subsystems into account, the following random variables are defined

$$b_{k,l}(n) = \begin{cases} s''_{k,l}(n), & \text{if } |s''_{k,l}(n)| > |s_{k,l}(n)| \\ s_{k,l}(n), & \text{otherwise} \end{cases} \tag{19}$$

$$b'_{k,l}(n) = \begin{cases} s'''_{k,l}(n), & \text{if } |s'''_{k,l}(n)| > |s'_{k,l}(n)| \\ s'_{k,l}(n), & \text{otherwise} \end{cases} \tag{20}$$

Since the estimated weights and conjugate estimated weights are updated simultaneously, the final constraints are given

$$c_{k,l}(n) = \begin{cases} b'_{k,l}(n), & \text{if } |b'_{k,l}(n)| > |b_{k,l}(n)| \\ b_{k,l}(n), & \text{otherwise} \end{cases} \tag{21}$$

Then, the compromised sensor or communication can be detected by using the following threshold test:

$$T_{k,l}(n) = c^2_{k,l}(n) \overset{\mathcal{H}_2}{\underset{\mathcal{H}_1}{\gtrless}} \gamma_k(n) \tag{22}$$

where $\mathcal{H}_1$ represents the normal operation scenario, $\mathcal{H}_2$ refers to the case that the system is potentially under attack. $\gamma_k(n)$ is a positive threshold and the detailed selection method given in Section IV-D.

Then, the set of reliable neighbors of node k at time n can be obtained as follows:

$$\Gamma'_k(n) = \left\{ l \in \Gamma_k | T_{k,l}(n) < \gamma_k(n) \right\} \tag{23}$$

It's obvious that the instantaneous set $\gamma_k(n)$ of trustworthy neighbors is time-varying in an adversarial environment. The combination rules based on the detected secure network topology should be redefined. Defining the degree $\varepsilon_k(n)$ of reliable node k, the combination weights $b_{k,l}$ can be designed according to the original combination rules with $\varepsilon_k(n) > 0$. If there is no reliable nodes for node k, we set $\underline{h}_k^r(n) = \overline{w}_k(n)$, $\underline{g}_k^r(n) = \overline{w}'_k(n)$, $b_{k,k} = 1$ and $b_{k,l} = 0$ for $l \neq k$. That is to say, the estimate of D-ACNSAF subsystem is equivalent to that of the reference.

Based on the coefficients $b_{k,l}$, $\underline{h}_k^r(n)$ and $\underline{g}_k^r(n)$ from neighbors, the combination step of the SD-ACNSAF is performed as

$$h_k(n+1) = \sum_{l \in \Gamma'_k(n)} b_{k,l} \underline{h}_l^r(n+1) \tag{24}$$

$$g_k(n+1) = \sum_{l \in \Gamma'_k(n)} b_{k,l} \underline{g}_l^r(n+1) \tag{25}$$

Finally, the update procedure of the proposed SD-ACNSAF algorithm are summarized in Table 1.

TABLE 1. Summary of the Proposed Algorithm.

```
Initialization:
  $\mathbf{h}_k(0),\ \mathbf{g}_k(0) = 0$ for all $k$
For $n = 1, 2, 3 \cdots$
  At node $k$, $k = 1 : N$
  **Adaptive step**:
  For $l \in \Gamma_k$
    $e_{l,m,D}(n) = d_{l,m,D}(n) - \mathbf{u}^T_{l,m}(n)\mathbf{h}_l(n) - \mathbf{u}^H_{l,m}(n)\mathbf{g}_l(n)$
  end
  $\underline{\mathbf{h}}_k(n+1) = \mathbf{h}_k(n) + \mu_k \sum_{m=1}^{N_s} \frac{e_{k,m,D}(n)}{\|\mathbf{u}_{k,m}(n)\|^2} \mathbf{u}^*_{k,m}(n)$
  $\underline{\mathbf{g}}_k(n+1) = \mathbf{g}_k(n) + \mu_k \sum_{m=1}^{N_s} \frac{e_{k,m,D}(n)}{\|\mathbf{u}_{k,m}(n)\|^2} \mathbf{u}_{k,m}(n)$
  **Detection step**:
  $\mathbf{h}_k(n+1) = \mathbf{h}_k(n) + \mu_k \sum_{m=1}^{N_s} \frac{e_{k,m,D}(n)}{\|\mathbf{u}_{k,m}(n)\|^2} \mathbf{u}^*_{k,m}(n)$
  $\mathbf{g}_k(n+1) = \mathbf{g}_k(n) + \mu_k \sum_{m=1}^{N_s} \frac{e_{k,m,D}(n)}{\|\mathbf{u}_{k,m}(n)\|^2} \mathbf{u}_{k,m}(n)$
  Find the trustworthy neighbors $\Gamma'_k(n)$ by using (22) and (23)
  **Combination step**:
  $\mathbf{h}_k(n+1) = \sum_{l \in \Gamma'_k(n)} b_{k,l} \underline{\mathbf{h}}^r_l(n+1)$
  $\mathbf{g}_k(n+1) = \sum_{l \in \Gamma'_k(n)} b_{k,l} \underline{\mathbf{g}}^r_l(n+1)$
end
```

Remark2: There are two main reasons to affect the performance of the network in an adversarial network environment. One is the false alarm; the other is the missing detection. If the false alarm happens, $\varepsilon_k(n)$ is reduced the performance of distributed estimation has also been reduced to a certain extent, but this does not cause the propagation of malicious data. On the other hand, if happens, the estimates of the whole network are all damaged by the improper data fusion due to missing the detection of compromised nodes or communications. Therefore, in order to reduce the probability of occurrence of the above situation, the thresholds $\gamma_k(n)$ and $\varepsilon_k(n)$ are all initialized to 0.

IV. THE PERFORMANCE ANALYSIS OF PROPOSED ALGORITHM

In this section, the stability performance of the proposed SD-ACNSAF algorithm is analyzed in detail. For analyzing manageable, the following assumptions are adopted.

A4: The step-size is statistically independent of the weight and input vectors [8].

A5: The fluctuations of $\|u_{k,m}(n)\|^2$ can be neglected for the long length of filter [33], [34].

These assumptions have been widely used in the analysis of diffusion algorithms and are very useful in many practical applications [35].

According to (17) and (18), a reliable neighbor l of node k should satisfy the following conditions

$$|\underline{h}_l^{r(j_3)}(n) - \overline{w}_k^{(j_3)}(n)| < \sqrt{\gamma_k(n)} \tag{26}$$

$$|\underline{g}_l^{r(j_4)}(n) - \overline{w}_k^{'(j_4)}(n)| < \sqrt{\gamma_k(n)} \tag{27}$$

The inequality (26) and (27) suggest that the estimators $\underline{h}_l^r(n)$ and $\underline{g}_l^r(n)$ should oscillate around $\overline{w}_k(n)$ and $\overline{w}'_k(n)$, so the (26) and (27) can be written in another form

$$\overline{w}_k(n) - \sqrt{\gamma_k(n)} 1^T \leq \underline{h}_l^r(n) \leq \overline{w}_k(n) + \sqrt{\gamma_k(n)} 1^T \tag{28}$$

$$\overline{w}_k'(n) - \sqrt{\gamma_k(n)}1^T \leq \underline{g}_l^r(n) \leq \overline{w}_k'(n) + \sqrt{\gamma_k(n)}1^T \tag{29}$$

Then, $\underline{h}_l^r(n)$ and $\underline{g}_l^r(n)$ is derived in the following form

$$\underline{h}_l^r(n) = \overline{w}_k(n) + \boldsymbol{\theta}_{k,l}(n) \tag{30}$$

$$\underline{g}_l^r(n) = \overline{w}_k'(n) + \boldsymbol{\theta}_{k,l}'(n) \tag{31}$$

where $\boldsymbol{\theta}_{k,l}(n) = [\boldsymbol{\theta}_{k,l}^{(1)}(n), \cdots, \boldsymbol{\theta}_{k,l}^{(M)}(n)]^T$ denotes the difference between $\underline{h}_l^r(n)$ and $\overline{w}_k(n)$, $\boldsymbol{\theta}_{k,l}'(n) = [\boldsymbol{\theta}_{k,l}^{'(1)}(n), \cdots, \boldsymbol{\theta}_{k,l}^{'(M)}(n)]^T$ denotes the difference between $\underline{g}_l^r(n)$ and $\overline{w}_k'(n)$. $\boldsymbol{\theta}_{k,l}(n)$ and $\boldsymbol{\theta}_{k,l}'(n)$ are the bounded M-dimensional random vector with each component $\boldsymbol{\theta}_{k,l}^{(j)}(n) < \sqrt{\gamma_k(n)}$ and $\boldsymbol{\theta}_{k,l}^{'(j)}(n) < \sqrt{\gamma_k(n)}$.

Because reliable neighbor for each node k cannot be detected with complete accuracy, the set $\Gamma_k(n)$ is divided into two disjoint sets

$$\Gamma_k^+(n) = \Gamma_k(n) \cap \Gamma_k^o(n) \; and \; \Gamma_k^-(n) = \Gamma_k(n) / \, \Gamma_k^+(n) \tag{32}$$

where $\Gamma_k^+(n)$ is the set of neighbors that are detected to be reliable and $\Gamma_k^-(n)$ is the attacked neighbors that are misdetected to be reliable, $\Gamma_k^o(n)$ denotes the true set of normal neighbors of node k.

