A Class of Diffusion Zero Attracting Stochastic Gradient Algorithms With Exponentiated Error Cost Functions

ZHENGYAN LUO, HAIQUAN ZHAO, (Senior Member, IEEE), AND XIANGPING ZENG

Key Laboratory of Magnetic Suspension Technology and Maglev Vehicle, Ministry of Education, Southwest Jiaotong University, Chengdu 610031, China
School of Electrical Engineering, Southwest Jiaotong University, Chengdu 610031, China

Corresponding author: Haiquan Zhao (hqzhao_swjtu@126.com)

This work was supported in part by the National Science Foundation, China, under Grant 61871461, Grant 61571374, and Grant 61433011, and in part by the Sichuan Science and Technology Program under Grant 19YYJC0681.

ABSTRACT

In this paper, a class of diffusion zero-attracting stochastic gradient algorithms with exponentiated error cost functions is put forward due to its good performance for sparse system identification. Distributed estimation algorithms based on the popular mean-square error criterion have poor behavior for sparse system identification with color noise. To overcome this drawback, a class of stochastic gradient least exponentiated (LE) algorithms with exponentiated error cost functions were proposed, which achieved a low steady-state compared with the least mean square (LMS) algorithm. However, those LE algorithms may suffer from performance deterioration in the sparse system. For sparse system identification in the adaptive network, a polynomial variable scaling factor improved diffusion least sum of exponentials (PZA-VSIDLSE) algorithm and an \( l_p \)-norm constraint diffusion least exponentiated square (LP-DLE2) algorithm are proposed in this work. Instead of using the \( l_1 \)-norm penalty, an \( l_p \)-norm penalty and a polynomial zero-attractor are employed as a substitution in the cost functions of the LE algorithms. Then, we perform mean behavior model and mean square behavior modal of the LP-DLE2 algorithm with several common assumptions. Moreover, simulations in the context of distributed network sparse system identification show that the proposed algorithms have a low steady-state compared with the existing algorithms.

INDEX TERMS

Diffusion adaptive algorithms, exponentiated error, zero attracting, sparse system, variable scaling factor.

I. INTRODUCTION

Distributed adaptive estimation over wireless sensor networks is an emerging research field in adaptive signal processing and has recently obtained significant attention. Distributed adaptive estimation can deal with the information using the collected data at nodes deployed in different locations, and it has been applied in environment monitoring, precision agriculture, source location and tracking, and disaster relief management [1]–[3]. Moreover, it can estimate parameters by cooperation and information exchange of interconnected nodes in the adaptive network. There are two strategies in the previous work, namely the incremental strategy and the diffusion strategy. The network structure of the incremental strategy is that the node only communicates with its adjacent node, so its structure is based on cycle [4].

However, the structure is unstable which can easily fail if one of the nodes is broken. Moreover, the diffusion strategy is more stable and it is widely used in the distributed estimation [5]. Due to its robustness, each node in the network can communicate with all its neighbors and fuses the local estimation by a combination manner. The diffusion strategy also contains combination stage and adaptive stage. Due to these two stages, the combine-then-adapt (CTA) diffusion strategy and the adapt-then-combine (ATC) diffusion strategy were put forward [6]. In fact, the performance of the ATC-type outperforms that of the CTA-type [7]–[9].

In the family of gradient descent (GD) algorithms, the least mean square algorithm was first proposed in [10]. Then, an increasing number of general GD algorithms with better performances for system identification were found [11], [12]. So, a class of diffusion gradient descent algorithms was put forward, and the diffusion least-mean-square (DLMS) algorithm is seen frequently. In [13], a class of stochastic
gradient least exponentiated (LE) algorithms with exponentiated error cost function was proposed. The least exponentiated square (LE2) algorithm form the LE algorithms had a low steady-state for colored inputs and the least sum of exponentials (LSE) algorithm exhibited good robustness. However, the family of gradient descent algorithms suffers from performance deterioration in the sparse system. To overcome this problem, the zero attractors were employed in cost function for sparse system identification.

In practical application, the sparse system is usually seen in acoustic paths in hearing aids and active noise control systems. Sparse systems are the ones in which most of the coefficients of the impulse response is either zero or close to zero [14]. However, most of the traditional algorithms don’t notice sparsity. Motivated by compressive sensing methods, one of the popular classes of sparse LMS algorithms are the ones which employ the concept of zero attraction. In the sparse algorithms based on a zero attractor, if most of the coefficients are zeros, the smaller coefficients of the model are forced to zero by the zero attractor. In this task, \( l_1 \)-norm penalty was utilized in the quadratic LMS cost function. In addition, attempts have been made to employ \( l_p \)-norm (0 \( \leq \) p \( \leq \) 1) instead of the \( l_1 \)-norm. However, the zero attracting sparse algorithms have a limitation in which the zero attractor promote zero on all taps uniformly [15]. Thus, the reweighted zero-attracting LMS (RZA-LMS) algorithm was proposed with reweighted step sizes of the zero attractor for different taps. Advance in reweighted zero attracting methods have resulted in improving the limitation of sparse algorithms in the recent years, the polynomial sparse algorithm is dependent on the proper selection of a set of algorithm parameters and have a similar performance of \( l_0 \)-norm [16].

So far, the zero attracting methods are rarely utilized in distributed estimation. Although the zero attracting DLMS (ZA-DLMS) algorithm and the reweighted zero attracting DLMS (RZA-DLMS) algorithm were usually seen, the diffusion zero attracting algorithms can be further improved. Thus, in this paper an \( l_p \)-norm constraint diffusion least exponentiated square (LP-DLE2) algorithm and a polynomial variable scaling factor improved diffusion least sum of exponentials (PZA-VSIDLSE) algorithm are proposed. By taking advantage of the merit of the \( l_p \)-norm and polynomial sparse algorithm, we applied it into the diffusion LE2 algorithm and the diffusion improved LSE algorithm. In addition, a variable scaling factor is employed in the improved LSE algorithm, which has a fast convergence rate and a low steady-state [17]. Extensively, we develop their ATC and CTA variants, namely the ATC-LP-DLE2, ATC-PZA-VSIDLSE, CTA-LP-DLE2 and CTA-PZA-VSIDLSE. Moreover, we conducted the mean behavior model and mean square behavior model for the ATC-type algorithms of the LP-DLE2 algorithm. Simulation results demonstrate that the proposed algorithms have a lower steady-state as compared to the performances of several exiting algorithms under different environments.

| TABLE 1. Mathematical notation. |
|--------------------------------|
| Operators | Description |
| (1) | Transposition |
| (\otimes) | Kronecker product |
| \( l_p \) | \( l_p \)-norm |
| \( E[\cdot] \) | Expectation |
| \( \text{Tr}(\cdot) \) | Trace operation |
| diag{\cdot} | Diagonal matrix |
| col{\cdot} | Standard vectorization operation |
| \( I_N \) | \( N \) by \( N \) identity matrix |
| \( |\cdot| \) | Absolute operation |
| \( \text{sinh}[\cdot] \) | Hyperbolic sine function |
| \( \text{cosh}[\cdot] \) | Hyperbolic cosine function |
| vec(\cdot) | Stack the columns of a matrix into a column vector |

**Note:** We use normal letters to denote scalars and use boldface letters to denote vectors or matrices. The mathematical notations used in this paper are summarized in Table 1.

**II. REVIEW OF THE RELATED ALGORITHMS**

**A. THE LEAST EXPONENTIATED SQUARE (LE2) ALGORITHM**

At each time \( i \), we observe an unknown vector \( w_o = [w_o(1), w_o(2), \ldots, w_o(L)]^T \) of length \( L \) through a linear system

\[
d(i) = u_i^T w_o + v(i)
\]

where \( d(i) \) denotes the output random scalar and \( u_i = [u(i), u(i-1), \ldots, u(i - L + 1)]^T \) is the \( L \)-dimensional input vector. We define the error signal \( e_i = d(i) - u_i^T w_{i-1} \), where \( w_i \) is the estimate of the unknown weight vector and \( v(i) \) is the background noise.

Given the data modal, the objective of the least exponentiated square algorithm is derived by minimizing a differentiable cost function with exponential error, which can be formulated as [13]

\[
J_{e2}(i) = \frac{1}{2} \exp(e_i^2)
\]

Based on the gradient vector of \( J_{e2}(i) \), the filter weight update formula using the steepest-descent method is given as

\[
w_i = w_{i-1} - \mu \frac{\partial J_{e2}(i)}{\partial w_{i-1}} = w_{i-1} + \mu e_i u_i \exp(e_i^2)
\]

However, the step size of the LE2 algorithm is restrictive. To overcome this problem, an attempt to minimize the a posterior error at every iteration was been made resulting in a normalized step size. Hence, the normalization of the step...
size is obtained as [13]
\[ \mu_{e2}(i) = \frac{\mu_{e2}}{u_i^T u_i \exp(\alpha e_i^2)} \]  (4)

and the range of the normalized step size is 0 ≤ \(\mu_{e2}\) ≤ 2. A positive multiplicative factor \(\alpha\) is introduced to control the steepness of the error surface.