Then, (4) and (5) can be rewritten as

$$h_k(n) = \sum_{l \in \Gamma_k^+(n)} b_{k,l} \underline{h}_l(n) + \sum_{l \in \Gamma_k^-(n)} b_{k,l}(\overline{w}_k(n) + \boldsymbol{\theta}_{k,l}(n)) \tag{33}$$

$$g_k(n) = \sum_{l \in \Gamma_k^+(n)} b_{k,l} \underline{g}_l(n) + \sum_{l \in \Gamma_k^-(n)} b_{k,l}(\overline{w}_k^{'}(n) + \boldsymbol{\theta}_{k,l}^{'}(n)) \tag{34}$$

Substituting (33) and (34) into (2) and (3), and then subtracting the optimum weights h_o and g_o on both sides, respectively. We have

$$\begin{aligned}
\tilde{h}_k(n+1) &= \sum_{l \in \Gamma_k^+(n)} b_{k,l} \left(\tilde{h}_l(n) + \mu_l \sum_{m=1}^{N_s} \frac{e_{l,m,D}(n)}{\|u_{l,m}(n)\|^2} u_{l,m}^*(n) \right) \\
&\quad + \sum_{l \in \Gamma_k^-(n)} b_{k,l}(\tilde{\overline{w}}_k(n+1) + \boldsymbol{\theta}_{k,l}(n+1)) \\
&= \sum_{l \in \Gamma_k^+(n)} b_{k,l} \left(\tilde{h}_l(n) + \mu_l \sum_{m=1}^{N_s} \frac{u_{l,m}^*(n) c_l(n)}{\|u_{l,m}(n)\|^2} \right) \\
&\quad + \sum_{l \in \Gamma_k^+(n)} b_{k,l} \left(\mu_l \sum_{m=1}^{N_s} \frac{u_{l,m}^*(n) \upsilon_{l,m,D}(n)}{\|u_{l,m}(n)\|^2} \right) \\
&\quad + \sum_{l \in \Gamma_k^-(n)} b_{k,l}(\tilde{\overline{w}}_k(n+1) + \boldsymbol{\theta}_{k,l}(n+1))
\end{aligned} \tag{35}$$

$$\begin{aligned}
\tilde{g}_k(n+1) &= \sum_{l \in \Gamma_k^+(n)} b_{k,l} \left(\tilde{g}_l(n) + \mu_l \sum_{m=1}^{N_s} \frac{e_{l,m,D}(n)}{\|u_{l,m}(n)\|^2} u_{l,m}(n) \right) \\
&\quad + \sum_{l \in \Gamma_k^-(n)} b_{k,l}(\tilde{\overline{w}}_k'(n+1) + \boldsymbol{\theta}_{k,l}'(n+1)) \\
&= \sum_{l \in \Gamma_k^+(n)} b_{k,l} \left(\tilde{g}_l(n) + \mu_l \sum_{m=1}^{N_s} \frac{u_{l,m}(n) c_l(n)}{\|u_{l,m}(n)\|^2} \right) \\
&\quad + \sum_{l \in \Gamma_k^+(n)} b_{k,l} \left(\mu_l \sum_{m=1}^{N_s} \frac{u_{l,m}(n) \upsilon_{l,m,D}(n)}{\|u_{l,m}(n)\|^2} \right) \\
&\quad + \sum_{l \in \Gamma_k^-(n)} b_{k,l}(\tilde{\overline{w}}_k'(n+1) + \boldsymbol{\theta}_{k,l}'(n+1))
\end{aligned} \tag{36}$$

where $c_l(n) = u_{l,m}^T(n)\tilde{h}_l(n) + u_{l,m}^H(n)\tilde{g}_l(n)$, $\tilde{h}_l(n) = h_l(n) - h_o$, $\tilde{g}_l(n) = g_l(n) - g_o$, $\tilde{\overline{w}}_l(n) = \overline{w}_l(n) - h_o$, and $\tilde{\overline{w}}_l'(n) = \overline{w}_l'(n) - g_o$ are the weight deviations at node l.

A. DATA MODEL

To generate the performance analysis of the whole network, some global quantities are defined as follows

$$\begin{aligned}
h(n) &= \operatorname{col}[h_1(n), h_2(n), \cdots, h_N(n)] \\
g(n) &= \operatorname{col}[g_1(n), g_2(n), \cdots, g_N(n)] \\
\underline{h}(n) &= \operatorname{col}\left[\underline{h}_1(n), \underline{h}_2(n), \cdots, \underline{h}_N(n)\right] \\
\underline{g}(n) &= \operatorname{col}\left[\underline{g}_1(n), \underline{g}_2(n), \cdots, \underline{g}_N(n)\right] \\
D &= \operatorname{diag}[\mu_1 I_M, \mu_2 I_M, \cdots, \mu_N I_M] \\
U(n) &= \operatorname{diag}[U_1(n), U_2(n), \cdots, U_N(n)] \\
U_k(n) &= [u_{k,1}(n), u_{k,2}(n), \cdots, u_{k,N_s}(n)] \\
d(n) &= \operatorname{col}[d_1(n), d_2(n), \cdots, d_N(n)] \\
d_k(n) &= \left[d_{k,1}(n), d_{k,2}(n), \cdots, d_{k,N_s}(n)\right] \\
\boldsymbol{\upsilon}(n) &= \operatorname{col}[\boldsymbol{\upsilon}_1(n), \boldsymbol{\upsilon}_2(n), \cdots, \boldsymbol{\upsilon}_N(n)] \\
\boldsymbol{\upsilon}_k(n) &= [\upsilon_{k,1}(n), \upsilon_{k,2}(n), \cdots, \upsilon_{k,N_s}(n)] \\
\boldsymbol{\theta}(n) &= \operatorname{col}[\boldsymbol{\theta}_1(n), \boldsymbol{\theta}_2(n), \cdots, \boldsymbol{\theta}_N(n)] \\
\boldsymbol{\theta}_k(n) &= \operatorname{col}\left[\boldsymbol{\theta}_{k,1}(n), \boldsymbol{\theta}_{k,2}(n), \cdots, \boldsymbol{\theta}_{k,N}(n)\right] \\
\boldsymbol{\theta}'(n) &= \operatorname{col}\left[\boldsymbol{\theta}_1'(n), \boldsymbol{\theta}_2'(n), \cdots, \boldsymbol{\theta}_N'(n)\right] \\
\boldsymbol{\theta}_k'(n) &= \operatorname{col}\left[\boldsymbol{\theta}_{k,1}'(n), \boldsymbol{\theta}_{k,2}'(n), \cdots, \boldsymbol{\theta}_{k,N}'(n)\right] \\
\overline{w}(n) &= \operatorname{col}[\overline{w}_1(n), \overline{w}_2(n), \cdots, \overline{w}_N(n)] \\
\overline{w}'(n) &= \operatorname{col}\left[\overline{w}_1'(n), \overline{w}_2'(n), \cdots, \overline{w}_N'(n)\right]
\end{aligned}$$

Two combination matrices $B^+(n)$ and $B^-(n)$ are introduced, where the kl-th element is defined as follows:

$$\left[B^+(n)\right]_{k,l} = \begin{cases} b_{k,l}(n), & if \; l \in \Gamma_k^+(n) \\ 0, & otherwise \end{cases} \tag{37}$$

and

$$\left[B^-(n)\right]_{k,l} = \begin{cases} b_{k,l}(n), & if \; l \in \Gamma_k^-(n) \\ 0, & otherwise \end{cases} \tag{38}$$

Besides, the extended matrices are given as

$$A^+(n) = B^+(n) \otimes I_M \tag{39}$$

$$A^-(n) = B^-(n) \otimes I_M \tag{40}$$

where $A^+(n) + A^-(n) = A(n)$, $\|A^+(n)\|$, $\|A^+(n)\| \leq 1$.

B. MEAN PERFORMANCE

According to the ahead definitions, (35) and (36) can be rewritten as

$$\begin{aligned}\tilde{h}(n+1) &= A^{+}(n)\tilde{h}(n) - DA^{+}(n)U^{*}(n)\Lambda(n)\\ &\quad\left(U^{T}(n)\tilde{h}(n) + U^{H}(n)\tilde{g}(n) + \boldsymbol{\upsilon}(n)\right)\\ &\quad + A^{-}(n)\tilde{\bar{w}}(n) + A^{-}(n)\boldsymbol{\theta}(n) \qquad (41)\\ \tilde{g}(n+1) &= A^{+}(n)\tilde{g}(n) - DA^{+}(n)U(n)\Lambda(n)\\ &\quad\left(U^{T}(n)\tilde{h}(n) + U^{H}(n)\tilde{g}(n) + \boldsymbol{\upsilon}(n)\right)\\ &\quad + A^{-}(n)\tilde{\bar{w}}'(n) + A^{-}(n)\boldsymbol{\theta}'(n) \qquad (42)\end{aligned}$$

where $\Lambda(n) = (U^{H}(n)U(n))^{-1}$, $\tilde{h}(n) = [\tilde{h}_1(n), \cdots, \tilde{h}_N(n)]^T$, $\tilde{g}(n) = [\tilde{g}_1(n), \cdots, \tilde{g}_N(n)]^T$, $\tilde{\bar{w}}(n) = [\tilde{\bar{w}}_1(n), \cdots, \tilde{\bar{w}}_N(n)]^T$ and $\tilde{\bar{w}}'(n) = [\tilde{\bar{w}}'_1(n), \cdots, \tilde{\bar{w}}'_N(n)]^T$.

Then, the weight deviation update is combined as

$$\begin{aligned}\begin{bmatrix}\tilde{h}(n+1)\\ \tilde{g}(n+1)\end{bmatrix} &= A^{+}(n)\begin{bmatrix} I - DU^{*}(n)\Lambda(n)U^{T}(n) & -DU^{*}(n)\Lambda(n)U^{H}(n)\\ -DU(n)\Lambda(n)U^{T}(n) & I - DU(n)\Lambda(n)U^{H}(n)\end{bmatrix}\begin{bmatrix}\tilde{h}(n)\\ \tilde{g}(n)\end{bmatrix}\\ &\quad - DA^{+}(n)\begin{bmatrix}U^{*}(n)\\ U(n)\end{bmatrix}\Lambda(n)\boldsymbol{\upsilon}(n)\\ &\quad + A^{-}(n)\begin{bmatrix}\tilde{\bar{w}}(n)\\ \tilde{\bar{w}}'(n)\end{bmatrix} + A^{-}(n)\begin{bmatrix}\boldsymbol{\theta}(n)\\ \boldsymbol{\theta}'(n)\end{bmatrix} \qquad (43)\end{aligned}$$

and (43) can be rewritten as

$$\begin{aligned}\tilde{w}(n+1) &= A^{a+}(n)\left(I_{2MN} - D^{a}U^{a*}(n)\Lambda(n)U^{aT}(n)\right)\tilde{w}(n)\\ &\quad - D^{a}A^{a+}(n)U^{a*}(n)\Lambda(n)\boldsymbol{\upsilon}(n)\\ &\quad + A^{a-}(n)\tilde{\bar{w}}^{a}(n) + A^{a-}(n)\boldsymbol{\theta}^{a}(n) \qquad (44)\end{aligned}$$

where $\tilde{w}(n) = [\tilde{h}^T(n), \tilde{g}^T(n)]^T$, $U^a(n) = [U^T(n), U^H(n)]^T$, $\tilde{\bar{w}}^{a}(n) = [\tilde{\bar{w}}^{T}(n), \tilde{\bar{w}}^{'T}(n)]^T$, $\boldsymbol{\theta}^{a}(n) = [\boldsymbol{\theta}^{T}(n), \boldsymbol{\theta}^{'T}(n)]^T$, $A^{a+}(n) = \mathrm{diag}[A^{+}(n), A^{+}(n)]$, $A^{a-}(n) = \mathrm{diag}[A^{-}(n), A^{-}(n)]$ and $D^a = \mathrm{diag}[D, D]$.

Under the *A3*, *A4* and *A5*, taking the expectation operator on both sides of the above equation, then

$$\begin{aligned}&E\left[\tilde{w}(n+1)\right]\\ &= E[A^{a+}(n)]\left\{I_{2MN} - D^{a}E\left[U^{a*}(n)\Lambda(n)U^{aT}(n)\right]\right\}\\ &\quad E[\tilde{w}(n)] + E\left[A^{a-}(n)\tilde{\bar{w}}^{a}(n)\right]\\ &\quad + E\left[A^{a-}(n)\boldsymbol{\theta}^{a}(n)\right] \qquad (45)\end{aligned}$$

Theorem 1: (Mean stability) Under the assumptions *A1*-*A5*, the estimator $w(n)$ asymptotically converges in mean sense if the step-size satisfies condition (76).

Proof: According to the definitions of $\tilde{\bar{w}}^{a}(n)$ and $\boldsymbol{\theta}^{a}(n)$, because the reference estimates $\bar{w}_k(n)$ and $\bar{w}'_k(n)$ converge to unbiased estimates of optimum weights, it's obvious that $E[\tilde{\bar{w}}^{a}(n)] = 0$ and $\|A^{a-}(n)\| \leq 1$. So we can conclude that $E\left[A^{a-}(n)\tilde{\bar{w}}^{a}(n)\right] = 0$ as $n \to \infty$.

Because $\boldsymbol{\theta}^{a}(n)$ is composed of $\boldsymbol{\theta}_{k,l}(n)$ and $\boldsymbol{\theta}_{k,l}(n)$, $\boldsymbol{\theta}^{(j)}_{k,l}(n) < \sqrt{\gamma_k(n)}$ and $\boldsymbol{\theta}^{'(j)}_{k,l}(n) < \sqrt{\gamma_k(n)}$. The $A^{a-}(n)\boldsymbol{\theta}^{a}(n)$ is bounded vector.

Since $\|A^{a+}(n)\| \leq 1$, we have

$$\begin{aligned}&|\lambda_{\max}(E[A^{a+}(n)](I_{2MN} - D^{a}E\left[U^{a*}(n)\Lambda(n)U^{aT}(n)\right]))|\\ &\quad < |\lambda_{\max}(I_{2MN} - D^{a}E\left[U^{a*}(n)\Lambda(n)U^{aT}(n)\right])| \qquad (46)\end{aligned}$$

As shown in appendix, for each node k, if the step-size satisfies (76), we have $|\lambda_{\max}(I_M - \mu_k E\left[U^{*}(n)\Lambda(n)U^{T}(n)\right])| < 1$, and thus, $|\lambda_{\max}(I_{2MN} - D^{a}E\left[U^{a*}(n)\Lambda(n)U^{aT}(n)\right])| < 1$. In other words, if μ_k satisfies (76), the first term in the RHS of (45) converges, no matter what $A^{+}(n)$ is. We can conclude that $w(n) = [h^T(n), g^T(n)]^T$ converges to a biased estimate of the optimum weight w^o with the bias governed by the step-size μ_k and combination matrix $A^{-}(n)$. In a particular case, when there is no missing detection of attacks, $A^{a-}(n)$ converges to an zero matrix, and thus $w(n)$ converges to an unbiased estimate of w^o.

Remark3: As can be seen from (45), when the detection of attacked sensors or communications is missing, the attacked data exist, and the third term in the RHS of (45) will not converge to zero. Hence, a biased estimate results in. Therefore, the mean stability of the proposed secure algorithm is significantly dependent on the error of missing detection.

C. MEAN SQUARE PERFORMANCE

The mean-square performance analysis is given by following the energy conservation framework [36]. First, Defining the squared weighted Euclidean norm of a complex-valued vector x, $\|x\|^2_\Sigma = x^H\Sigma x$, where Σ is any symmetric nonnegative definite matrix with the $2MN \times 2MN$ dimension. And taking the weighted Euclidean norm on both sides of (44) yields

$$\begin{aligned}\|\tilde{w}(n+1)\|^2_\Sigma &= \|\tilde{w}(n)\|^2_{\Sigma'} + \boldsymbol{\upsilon}^H(n)Y(n)\boldsymbol{\upsilon}(n)\\ &\quad + \tilde{\bar{w}}^{aH}(n)A^{a-H}(n)\Sigma A^{a-}(n)\tilde{\bar{w}}^{a}(n)\\ &\quad + \boldsymbol{\theta}^{aH}(n)A^{a-H}(n)\Sigma A^{a-}(n)\boldsymbol{\theta}^{a}(n)\\ &\quad + \tilde{\bar{w}}^{aH}(n)A^{a-H}(n)\Sigma A^{a-}(n)\boldsymbol{\theta}^{a}(n)\\ &\quad + \boldsymbol{\theta}^{aH}(n)A^{a-H}(n)\Sigma A^{a-}(n)\tilde{\bar{w}}^{a}(n)\\ &\quad + \left\{\textit{Cross term which are involving } \boldsymbol{\upsilon}(n)\right\} \qquad (47)\end{aligned}$$

where

$$\begin{aligned}\Sigma' &= (I_{2MN} - D^{a}U^{a*}(n)\Lambda(n)U^{aT}(n))^{H}A^{a+H}(n)\Sigma\\ &\quad A^{a+}(n)(I_{2MN} - D^{a}U^{a*}(n)\Lambda(n)U^{aT}(n)) \qquad (48)\\ Y(n) &= \Lambda^{H}(n)U^{aT}(n)A^{a+H}(n)D^{aH}\Sigma D^{a}A^{a+}(n)U^{a*}(n)\Lambda(n) \qquad (49)\end{aligned}$$

Taking the expectation of (47) results in

$$\begin{aligned}E\left[\|\tilde{w}(n+1)\|^2_\Sigma\right] &= E\left[\|\tilde{w}(n)\|^2_{\Sigma'}\right] + E\left[\boldsymbol{\upsilon}^H(n)Y(n)\boldsymbol{\upsilon}(n)\right]\\ &\quad + E\left[\tilde{\bar{w}}^{aH}(n)A^{a-H}(n)\Sigma A^{a-}(n)\tilde{\bar{w}}^{a}(n)\right]\end{aligned}$$

$$
\begin{aligned}
&+E\left[\boldsymbol{\theta}^{aH}(n)A^{a-H}(n)\Sigma A^{a-}(n)\boldsymbol{\theta}^{a}(n)\right]\\
&+2\Re\Big\{E\Big[\tilde{\overline{w}}^{aH}(n)A^{a-H}(n)\Sigma\\
&A^{a-}(n)\boldsymbol{\theta}^{a}(n)\Big]\Big\}
\end{aligned}
\tag{50}
$$

To calculate (50) tractable, the vectorization operation is introduced. For any matrices $\left\{X, \Sigma, Y\right\}$ of compatible dimensions, the following property of Kronecker product is given [34]

$$
\mathrm{vec}(X, \Sigma, Y) = (Y^T \otimes X)\mathrm{vec}(\Sigma) \tag{51}
$$

By applying (51) into (48), the following vector relation can be derived

$$
\boldsymbol{\sigma}' = \mathrm{vec}(\Sigma') = F\boldsymbol{\sigma} \tag{52}
$$

where

$$
\begin{aligned}
F &= E\Big\{((I_{2MN} - D^a U^{a*}(n)\Lambda(n)U^{aT}(n))^T A^{a+T}(n))\\
&\quad\otimes((I_{2MN} - D^a U^{a*}(n)\Lambda(n)U^{aT}(n))^H A^{a+T}(n))\Big\}\\
&= \Big((I_{2MN} - D^a E\{U^{a*}(n)\Lambda(n)U^{aT}(n)\})^T \otimes\\
&\quad (I_{2MN} - D^a E\{U^{a*}(n)\Lambda(n)U^{aT}(n)\})^H\Big)\\
&\quad (E[A^{a+T}(n)] \otimes E[A^{a+H}(n)])\\
&= \Big(I_{2MN} - I_{2MN} \otimes (E\{U^{a*}(n)\Lambda^H(n)U^{aT}(n)\}D^{aH})\\
&\quad - (E\{U^{a}(n)\Lambda(n)U^{aH}(n)\}D^{a}) \otimes I_{2MN}\\
&\quad + (E\{U^{a}(n)\Lambda(n)U^{aH}(n)\}D^{a})\\
&\quad \otimes(E\{U^{a*}(n)\Lambda^H(n)U^{aT}(n)\}D^{aH})\Big)\\
&\quad (E[A^{a+T}(n)] \otimes E[A^{a+H}(n)])
\end{aligned}
\tag{53}
$$