**B. THE LEAST SUM OF EXPONENTIALS (LSE) ALGORITHM**

The LSE algorithm is one of the classes of least exponential algorithms, which is derived by minimizing a similar cost function with the hyperbolic cosine of the adaptive error. The three different stochastic cost functions, namely the mean-square cost function, the exponential square cost function and the hyperbolic cosine cost function are compared in Fig. 1.

![Stochastic cost functions](image)

**FIGURE 1. Stochastic cost functions.**

We can observe that both the exponential square cost function and the hyperbolic cosine cost function are steeper for larger error values while the surface of the hyperbolic cosine cost function is between the mean-square cost function and the exponential square cost function.

Then, the cost function of the least sum of exponentials algorithm is given by [13]
\[ J_{se}(i) = \exp(e_i) + \exp(-e_i) \]  (5)

Obviously, the hyperbolic cosine cost function is linear combination of all adaptive error with even moment, which has more information in the LSE algorithm. Thus, according to the steepest descent method, the weight update rule of the cost function can be expressed as
\[ w_i = w_{i-1} - \mu \frac{\partial J_{se}(i)}{\partial w_{i-1}} = w_{i-1} + \mu u_i(\exp(e_i) - \exp(-e_i)) \]  (6)

where \(\mu\) is a fixed step size. In the same way as for the least exponentiated square (LE2) algorithm, a normalization of the step size can be obtained as
\[ \mu_{se}(i) = \frac{\mu_{se} e_i}{u_i^T u_i(\exp(\alpha e_i) - \exp(-\alpha e_i))} \]  (7)

and the range of \(\mu_{se}\) is also the same with that of the LE2 algorithm [13]. In addition, a positive factor \(\alpha\) is introduced to control the effect of these exponential terms.

**III. PROPOSED ALGORITHMS**

**A. THE \(l_p\)-NORM CONSTRAINT DIFFUSION LE2 ALGORITHM**

Consider the system identification problem in the adaptive networks, the network is composed of \(k \in N\) nodes where each node measures its data \([d_k(i), u_k(i)]\) to estimate the unknown weight \(w_o\) of length \(L\) through a linear model:
\[ d_k(i) = u_{k,i}^T w_o + v_k(i) \]  (8)

where \(d_k(i)\) is the output signal observation of each node, the input vector \(u_{k,i} = [u_k(i), u_k(i-1), \ldots, u_k(i-L+1)]^T\) is the \(L\)-dimensional vector. \(v_k(i)\) represents the observation noise in the adaptive networks. The error signal is defined as
\[ e_{k,i} = d_k(i) - u_{k,i}^T w_{k,i-1} \]  (9)

where \(w_{k,i-1}\) is the estimate of the weight vector \(w_o\) at time \(i-1\) and node \(k\).

Assuming that the set of nodes combined with the node \(k\) (including itself) is \(N_k\), its neighboring nodes can share information with each other by the network topology. To derive the LP-DLE2 algorithm, the local cost function can be formulated as a linear combination of an \(l_p\)-norm penalty on the coefficients and the weighted least exponential difference.

\[ J_k^{lo}(w) = \sum_{i \in N_k} a_{i,k} E[\exp(e_{1,i}^2)] + \gamma \|w_o\|^p \]

\[ = \sum_{i \in N_k} a_{i,k} E[|d(i) - u_{i}^T w|^2] + \gamma \|w_o\|^p \]  (10)

where \(a_{i,k}\) denotes the combination rule, \(\gamma\) represents the positive factor to control its zero attractor and \(\|\cdot\|_p\) is the \(l_p\)-norm. Note that the combination rule \(a_{i,k}\) satisfies
\[ a_{i,k} = 0 \text{ if } l \notin N_k \quad \Gamma = 1^T \]  (11)

where \(\Gamma\) is a \(N \times N\) matrix with individual entries \(a_{i,k}\).