The second term of the right side of (50) can be written as

$$
\begin{aligned}
E\left\{\boldsymbol{\upsilon}^H(n)Y(n)\boldsymbol{\upsilon}(n)\right\} &= E\left\{\mathrm{Tr}(\boldsymbol{\upsilon}(n)\boldsymbol{\upsilon}^H(n)Y(n))\right\}\\
&= \mathrm{Tr}\Big(E\{\boldsymbol{\upsilon}(n)\boldsymbol{\upsilon}^H(n)Y(n)\}\Big)\\
&= \Big(\mathrm{vec}(E\{\boldsymbol{\upsilon}(n)\boldsymbol{\upsilon}^H(n)\}^T)\Big)^T\\
&\quad \mathrm{vec}(E\{Y(n)\})
\end{aligned}
\tag{54}
$$

Then, using (52) and (54), (50) becomes

$$
\begin{aligned}
&E\left\{\|\tilde{w}(n+1)\|^2_{\mathrm{vec}^{-1}(\boldsymbol{\sigma})}\right\}\\
&\quad = E\left\{\|\tilde{w}(n)\|^2_{\mathrm{vec}^{-1}(F\boldsymbol{\sigma})}\right\} + \boldsymbol{\Upsilon}(n)\boldsymbol{\sigma}\\
&\qquad + E\left[\tilde{\overline{w}}^{aH}(n)A^{a-H}(n)\Sigma A^{a-}(n)\tilde{\overline{w}}^{a}(n)\right]\\
&\qquad + E\left[\boldsymbol{\theta}^{aH}(n)A^{a-H}(n)\Sigma A^{a-}(n)\boldsymbol{\theta}^{a}(n)\right]\\
&\qquad + 2\Re\Big\{E\Big[\tilde{\overline{w}}^{aH}(n)A^{a-H}(n)\Sigma A^{a-}(n)\boldsymbol{\theta}^{a}(n)\Big]\Big\}
\end{aligned}
\tag{55}
$$

where

$$
\begin{aligned}
\boldsymbol{\Upsilon}(n) = &\Big(\mathrm{vec}(E\{\boldsymbol{\upsilon}(n)\boldsymbol{\upsilon}^H(n)\}^T)\Big)^T E\{(\Lambda(n)\otimes\Lambda^H(n))\}\\
&E\{(U^{aH}(n)\otimes U^{aT}(n))\}(E(A^{a+}(n)) \otimes E(A^{a+H}(n)))\\
&(D^a \otimes D^{aH})
\end{aligned}
\tag{56}
$$

Theorem 2: (Mean-square stability) Assume the step-size is sufficiently small, the diffusion strategy (33) and (34) is mean-square stable if the matrix F is stable. The stability condition of F is guaranteed by sufficiently small step-sizes that also satisfy (47).

Proof: Iterating recursion (55) starting from $n = 0$, we find that

$$
\begin{aligned}
&E\left[\|\tilde{w}(n+1)\|^2_{vec^{-1}(\boldsymbol{\sigma})}\right]\\
&\quad = E\left[\|\tilde{w}(0)\|^2_{vec^{-1}(F^{n+1}\sigma)}\right] + \sum_{p=0}^{n}\boldsymbol{\Upsilon}(p)\boldsymbol{\sigma}\\
&\qquad + \sum_{p=0}^{n} E\left[\tilde{\overline{w}}^{aH}(p)A^{a-H}(p)\Sigma A^{a-}(p)\tilde{\overline{w}}^{a}(p)\right]\\
&\qquad + \sum_{p=0}^{n} E\left[\boldsymbol{\theta}^{aH}(p)A^{a-H}(p)\Sigma A^{a-}(p)\boldsymbol{\theta}^{a}(p)\right]\\
&\qquad + 2\sum_{p=0}^{n}\Re\Big\{E\Big[\tilde{\overline{w}}^{aH}(p)A^{a-H}(p)\Sigma A^{a-}(p)\boldsymbol{\theta}^{a}(p)\Big]\Big\}
\end{aligned}
\tag{57}
$$

Provided that F is stable, the first and second terms on the RHS of (57) converge as $n \to \infty$, to zero for the former, and to a finite value for the latter. The third term depends on the cooperation matrix $A^{-}(n)$ and the MSD of reference subsystem. Since $\|A^{-}(n)\|_2 \leq 1, \overline{w}_k(n)$ and $\overline{w}'_k(n)$ converge to the optimum weight and optimum conjugate weight, the third term is bounded. Since $\boldsymbol{\theta}_{k,l}(n)$ and $\boldsymbol{\theta}'_{k,l}(n)$ are the bounded M-dimensional random vector with each component $\boldsymbol{\theta}^{(m)}_{k,l}(n) < \sqrt{\gamma_k(n)}$ and $\boldsymbol{\theta}^{'(m)}_{k,l}(n) < \sqrt{\gamma_k(n)}$, the forth term is bounded by a value govered by the threshold $\gamma_{k,max}(n)$. The fifth term is bounded by the $\overline{w}_k(n)$ and $\overline{w}'_k(n)$ and converges to zero. We conclude that $E\left\{\|\tilde{w}(n+1)\|^2_{\mathrm{vec}^{-1}(\boldsymbol{\sigma})}\right\}$ converges to a bounded value as $n \to \infty$, and the algorithm is said to be mean-square stable. Therefore, the overall MSD converges and is bounded by the step-size.

However, since the cooperation matrixs $A^{-}(n)$ and $A^{+}(n)$ are time-varying and depend on the detecting result, we cannot give an exact constant to denote overall MSD. If no missing detection or the attacks are all detected, that is to say, the $A^{-}(n)$ converges to zero matrix, $A^{+}(n)$ converges to a stable combination matrix $A^{+'}$, and $A^{a+}(n)$ converges to a stable combination matrix $A^{a+'}$. In this special case, the last three terms in (55) converge to zeros, we get the corresponding MSD as follows

$$
E\left\{\|\tilde{w}(n+1)\|^2_{\mathrm{vec}^{-1}(\boldsymbol{\sigma})}\right\} = E\left\{\|\tilde{w}(n)\|^2_{\mathrm{vec}^{-1}(F'\boldsymbol{\sigma})}\right\} + \boldsymbol{\Upsilon}'(n)\boldsymbol{\sigma}
\tag{58}
$$

where

$$
\begin{aligned}
F' = E\Big\{&((I_{2MN} - D^a U^{a*}(n)\Lambda(n)U^{aT}(n))^T A^{a+'T})\\
&\otimes((I_{2MN} - D^a U^{a*}(n)\Lambda(n)U^{aT}(n))^H A^{a+'T})\Big\}
\end{aligned}
$$

$$
\begin{aligned}
&= \Big((I_{2MN} - D^a E\{U^{a*}(n)\Lambda(n)U^{aT}(n)\})^T \otimes \\
&\quad (I_{2MN} - D^a E\{U^{a*}(n)\Lambda(n)U^{aT}(n)\})^H\Big) \\
&\quad (E[A^{a+'T}] \otimes E[A^{a+'H}]) \\
&= \Big(I_{2MN} - I_{2MN} \otimes (E\{U^{a*}(n)\Lambda^H(n)U^{aT}(n)\}D^{aH}) \\
&\quad + (E\{U^a(n)\Lambda(n)U^{aH}(n)\}D^a) \otimes I_{2MN} \\
&\quad + (E\{U^a(n)\Lambda(n)U^{aH}(n)\}D^a) \\
&\quad \otimes (E\{U^{a*}(n)\Lambda^H(n)U^{aT}(n)\}D^{aH})\Big) \\
&\quad (E[A^{a+'T}] \otimes E[A^{a+'H}]) \qquad (59)
\end{aligned}
$$

and

$$
\begin{aligned}
\boldsymbol{\Upsilon}'(n) = &\Big(\mathrm{vec}(E\{\boldsymbol{\upsilon}(n)\boldsymbol{\upsilon}^H(n)\}^T)\Big)^T E\{(\Lambda(n)\otimes\Lambda^H(n))\} \\
&E\{(U^{aH}(n)\otimes U^{aT}(n))\}(E(A^{a+'}) \otimes E(A^{a+'H})) \\
&(D^a \otimes D^{aH}) \qquad (60)
\end{aligned}
$$

(58) has a similar form as the common D-ACNSAF except that the combination weight matrix $A^{a+'}$.

When the proposed algorithm has converged to the steady-state, (58) can be written as

$$
E\left\{\|\tilde{w}(\infty)\|^2_{\mathrm{vec}^{-1}(\boldsymbol{\sigma})}\right\} = E\left\{\|\tilde{w}(\infty)\|^2_{\mathrm{vec}^{-1}(F'\boldsymbol{\sigma})}\right\} + \boldsymbol{\Upsilon}'(n)\boldsymbol{\sigma} \qquad (61)
$$

Then,

$$
E\left\{\|\tilde{w}(\infty)\|^2_{\mathrm{vec}^{-1}((I_{2MN}-F')\boldsymbol{\sigma})}\right\} = \boldsymbol{\Upsilon}'(n)\boldsymbol{\sigma} \qquad (62)
$$

Choosing $\boldsymbol{\sigma} = (I_{2MN} - F')^{-1}$, the sum of steady-state mean square deviation for all the nodes can be given as

$$
MSD(\infty) = \boldsymbol{\Upsilon}'(n)(I_{2MN} - F')^{-1} \qquad (63)
$$

The network MSD is the average of MSD across all the network nodes. Then, the network steady-state MSD can be obtained as follows:

$$
MSD^{network}(\infty) = \frac{1}{N}MSD(\infty) \qquad (64)
$$

As illustrated in (62), when there is no missing detection, the steady-state MSD is determined by the step-size, input signal and the combination matrix $A^{+'}(n)$.

According to the above analysis, for small step-sizes, we can conclude that the performance of SD-ACNSAF lies somewhere around the reference subsystem, and does not deviate it much. If the deviation is large, the hypothesis test (22) is not satisfied, information is considered to be broken, and is not used in combination step. The impact of the number of attacks on the performance of the algorithm will be discussed in the section V.

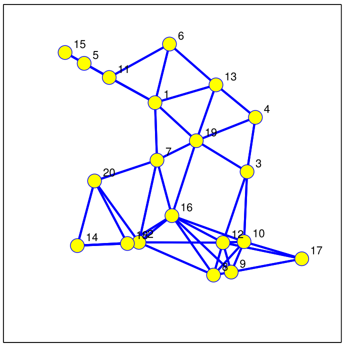
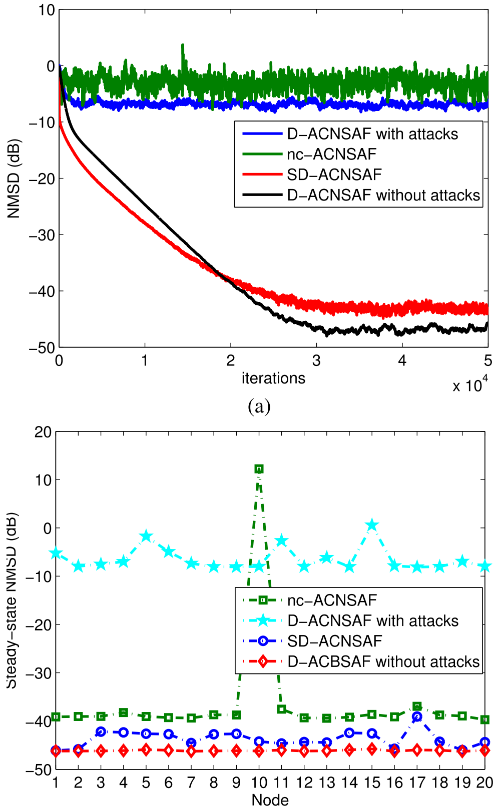


FIGURE 3. Network topology with 20 nodes.



(a)

(b)

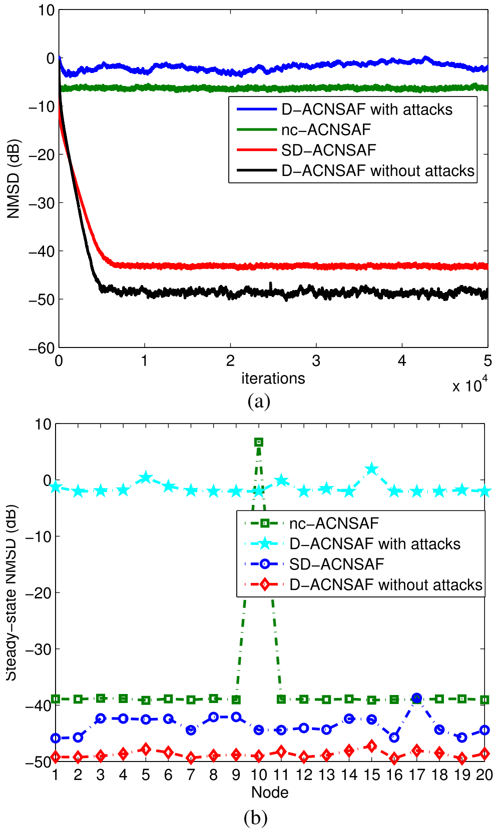
FIGURE 4. Performance comparisons of different algorithms for noncircular ARMA complex-valued signals under circular single point attack. (a) Transient network MSD. (b) Steady-state network MSD verus node.

D. THE SELECTION OF THRESHOLD $\gamma_k(n)$

It's obvious that the selection of threshold will have a great impact on the performance of the proposed SD-ACNSAF algorithm. The threshold selection method will be given in the following section.

Substituting (30) into the combination step (24),

$$
\begin{aligned}
h_k(n) &= \sum_{l\in\Gamma'_k(n)} b_{k,l}\left(\overline{w}_k(n) + \boldsymbol{\theta}_{k,l}(n)\right) \\
&= \overline{w}_k(n) + \sum_{l\in\Gamma'_k(n)} b_{k,l}\boldsymbol{\theta}_{k,l}(n) \\
&= \overline{w}_k(n) + B(n)\boldsymbol{\theta}_k(n) \qquad (65)
\end{aligned}
$$



(a)

(b)

FIGURE 5. Performance comparisons of different algorithms for noncircular Ikeda map complex-valued signals under circular single point attack. (a) Transient network MSD. (b) Steady-state network MSD verus node.

Subtracting the optimum weight h_o and obtaining the squared deviation of (65)

$$\|\tilde{h}_k(n)\|^2 = \|\tilde{\bar{w}}_k(n)\|^2 + \|B(n)\boldsymbol{\theta}_k(n)\|^2 + \tilde{\bar{w}}_k^H(n)B(n)\boldsymbol{\theta}_k(n) + \boldsymbol{\theta}_k^H(n)B(n)\tilde{\bar{w}}_k(n) \quad (66)$$

Then,

$$\|B(n)\boldsymbol{\theta}_k(n)\|^2 \leq 2\Re\left\{\tilde{\bar{w}}_k^H(n)B(n)\boldsymbol{\theta}_k(n)\right\} \leq \|\tilde{\bar{w}}_k^H(n)B(n)\boldsymbol{\theta}_k(n)\| \quad (67)$$

A sufficient condition for ensuring (66) can be obtained

$$|\sum_{l\in\Gamma_k'(n)} b_{k,l}\boldsymbol{\theta}_{k,l}^{(m)}(n)| < 2|\tilde{\bar{w}}_k^{(m)}(n)| \quad (68)$$

Which is equivalent to

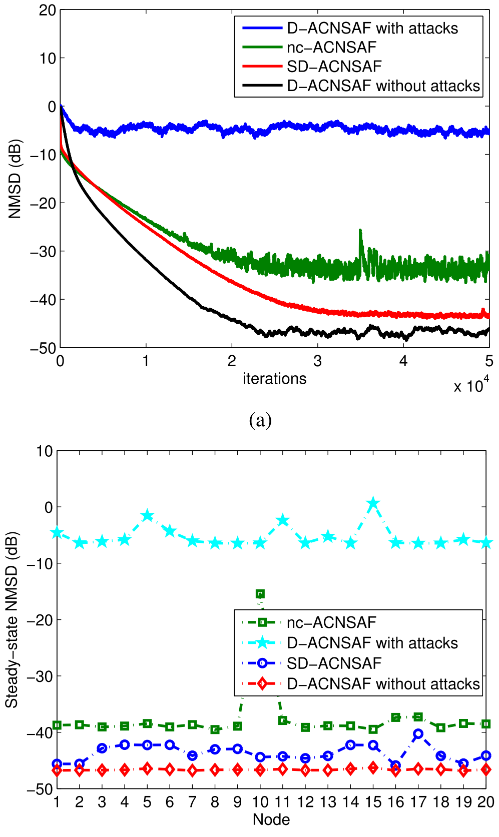
$$|\sum_{l\in\Gamma_k'(n)} b_{k,l}\boldsymbol{\theta}_{k,l}^{(m)}(n)| < \sqrt{\beta_k(n)} \quad (69)$$

Combining (68) and (69)

$$\beta_k(n) < 4\|\tilde{\bar{w}}_k^{(m)}(n)\|^2 \quad (70)$$

Similarly, another threshold is derived

$$\beta_k'(n) < 4\|\tilde{\bar{w}}_k^{\prime(m)}(n)\|^2 \quad (71)$$



(a)

(b)

FIGURE 6. Performance comparisons of different algorithms for noncircular ARMA complex-valued signals under noncircular single point attack. (a) Transient network MSD. (b) Steady-state network MSD verus node.

As a result, the threshold $\gamma_k(n)$ can be derived as follows

$$\gamma_k(n) = \min(\beta_k(n), \beta_k'(n)) \quad (72)$$

Since the optimum weights h_o and g_o are unknown in practical applications, $\|\tilde{\bar{w}}_k^{(m)}(n)\|^2$ and $\|\tilde{\bar{w}}_k^{\prime(m)}(n)\|^2$ can be obtained by employing the time average method as follows

$$\begin{aligned}
\hat{\boldsymbol{\mu}}_{k1}(n+1) &= [n\hat{\boldsymbol{\mu}}_{k1}(n) + \bar{w}_k(n)]/(n+1)\\
\hat{\boldsymbol{\sigma}}_{k1}(n+1) &= [(n)\hat{\boldsymbol{\sigma}}_{k1}(n) + \|\bar{w}_k(n) - \hat{\boldsymbol{\mu}}_{k1}(n)\|^2/M]/(n+1)\\
\hat{\boldsymbol{\mu}}_{k2}(n+1) &= [n\hat{\boldsymbol{\mu}}_{k2}(n) + \bar{w}_k'(n)]/(n+1)\\
\hat{\boldsymbol{\sigma}}_{k2}(n+1) &= [(n)\hat{\boldsymbol{\sigma}}_{k2}(n) + \|\bar{w}_k'(n) - \hat{\boldsymbol{\mu}}_{k2}(n)\|^2/M]/(n+1)
\end{aligned}$$

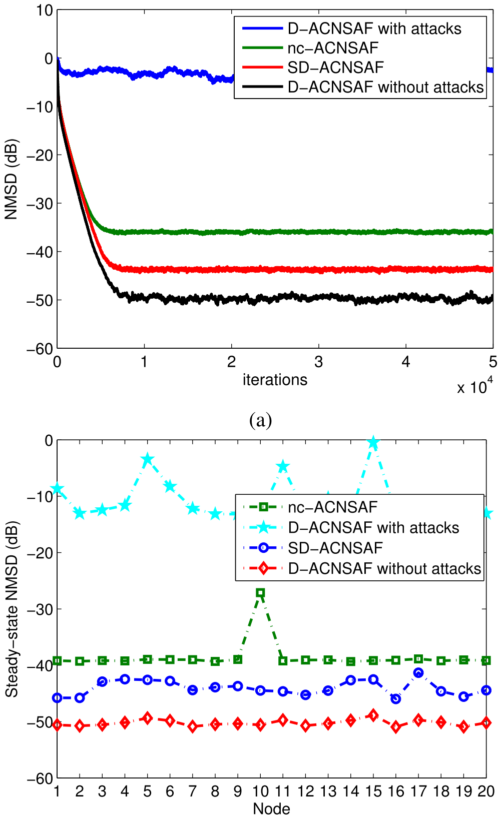
where $\hat{\boldsymbol{\mu}}_{k1}(1)$ and $\hat{\boldsymbol{\mu}}_{k2}(1)$ are all initialized as zero vectors. $\hat{\boldsymbol{\sigma}}_{k1}(1)$ and $\hat{\boldsymbol{\sigma}}_{k2}(1)$ are all equal to 0.