Using the steepest-descent method at node \(k\), we can obtain the following recursion
\[ w_{k,i} = w_{k,i-1} - \mu \sum_{i \in N_k} \frac{\partial \left( E[\exp(e_{1,i}^2)] + \gamma \|w_o\|^p \right)}{\partial w} \Big|_{w_{k,i-1}} \]  (12)

According to the linear combination assumption, the estimate \(w_{k,i-1}\) can be rewritten as
\[ w_{k,i-1} = \sum_{l \in N_k} a_{i,k} \varphi_{l,i-1} \]  (13)

where \(\varphi_{l,i-1}\) is the local estimates at node \(l\) and at time \(i-1\). In addition, \(\|w\|_p\) denotes the \(l_p\)-norm of the weight vector \(w\), which is defined as \(\|w\|_p = (\sum_{m=0}^{L-1} |w(m)|^p)^{1/p}\). Without loss
of generality [14], [17], $|w(m)|^p$ can be replaced with the soft parameter function

$$ F_\beta[w(m)] = (1 + \beta^{-1})(1 - \exp[-\beta|w(m)|]) $$  \hfill (14)$$

where $\beta$ is an assumpive factor to control $p$ of $|w|_p$. So the approximation of $F_\beta[w(m)]$ at the extremes $\beta = \infty$ and $\beta = 0$ equals $|w(m)|^p$ with $p = 0$ and $p = 1$, respectively.

Then, introducing (13) and (14) to (12), an iterative formula for the intermediate estimate is obtained.

$$ \varphi_{l,i} = \varphi_{l,i-1} - \mu \frac{\partial E[\exp(\epsilon_{l,i}^2)] + \gamma \|w\|^p}{\partial w} w_{k,i-1} $$
$$ \approx \varphi_{l,i-1} - \mu \frac{\partial E[\exp(\epsilon_{l,i}^2)] + \gamma \|w\|^p}{\partial w} w_{l,i-1} $$
$$ = \varphi_{l,i-1} + \mu u_{l,i} e_{l,i} \exp(\epsilon_{l,i}^2) $$
$$ - \rho(1 + \beta) \exp(-\beta|\varphi_{l,i-1}|) \text{sgn}(|\varphi_{l,i-1}|) $$  \hfill (15)$$

where $\rho$ is a positive factor. However, it is not available for the linear combination estimate $w_{k,i-1}$ to require gradient calculation at node $l$. Considering the approximation in [18], the difference between $w_{k,i-1}$ and $w_{l,i-1}$ is not too large. Also, both are linear-combination estimate of $w_{l}$ at instant $i$.

Thus, the updated formula of $l_0$-norm constraint diffusion LE2 (LP-DLE2) algorithm can be obtained by transforming the steepest-descent type as follows

$$ \begin{cases} 
\varphi_{k,i} = w_{k,i-1} - \mu \frac{\partial E[\exp(\epsilon_{k,i}^2)] + \gamma \|w\|^p}{\partial w} w_{k,i-1} \\
w_{k,i} = \sum_{l \in N_k} a_{l,k} \varphi_{l,i} 
\end{cases} $$  \hfill (16)$$

where the local estimate $\varphi_{k,i}$ is replaced by linear combination $w_{k,i-1}$. In this way, the linear combination has more information from neighbor nodes than $\varphi_{k,i-1}$. Considering two-step iteration in the diffusion strategy, we can summarize its ATC and CTA forms as in Table 2.

### B. THE POLYNOMIAL VARIABLE SCALING FACTOR DIFFUSION ILSE ALGORITHM

In [17], a variable scaling factor strategy is employed in the improved LSE algorithm, which provides a faster convergence rate and smaller steady-state error. Inspired by the variable step size method, a variable scaling factor $\lambda_i$ is introduced in the diffusion improved least sum of exponentials (ILSE) algorithm.

Moreover, the polynomial sparse adaptive algorithm is utilized in the diffusion ILSE algorithm for the sparse system identification. The polynomial zero attractor can be achieved by using an $l_0$-norm in conjunction with an $l_2$-norm in the objective function of the system identification problem, and its modelling accuracy, inspired by the RZA-LMS, is constructed on a set of algorithm parameters from the proper selection. Thus, a new cost function with the network topology is proposed

$$ J_k^{loc}(w) = \sum_{l \in N_k} a_{l,k} E[\cosh(\lambda_k(i) e_{k,i})] $$
$$ + \gamma \sum_{i=0}^{L-1} \begin{cases} 
|w(i)| & \text{if } |w(i)| \leq \frac{1}{1 - \alpha'} \\
(1 + |w(i)|)^{\alpha'} - |w(i)|^{\alpha'} - 1 & \frac{1}{1 - \alpha'} \end{cases} $$  \hfill (17)$$