Then, the threshold is given as

$$\gamma_k(n) = \min(\hat{\boldsymbol{\sigma}}_{k1}(n), \hat{\boldsymbol{\sigma}}_{k2}(n)) \quad (73)$$

V. SIMULATION RESULTS

In this section, several numerical simulations have been done to examine the effectiveness of the proposed algorithm. The unknown channel is a widely linear moving average (WL-MA) process and is randomly generated as unit magnitude in the form of complex values. The cosine modulated



(a)

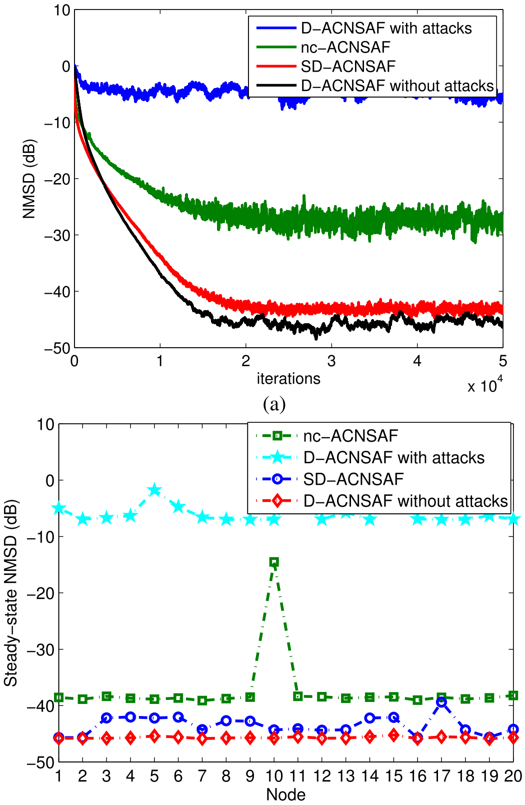
(b)

FIGURE 7. **Performance comparisons of different algorithms for noncircular Ikeda map complex-valued signals under noncircular single point attack. (a) Transient network MSD. (b) Steady-state network MSD verus node.**

filter banks are used as the synthesis and analysis filters. The number of subband N_s is set to 4 and the length of prototype filter is 32. The Metropolis rule is used for the combination weights. The network profile with 20 nodes is used shown in Fig. 3. There are 39 links between nodes in the network. The communication between node 5 and node 15 is compromised. The attack measurement $q_k(n)$ for compromised sensor is a noncircular complex-valued doubly white Gaussian ($E\{q_k^2(n)\} = 0.9$) with $q \sim N(0, 1)+iN(0, 1)$. The disturbance vectors $z^a_{k,l}(n)$ and $z^{a'}_{k,l}(n)$ for compromised communications are generated from zero-mean complex-valued doubly Gaussian distribution ($E\{z^{a^2}(n)\} = 0.9$) with $z^a(n) \sim N(0, 0.5)+iN(0, 0.5)$. The complex-valued network mean-square deviation (NMSD) is used as the evaluation criteria, which is defined as:

$$NMSD(n) = \frac{1}{N}\sum_{k=1}^{N} E\big\{\|h_o - h_k(n)\|^2 + \|g_o - g_k(n)\|^2\big\} \tag{74}$$

The simulation results are achieved by averaging 20 iterations and perform on benchmark complex-valued noncircular input signals.



(a)

(b)

FIGURE 8. **Performance comparisons of different algorithms for noncircular ARMA complex-valued signals under time-varying noncircular single point attack. (a) Transient network MSD. (b) Steady-state network MSD verus node.**

One of the noncircular complex-valued signals is an autoregressive moving average (ARMA) complex process, which is given as follows [37]

$$\begin{aligned} u_k(n) = 0.85u_k(n-1) + 2o(n) + 0.45o^*(n) \\ + o(n-1) + 0.9o^*(n-1) \end{aligned} \tag{75}$$

where $o(n)$ is a complex-valued doubly second order noncircular white Gaussian process with $\sigma_o^2 = 1$ and $\tilde{\sigma}_o^2 = 0.8$.

Another benchmark noncircular signal is the Ikeda signal (nonlinear and with coupled states), given by [38]

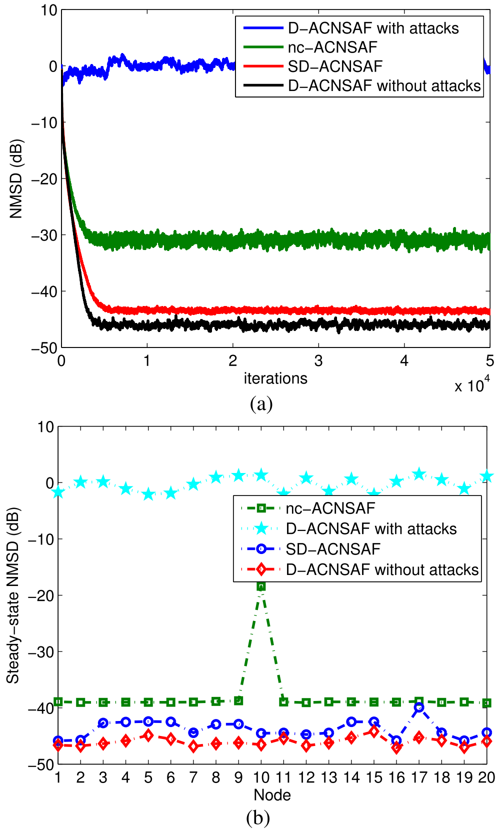
$$\begin{aligned} u(n) &= 1 + \alpha(u(n-1)\cos(t(n-1))) \\ &\quad - v(n-1)\sin(t(n-1)) \end{aligned} \tag{76}$$

$$\begin{aligned} v(n) &= \alpha(u(n-1)\sin(t(n-1))) \\ &\quad + v(n-1)\cos(t(n-1)) \end{aligned} \tag{77}$$

where $\alpha = 0.9$ and $t(n-1) = 0.4 - 6/(1+u^2(n-1)+v^2(n-1))$. $u(n)$ and $v(n)$ are the real and imaginary parts of $u_k(n)$.

A. SINGLE POINT ATTACK

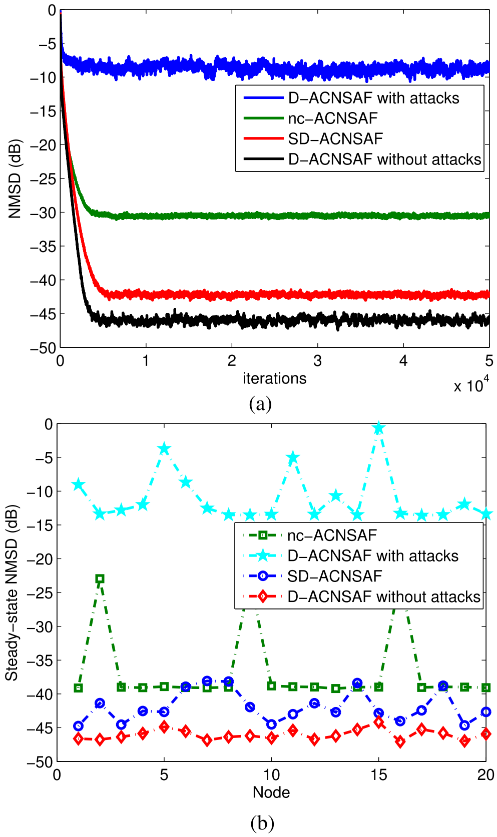
First, to verify the anti-circular attack effect of proposed algorithm, we present the comparisons of the performance of the proposed SD-ACNSAF algorithm, nc-ACNSAF and the



(a)

(b)

FIGURE 9. Performance comparisons of different algorithms for noncircular Ikeda map complex-valued signals under time-varying noncircular single point attack. (a) Transient network MSD. (b) Steady-state network MSD verus node.



(a)

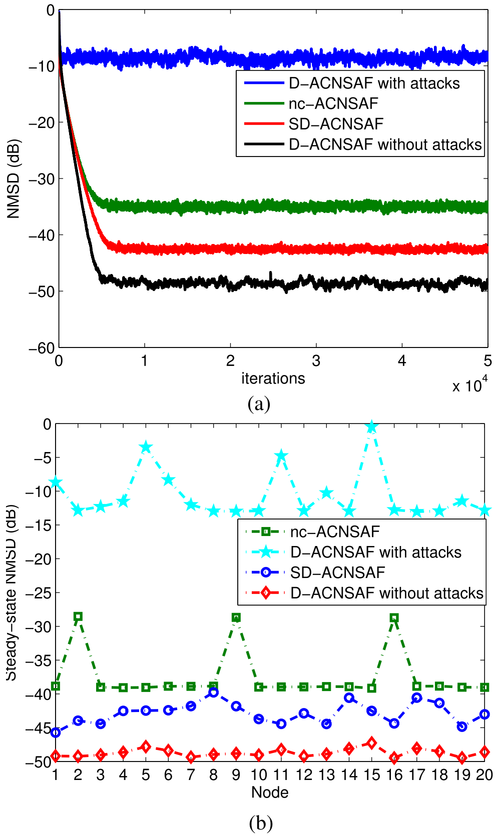
(b)

FIGURE 10. Performance comparisons of different algorithms for noncircular Ikeda map complex-valued signals under noncircular multipoint attacks. (a) Transient network MSD. (b) Steady-state network MSD verus node.

original D-ACNSAF with and without attacks for noncircular complex-valued input signals under circular single point attack. The learning curves of these algorithms are shown in Fig. 4 and 5. The compromised sensor is chosen as 10. The length of unknown channel is set as 4. The step-size μ_k is chosen as 0.00145. The background noise of each node is a circular complex-valued doubly white Gaussian, and the variances are within (0, 0.1). As can be seen from Figs. 4(a) and 5(a), since the damaged information is also used by other adjacent nodes, the performance of D-ACNSAF decreases most. The performance of nc-ACNSAF algorithm is a litter better than the D-ACNSAF with attacks because the damaged information is not exchanged. Due to the detection scheme, the proposed SD-ACNSAF behaves a similar performance with the D-ACNSAF without attacks in an adverse environment. From Fig. 4(b) and 5(b), the simulation results further validate our theoretical analysis. Because the network MSD of the compromised node is extremely large, the averaged performance of the network is poor. The performance of nc-ACNSAF is superior to D-ACNSAF with attacks, but inferior to our proposed algorithm. The proposed SD-ACNSAF algorithm shows low steady-state NMSD and closes to the theoretical value without attacks due to making a detection of reliable neighbors at first and then performing data fusion on the reliable neighbors. The small difference between them is due to the removal of the measurements from the unaffected nodes that are detected as being attacked.