where $\alpha' \geq 1$ is a power factor to select the proper parameters, and the variable scaling factor $\lambda_k(i)$ is similar to variable step size at node $k$ and time $i$. Its scheme is given as [17]

$$ \lambda_k(i) = \beta \lambda_k(i-1) + \gamma \frac{1}{\lambda_k(i-1)} \cosh[\lambda_k(i-1) e_{k,i}] $$  \hfill (18)$$

where $\lambda_k(i) \in [\lambda_{min}, \lambda_{max}]$. According to [17], when $e_{k,i}$ is larger, the cost function with larger $\lambda$ has a larger gradient, resulting in a faster convergence rate; otherwise, the cost function with smaller $\lambda$ has a flat gradient curve, leading to the improvement of filter stability. Thus, the initial value closes to $\lambda_{max}$ must be chosen for $\lambda_k(i)$ to achieve a fast convergence rate, and $0 < \beta < 1$ is a positive factor.

Using the same method in the LP-DLE2 algorithm, the update formula for polynomial variable scaling factor improved diffusion LSE (PZA-VSIDLSE) algorithm can be
obtained as

\[
\begin{align*}
\varphi_{k,i} &= w_{k,i-1} - \mu \frac{\partial J_k(w)}{\partial w} w_{k,i-1} \\
w_{k,i} &= \sum_{l \in N_k} a_{l,k} \varphi_{l,i} \tag{19}
\end{align*}
\]

However, the diffusion strategy contains ATC type and CTA type and the derivation of the PZA-VSIDLSE algorithm is like that of the LP-DLE2 algorithm. Hence, we can obtain the update equations for the ATC-PZA-VSIDLSE and CTA-PZA-VSIDLSE algorithms, and they are summarized in Table 3.

### Table 3. The summary of the PZA-VSIDLSE algorithm.

| Initialization | Parameters | ATC type | CTA type |
|---------------|------------|----------|----------|
| $w_{k,i} = \varphi_{k,i} = \theta, g_{k,i} = \vartheta, \lambda_{k,i} = 1.5$ | $\mu_{w} = \alpha_{w}, \beta = \gamma_{0}, \alpha' = 6$ | $d_{i}(t) = w_{i,k} r_{w_{i,k} + v_{i}(t)}$ | $d_{i}(t) = w_{i,k} r_{w_{i,k} + v_{i}(t)}$ |
| $\varphi_{k,i} = w_{k,i-1} - \mu u_{k,i} e_{k,i} \text{sgn}(e_{k,i})$ | $\phi_{i}(t) = \beta \lambda_{i}(t-1)$ | $\varphi_{k,i} = w_{k,i-1} - \mu u_{k,i} e_{k,i} \text{sgn}(e_{k,i})$ | $\varphi_{k,i} = w_{k,i-1} - \mu u_{k,i} e_{k,i} \text{sgn}(e_{k,i})$ |
| $w_{k,i} = \sum_{l \in N_k} a_{l,k} \varphi_{l,i}$ | $e_{k,i} = \sum_{l \in N_k} a_{l,k} e_{l,i}$ | $w_{k,i} = \sum_{l \in N_k} a_{l,k} \varphi_{l,i}$ | $w_{k,i} = \sum_{l \in N_k} a_{l,k} \varphi_{l,i}$ |

### IV. PERFORMANCE ANALYSIS
In this section, we perform the transient behavior analysis for the LP-DLE2 algorithm. Since the proposed LP-DLE2 and the proposed PZA-VSIDLSE algorithms have similarities in term of analysis, so we only view the LP-DLE2 algorithm. Since both the ATC and CTA algorithms are similar in analysis, we only provide analysis for the ATC-type algorithm. To make analysis tractable, the unified model of the LP-DLE2 algorithm is rewritten as

\[
\begin{align*}
\varphi_{k,i} &= w_{k,i-1} + \mu u_{k,i} e_{k,i} \text{sgn}(e_{k,i}) - \rho g[w_{k,i-1}] \\
w_{k,i} &= \sum_{l \in N_k} a_{l,k} \varphi_{l,i} \tag{20}
\end{align*}
\]

where $g[w_{k,i-1}]$ represents $(1 + \beta) \exp[-\beta |w_{k,i-1}|] \text{sgn} [w_{k,i-1}]$ for the ATC-LP-DLE2 algorithm. First, some necessary assumptions and approximations are introduced.

**Assumption 1:** The repressor $u_{k,i}$ is temporally and spatially independent and identically distributed (i.i.d.) with zero mean $[5], [6], [19]$.

**Assumption 2:** The observed noise $v_{k}(i)$ is i.i.d. with zero mean and variance $\sigma_{v_{k}}^2$. In addition, it is independent with $u_{k,i} [1], [20]$.

**Assumption 3:** $\tilde{w}_{k,i}(m)$ denotes the $m$-th entry of the weight error vector at node $k$ and time $i$. So it meets the Gaussian distribution $\tilde{w}_{k,i}(m) \sim N(\mu_{k,i}(m), \sigma_{k,i}^2(m))$ with mean $\mu_{k,i}(m)$ and variance $\sigma_{k,i}^2(m)$. In this way, the $m$-th entry of the weight vector $w_{k,i}$ follows Gaussian distribution as $[21]–[23], [25], [26]$.$w_{k,i}(m) = w_{0}(m) - \tilde{w}_{k,i}(m) \sim N(w_{0}(m) - \mu_{k,i}(m), \sigma_{k,i}^2(m))$

**Approximation 1:** When $m \neq n$, we can consider the approximations

\[
E(g[w_{k,i}(m)]g[w_{k,i}(n)]) \approx E(g[w_{k,i}(m)])E(g[w_{k,i}(n)]) \quad \text{and} \quad E(w_{k,i}(m)g[w_{k,i}(n)]) \approx E(w_{k,i}(m))E(g[w_{k,i}(n)]) \quad [21]–[23].
\]

**Approximation 2:** When the waves of $w_{k,i}$ is too small from one iteration to the next iteration, we can make the approximations as follows $[23], [24]$

\[
E((1 + \beta) \exp[-\beta |w_{k,i-1}(m)|] \text{sgn}[w_{k,i-1}(m)]) \\
\approx (1 + \beta)E(\exp[-\beta |w_{k,i-1}(m)|]E(\text{sgn}[w_{k,i-1}(m)])
\]

**Approximation 3:** For the iteration $i \rightarrow \infty$, the Taylor series expansion of exponential error $\exp[e_{i}^2]$ can be approximated $[25], [26]$

\[
\exp[e_{i}^2] = I + \frac{1}{2!} e_{i}^2 + \frac{1}{3!} e_{i}^4 + O(\cdot) \approx I + e_{i}^2
\]

with the Peano surplus.

**Remark 1:** Assumptions 1-3 can be used in analyzing the adaptive filtering algorithms successfully. Moreover, Approximations 1-3 have the desired calculations for non-linear adaptive filtering algorithms, especially under Gaussian input signals. Furthermore, it is feasible for using these approximations to predict the behaviors of the proposed LP-DLE2 algorithm.

### A. MEAN BEHAVIOR MODEL
We define the weight vector $\bar{w}_{k,i}$ as $\bar{w}_{k,i} = w_{0} - w_{k,i}$ and the local estimate $\hat{\varphi}_{k,i}$ as $\bar{\varphi}_{k,i} = \varphi_{k,i} - \bar{w}_{k,i}$. Moreover, we make an assumption $f(e_{k,i}^2) = \exp[e_{k,i}^2]$. The global quantities of the network vectors and matrices are introduced as follows

\[
\begin{align*}
\omega_{opt} &\triangleq \text{col} \{w_{0}, w_{0}, \ldots, w_{0}\} \quad [21] \\
w_{i} &\triangleq \text{col} \{w_{1,i}, w_{2,i}, \ldots, w_{N,i}\} \quad [22] \\
\bar{w}_{i} &\triangleq \text{col} \{\bar{w}_{1,i}, \bar{w}_{2,i}, \ldots, \bar{w}_{N,i}\} \quad [23] \\
\hat{\varphi}_{i} &\triangleq \text{col} \{\hat{\varphi}_{1,i}, \hat{\varphi}_{2,i}, \ldots, \hat{\varphi}_{N,i}\} \quad [24]
\end{align*}
\]
\begin{equation}
\begin{aligned}
g[w_i] & \triangleq \text{col}[g[w_{1,i}], g[w_{2,i}], \ldots, g[w_{N,i}]] \\
U_i & \triangleq \text{diag}[u_{1,i}, u_{2,i}, \ldots, u_{N,i}] \\
f(e^2_i) & \triangleq \text{col}[f(e^2_{1,i}), f(e^2_{2,i}), \ldots, f(e^2_{N,i})]
\end{aligned}
\end{equation}