Then, in order to test the anti-noncircular attack effect of proposed algorithm, the comparisons of the performance of the proposed SD-ACNSAF algorithm, nc-ACNSAF and the original D-ACNSAF with and without attacks for noncircular complex-valued input signals under noncircular single point attack are presented. The learning curves of these algorithms are shown in Fig. 6 and 7. The compromised sensor is chosen as 10. The length of unknown channel is set as 4. The step-size μ_k is chosen as 0.00245. Since these algorithms are all using the WL model, they have a good capability of processing noncircular complex-valued signals. Besides, the proposed SD-ACNSAF algorithm has a good performance against both the circular and noncircular attacks. Also note the steady-state network MSD of the proposed SD-ACNSAF algorithm is quite close to that of the D-ACNSAF without noncircular attacks.

Afterwards, we compare the proposed SD-ACNSAF algorithm with nc-ACNSAF and the original D-ACNSAF with and without attacks for noncircular complex-valued input signals under time-varying noncircular single point attack. The network transient behaviors and steady-state behaviors in



(a)

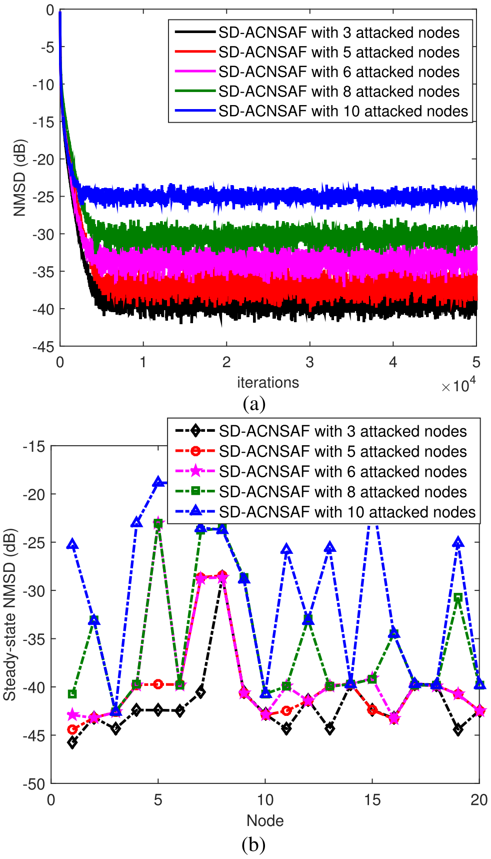
(b)

FIGURE 11. Performance comparisons of different algorithms for noncircular Ikeda map complex-valued signals under time-varying noncircular multipoint attacks. (a) Transient network MSD. (b) Steady-state network MSD verus node.

term of MSD for these algorithms are shown in Fig. 8 and 9. The compromised sensor is chosen as 10. The length of unknown channel is set as 4. The step-size μ_k is chosen as 0.0185. As can be seen from Fig. 8 and 9, the proposed algorithm performs better than the nc-ACNSAF and D-ACNSAF with attacks and has a similar convergence performance with the D-ACNSAF algorithm without attacks in a time-varying noncircular attack environment.

B. MULTIPOINT POINT ATTACK

To further verify the anti-noncircular attack effect of proposed algorithm under time-varying and non-time-varying noncircular multipoint attacks, the comparisons of the performance of the proposed SD-ACNSAF algorithm, nc-ACNSAF and the original D-ACNSAF with and without attacks for noncircular complex-valued Ikeda map input signals are presented. In Fig. 10 and 11, we show the learning curves of these algorithms. The compromised sensors are chosen as 2, 9 and 16. The length of unknown channel is set as 4. The step-size μ_k is chosen as 0.0245. As can be seen from Fig. 10 and 11, due to the ability of processing noncircular complex-valued signals and making a detection of reliable neighbors, the proposed algorithm also has good ability to resist multi-point noncircular attacks and achieves a similar convergence performance with the D-ACNSAF algorithm without attacks. As a result, the proposed algorithm has a good anti-attack capability, whether the attacks are circular or non-circular, time-varying or non-time-varying, single or multipoint.



(a)

(b)

FIGURE 12. Performance comparisons of SD-ACNSAF algorithm for noncircular Ikeda map complex-valued signals with different attacked nodes. (a) Transient network MSD. (b) Steady-state network MSD verus node.

C. DISCUSSION ON THE INFLUENCE OF ATTACK NUMBER

In order to study the influence of the number of attacks on the performance of the proposed algorithm, the performance comparison of the SD-ACNSAF with different attack numbers for noncircular complex-valued Ikeda map input signals are presented. The compromised sensors are chosen as 1, 2, 4, 5, 7, 8, 9, 11, 13, 16, respectively. The attacked nodes are randomly selected in the simulation. The parameter setting is the same as the previous experiment. As shown in Fig. 12, the proposed algorithm has a robustness performance against complex-valued attacks. However, As the number of attacks increases, the performance of the SD-ACNSAF algorithm decreases. Since the proposed algorithm is derived based on assumption 1, when the number of attacks is more than or equal to half of the number of network nodes, the anti attack performance of the algorithm will decline. This is consistent with the previous analysis. The performance of the proposed algorithm is determined by the success rate of attack detection. It is obvious that the increase of the number of attacks will bring difficulties to attack detection. However, the impact is acceptable under the assumption 1.

VI. CONCLUSION

In this article, to solve the problem of complex-valued attacks in sensor network, a secure diffusion augmented complex-valued normalized subband adaptive filter (SD-ACNSAF) algorithm is proposed to protect information against noncircular attacks, which is derived from a novel complex-valued detection method. This complex-value detection method can be considered to consist of two parts. It first needs to detect the trustworthy neighbors of each node by using reliable reference estimation and then reassembles the information from the trustworthy neighbors by using the D-ACNSAF algorithm. The theoretical analyses of the mean and mean-square performance of the proposed algorithm are presented. Simulation results show that the proposed algorithm exhibits good performance.

APPENDIX

As can be seen from (12) and (13), the elements of $\overline{w}_k(n)$ and $\overline{w}'_k(n)$ are determined by the nc-ACNSAF algorithm. Suppose the sensor is not attacked, the mean convergence condition of nc-ACNSAF can be given as follows [27]:

$$0 < \mu_k < \frac{2}{\lambda_{\max}\left(\sum_{m=1}^{N_s} \frac{E(u_{k,m}^{a*}(n)u_{k,m}^{aT}(n))}{E(u_{k,m}^{H}(n)u_{k,m}(n))}\right)} \tag{78}$$

CASE 1: Assume no attack occurs on the network. Under the above convergence condition, the weights of the nc-ACNSAF algorithm will converges to the optimum values. Since the elements of $\overline{w}_k(n)$ and $\overline{w}'_k(n)$ are composed of weights of the nc-ACNSAF algorithm, the $\overline{w}_k(n)$ and $\overline{w}'_k(n)$ converge to an unbiased estimate of optimum weights.

CASE 2: Suppose that the attack occurs on some sensors or communication paths and the attacks are random. According to the assumption *A1*, each node k has no less than $\lceil \frac{n_k}{2} \rceil$ reliable neighbors. Therefore, the reliable neighbors should lin in both sides of $\overline{w}_k^{(j)}(n)$ and $\overline{w}_k^{'(j)}(n)$. Such as:

$$\Re(w_{l_t}^{(j)}(n)) < \Re(\overline{w}_k^{(j)}(n)) < \Re(w_{l_s}^{(j)}(n)) \tag{79}$$

$$\Im(w_{l_t}^{(j)}(n)) < \Im(\overline{w}_k^{(j)}(n)) < \Im(w_{l_s}^{(j)}(n)) \tag{80}$$

$$\Re(w_{l_t}^{'(j)}(n)) < \Re(\overline{w}_k^{'(j)}(n)) < \Re(w_{l_s}^{'(j)}(n)) \tag{81}$$

$$\Im(w_{l_t}^{'(j)}(n)) < \Im(\overline{w}_k^{'(j)}(n)) < \Im(w_{l_s}^{'(j)}(n)) \tag{82}$$

where $w_{l_t}^{(j)}(n)$, $w_{l_s}^{(j)}(n)$, $w_{l_t}^{'(j)}(n)$ and $w_{l_s}^{'(j)}(n)$ are the reliable neighbors of node k. The elements of $\overline{w}_k(n)$ and $\overline{w}'_k(n)$ can be considered reliable.

As a result, the references $\overline{w}_k(n)$ and $\overline{w}'_k(n)$ converge to the true expectation of the estimation.

REFERENCES

[1] D. Estrin, L. Girod, G. Pottie, and M. Srivastava, "Instrumenting the world with wireless sensor networks," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Process. (ICASSP)*, vol. 4, May 2001, pp. 2033–2036.

[2] C. G. Lopes and A. H. Sayed, "Diffusion least-mean squares over adaptive networks: Formulation and performance analysis," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 3122–3136, Jul. 2008.

[3] S. Y. Tu and A. H. Sayed, "Mobile adaptive networks," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 4, pp. 649–664, Aug. 2011.