In addition, the error vector, the noise vector and the desired vector of the network are defined as
\begin{equation}
\begin{aligned}
e_i & \triangleq \text{col}[e_{1,i}, e_{2,i}, \ldots, e_{N,i}] \\
v_i & \triangleq \text{col}[v_{1}(i), v_{2}(i), \ldots, v_{N}(i)] \\
d_i & \triangleq \text{col}[d_{1}(i), d_{2}(i), \ldots, d_{N}(i)]
\end{aligned}
\end{equation}

Further the step size \(\mu\), the positive factor \(\rho\) and the combination rule \(a_{l,k}\) are defined as
\(M \triangleq \text{diag}[\mu, \ldots, \mu]\), \(\rho_s \triangleq \text{diag}[\rho, \ldots, \rho]\) and \(\Gamma_{N \times N}\). Consider the error vector \(e_i = U_i^T \tilde{w}_{l-1} + v_i\), the adaptation stage in (20) can be rewritten as
\begin{equation}
\tilde{\phi}_i = w_{opt} - w_{l-1} - QU_i u_i f(e^2_i) + \rho g[w_{l-1}]
= \tilde{w}_{l-1} - QU_i U_i^T \tilde{w}_{l-1} + v_i f(e^2_i) + \rho g[w_{l-1}]
= \tilde{w}_{l-1} - QU_i U_i^T \tilde{w}_{l-1} f(e^2_i) - QU_i v_i f(e^2_i) + \rho g[w_{l-1}]
\end{equation}
where \(Q = M \otimes I_L\) and \(\rho = \rho_s \otimes I_L\).
Introducing (30) to (20), the weight vector is rewritten as
\begin{equation}
\tilde{w}_i = P \tilde{\phi}_i
= P \tilde{w}_{l-1} - PQU_i U_i^T \tilde{w}_{l-1} f(e^2_i) + PQU_i v_i f(e^2_i) - P \rho g[w_{l-1}]
\end{equation}
where \(P = \Gamma \otimes I_L\). According to the Assumptions 1-2, the expectation of the weight vector can be given as
\begin{equation}
E[\tilde{w}_i] = PE[\tilde{w}_{l-1}] - PQE[U_i U_i^T]E[\tilde{w}_{l-1} f(e^2_i)] + P \rho E[g[w_{l-1}]]
\end{equation}
where \(E[U_i U_i^T] = S \otimes I_L\), and \(\text{Sdiag}[\sigma^2_{w_1}, \sigma^2_{w_2}, \ldots, \sigma^2_{w_N}]\).
According to the Approximation 3 and the error vector \(e_i\), we can obtain
\begin{equation}
\exp[e^2_i] \approx I + U_i \tilde{w}_{l-1} U_i^T + v_i v_i^T + U_i \tilde{w}_{l-1} v_i^T + v_i \tilde{w}_{l-1} U_i^T
\end{equation}
From the Approximation 2, the approximation of \(E[g[w_{l-1}]]\) is given. Using the Taylor series expansion of \(\exp[e^2_i]\), we can obtain that
\(E[\exp[-\beta w_{k,i-1}]] = \sum_{q=0}^{\infty} \frac{(-\beta)^q}{q!} E[|w_{k,i-1}|^q]\).
Moreover, through the Assumption 3, the \(m\)-th component of \(E[\text{sgn}[w_{k,i-1}]]\) is given by \(E[\text{sgn}[w_{k,i-1}]]\), which can be rewritten as
\begin{equation}
\begin{aligned}
E[\text{sgn}[w_{k,i-1}]]
&= -\int_{-\infty}^{0} \frac{1}{\sqrt{2\pi \sigma^2_{k,i-1}(m)}} \exp\left(-\frac{(x - (w_o(m) - \mu_{k,i-1}(m)))^2}{2\sigma^2_{k,i-1}(m)}\right) dx \\
&+ \int_{0}^{+\infty} \frac{1}{\sqrt{2\pi \sigma^2_{k,i-1}(m)}} \exp\left(-\frac{(x - (w_o(m) - \mu_{k,i-1}(m)))^2}{2\sigma^2_{k,i-1}(m)}\right) dx
\end{aligned}
\end{equation}