[4] F. S. Cattivelli and A. H. Sayed, "Modeling bird flight formations using diffusion adaptation," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 2038–2051, May 2011.

[5] G. Wen, W. Yu, Z. Li, X. Yu, and J. Cao, "Neuro-adaptive consensus tracking of multiagent systems with a high-dimensional leader," *IEEE Trans. Cybern.*, vol. 47, no. 7, pp. 1730–1742, Jul. 2017.

[6] X. Zhao and A. H. Sayed, "Distributed clustering and learning over networks," *IEEE Trans. Signal Process.*, vol. 63, no. 13, pp. 3285–3300, Jul. 2015.

[7] P. Wen and J. Zhang, "Variable step-size diffusion normalized sign-error algorithm," *Circuits Sys. Signal Process.*, vol. 37, pp. 1–12, Nov. 2018.

[8] A. Sayed, "Adaptation, learning, and optimization over networks," *Found. Trends Mach. Learn.*, vol. 7, nos. 4–5, pp. 311–801, Jul. 2014.

[9] A. H. Sayed, S.-Y. Tu, J. Chen, X. Zhao, and Z. J. Towfic, "Diffusion strategies for adaptation and learning over networks," *IEEE Signal Process. Mag.*, vol. 30, no. 3, pp. 155–171, May 2013.

[10] J. Chen, C. Richard, and A. H. Sayed, "Multitask diffusion LMS over networks," *IEEE Trans. Signal Process.*, vol. 62, no. 16, pp. 4129–4144, Aug. 2014.

[11] F. S. Cattivelli and A. H. Sayed, "Diffusion LMS strategies for distributed estimation," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1035–1048, Mar. 2010.

[12] F. S. Cattivelli, C. G. Lopes, and A. H. Sayed, "Diffusion recursive least-squares for distributed estimation over adaptive networks," *IEEE Trans. Signal Process.*, vol. 56, no. 5, pp. 1865–1877, May 2008.

[13] C. Li, S. Huang, Y. Liu, and Y. Liu, "Distributed TLS over multitask networks with adaptive intertask cooperation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 52, no. 6, pp. 3036–3052, Dec. 2016.

[14] A. Vempaty, P. Ray, and P. Varshney, "False discovery rate based distributed detection in the presence of byzantines," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 50, no. 3, pp. 1826–1840, Jul. 2014.

[15] B. Kailkhura, S. Brahma, Y. S. Han, and P. K. Varshney, "Distributed detection in tree topologies with byzantines," *IEEE Trans. Signal Process.*, vol. 62, no. 12, pp. 3208–3219, Jun. 2014.

[16] B. Kailkhura, Y. S. Han, S. Brahma, and P. K. Varshney, "Distributed Bayesian detection in the presence of byzantine data," *IEEE Trans. Signal Process.*, vol. 63, no. 19, pp. 5250–5263, Oct. 2015.

[17] P. Kaligineedi, M. Khabbazian, and V. K. Bhargava, "Secure cooperative sensing techniques for cognitive radio systems," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2008, pp. 3406–3410.

[18] G. Lu, W. Chen, and D. Huang, "Distributed diffusion least mean square algorithm based on the reputation mechanism," (in Chines), *J. Electron. Inf. Technol.*, vol. 37, no. 5, pp. 1234–1240, May 2015.

[19] Y. Liu and C. Li, "Secure distributed estimation over wireless sensor networks under attacks," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 54, no. 4, pp. 1815–1831, Aug. 2018.

[20] P. K. Dash, A. K. Pradhan, and G. Panda, "Frequency estimation of distorted power system signals using extended complex Kalman filter," *IEEE Trans. Power Del.*, vol. 14, no. 3, pp. 761–766, Jul. 1999.

[21] K. Manandhar, X. Cao, F. Hu, and Y. Liu, "Detection of faults and attacks including false data injection attack in smart grid using Kalman filter," *IEEE Trans. Control Netw. Syst.*, vol. 1, no. 4, pp. 370–379, Dec. 2014.

[22] A. Mohammadi and K. N. Plataniotis, "Improper complex-valued multiple-model adaptive estimation," *IEEE Trans. Signal Process.*, vol. 63, no. 6, pp. 1528–1542, Mar. 2015.

[23] A. Mohammadi and K. N. Plataniotis, "Noncircular attacks on phasor measurement units for state estimation in smart grid," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 4, pp. 777–789, Aug. 2018.

[24] P. Wen and J. Zhang, "Widely linear complex-valued diffusion subband adaptive filter algorithm," *IEEE Trans. Signal Inf. Process. over Netw.*, vol. 5, no. 2, pp. 248–257, Jun. 2019.

[25] A. Monticelli, *State Estimation in Electric Power Systems, A Generalized Approach*. Norwell, MA, USA: Kluwer 1999.

[26] Y. Liu, M. K. Reiter, and P. Ning, "False data injection attacks against state estimation in electric power grids," in *Proc. 16th ACM Conf. Comput. Commun. Secur. (CCS)*, May 2009, vol. 14, no. 1, pp. 1–33.

[27] D. Mandic and V. S. L. Goh, *Complex Valued Nonlinear Adaptive Filters: Noncircularity, Widely Linear and Neural Models*. Hoboken, NJ, USA: Wiley, 2009.

[28] Y. Xia, D. P. Mandic, and A. H. Sayed, "An adaptive diffusion augmented CLMS algorithm for distributed filtering of noncircular complex signals," *IEEE Signal Process. Lett.*, vol. 18, no. 11, pp. 659–662, Nov. 2011.

[29] T. T. Kim and H. V. Poor, "Strategic protection against data injection attacks on power grids," *IEEE Trans. Smart Grid*, vol. 2, no. 2, pp. 326–333, Jun. 2011.

[30] L. Liu, M. Esmalifalak, Q. Ding, V. A. Emesih, and Z. Han, "Detecting false data injection attacks on power grid by sparse optimization," *IEEE Trans. Smart Grid*, vol. 5, no. 2, pp. 612–621, Mar. 2014.

[31] S. Zhang, H. Han, X. Jin, and Y. Xia, "Complementary mean-square analysis of CNLMS algorithm using Pseudo-Energy-Conservation method," *IEEE Signal Process. Lett.*, vol. 27, pp. 1345–1349, Jul. 2020.

[32] P. Wen, J. Zhang, S. Zhang, and D. Li, "Augmented complex-valued normalized subband adaptive filter: Algorithm derivation and analysis," *J. Franklin Inst.*, vol. 356, no. 3, pp. 1604–1622, Feb. 2019.

[33] P. Wen, S. Zhang, and J. Zhang, "A novel subband adaptive filter algorithm against impulsive noise and it's performance analysis," *Signal Process.*, vol. 127, pp. 282–287, Oct. 2016.

[34] S. Zhang and W. X. Zheng, "Mean-square analysis of multi-sampled multiband-structured subband filtering algorithm," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 66, no. 3, pp. 1051–1062, Mar. 2019.

[35] X. Zhao and A. H. Sayed, "Performance limits for distributed estimation over LMS adaptive networks," *IEEE Trans. Signal Process.*, vol. 60, no. 10, pp. 5107–5124, Oct. 2012.

[36] S. Zhang, J. Zhang, and H. C. So, "Mean square deviation analysis of LMS and NLMS algorithms with white reference inputs," *Signal Process.*, vol. 131, pp. 20–26, Feb. 2017.

[37] Y. Xia and D. P. Mandic, "Complementary mean square analysis of augmented CLMS for second-order noncircular Gaussian signals," *IEEE Signal Process. Lett.*, vol. 24, no. 9, pp. 1413–1417, Sep. 2017.

[38] E. C. Menguc and N. Acir, "An augmented complex-valued least-mean kurtosis algorithm for the filtering of noncircular signals," *IEEE Trans. Signal Process.*, vol. 66, no. 2, pp. 438–448, Jan. 2018.

PENGWEI WEN received the B.S. degree from the College of Mathematics and Information Sciences, North China University of Water Resources and Electric Power, Zhengzhou, China, in 2014, and the Ph.D. degree in signal and information processing from Southwest Jiaotong University, Chengdu, China, in 2019. In 2019, he joined the School of Electronic and Information Engineering, Zhongyuan University of Technology, Zhengzhou. His research interest includes the area of distributed signal processing and adaptive filtering algorithms.

JIASHU ZHANG received the B.S. degree from the University of Electronic Science and Technology of China, Chengdu, China, in 1987, the M.S. degree from Chongqing University, Chongqing, China, in 1990, and the Ph.D. degree from the University of Electronic Science and Technology of China, in 2001. In 2001, he joined the School of Information Science and Technology, Southwest Jiaotong University, Chengdu. He is currently a Professor and the Director of the Sichuan Province Key Laboratory of Signal and Information Processing, Southwest Jiaotong University. He has published more than 200 journal and conference papers. His research interests include nonlinear and adaptive signal processing, biometric security and privacy, and information forensic and data hiding. He is a Senior Member of the Chinese Institute of Electronics and the Chinese Institute of Communications.

SHENG ZHANG (Member, IEEE) received the B.S. degree from the College of Mathematics and Information Sciences, North China University of Water Resources and Electric Power, Zhengzhou, China, in 2010, and the Ph.D. degree in signal and information processing from Southwest Jiaotong University, Chengdu, China, in 2016. From 2016 to 2017, he was a Research Fellow with the School of Computing, Engineering and Mathematics, Western Sydney University, Sydney, Australia. In 2017, he joined the School of Information Science and Technology, Southwest Jiaotong University. His research interests include the area of adaptive filtering algorithms.

GAO CHEN received the master's degree from Xiamen University, Xiamen, China, in 2009, and the Ph.D. degree from Southwest Jiaotong University, Chengdu, China, in 2016. From October 2016 to October 2018, he was a Postdoctoral Researcher with the Department of Electronic Engineering, Tsinghua University, Beijing, China. He is currently a Lecturer with the School of Electrical Engineering and Intelligentization, Dongguan University of Technology, Dongguan, China. His current research interests include signal processing and image processing.