B. MEAN SQUARE BEHAVIOR MODEL
Multiplying both sides of (32) by \(\tilde{w}_i^T\) gives
\begin{equation}
\tilde{w}_i w_i^T = P \tilde{w}_{l-1} \tilde{w}_{l-1}^T P^T + PQU_i U_i^T \tilde{w}_{l-1} f(e^2_i) + PQU_i v_i f(e^2_i) + P \rho g[w_{l-1}] \rho \rho^T
\end{equation}
\[ -P\tilde{w}_{i-1}f(e_i^2)\tilde{w}_{i-1}^T U_i^T Q_i^T P_i^T + P\tilde{w}_{i-1}g[w_{i-1}]P_i^T - PQU_i^T \tilde{w}_{i-1}f(e_i^2)\tilde{w}_{i-1}^TP_i^T + PQU_i^T \tilde{w}_{i-1}f(e_i^2)\tilde{w}_{i-1}^TP_i^T \]
\[ -PQU_i^T \tilde{w}_{i-1}f(e_i^2)g[w_{i-1}]P_i^TP_i^T - PQU_i^T \tilde{w}_{i-1}f(e_i^2)\tilde{w}_{i-1}^TP_i^T + P\rho g[w_{i-1}]\tilde{w}_{i-1}^TP_i^T \]
\[ -P\rho g[w_{i-1}]f(e_i^2)\tilde{w}_{i-1}^TP_i^T \] (38)

Invoking Assumption 1, Assumption 2 and taking the expectations of both sides of (36), we obtain
\[ E[\tilde{w}_{i-1}] = PE[\tilde{w}_{i-1}]P_i^T + P\rho E[g[w_{i-1}]g[w_{i-1}]P_i^T \]
\[ + PQE[U_i^T]E[\tilde{w}_{i-1}f(e_i^2)f(e_i^2)]\tilde{w}_{i-1}^TP_i^T - PQE[U_i^T]Q_i^T P_i^T \]
\[ - PE[\tilde{w}_{i-1}f(e_i^2)\tilde{w}_{i-1}^TP_i^T] - PE[\tilde{w}_{i-1}g[w_{i-1}]\tilde{w}_{i-1}^TP_i^T \]
\[ - PQE[U_i^T]E[\tilde{w}_{i-1}f(e_i^2)g[w_{i-1}]P_i^T] + P\rho E[g[w_{i-1}]f(e_i^2)\tilde{w}_{i-1}^TP_i^T \]
\[ - P\rho E[g[w_{i-1}]f(e_i^2)\tilde{w}_{i-1}E[U_i^T]Q_i^TP_i^T \] (39)

To complete the analysis, the Kronecker product operation is introduced. Thus, we have vec(XYZ) = (Z^T ⊗ X) vec(Y) where the arbitrary matrices \{X, Y, Z\} are compatible dimensions. Employing this operation in (37), the new equation will be yielded
\[ vec(\tilde{W}_i) = (AT ⊗ AT)vec(\tilde{W}_{i-1}) + (B^T ⊗ B^T)vec(E[g[w_{i-1}]g[w_{i-1}]]) \]
\[ + (C^T ⊗ C^T)vec(E[U_i^T]f(e_i^2)f(e_i^2)]\tilde{w}_{i-1}^TP_i^T] \]
\[ - (B^T ⊗ B^T)vec(E[\tilde{w}_{i-1}f(e_i^2)\tilde{w}_{i-1}^TP_i^T] \]
\[ - (D^T ⊗ D^T)vec(E[\tilde{w}_{i-1}g[w_{i-1}]\tilde{w}_{i-1}^TP_i^T] \]
\[ - (B^T ⊗ D^T)vec(E[\tilde{w}_{i-1}f(e_i^2)g[w_{i-1}]\tilde{w}_{i-1}^TP_i^T] \]
\[ - (D^T ⊗ D^T)vec(E[g[w_{i-1}]f(e_i^2)\tilde{w}_{i-1}^TP_i^T] \] (40)

where \( W_iE[w_i^T], A = P^TPB = E[U_i^T]Q_i^TP_i^T, C = Q_i^TP_i^T, \) \( D = \rho P_i^T \). The network mean-square deviation (MSD) is defined as the average of every MSD at node \( k, i.e., \)
\[ MSD_{net,i} = \frac{1}{N} \sum_{k=1}^{N} MSD_{h,i}. \] And the MSD_{net,i} is given by (33) and (40). In addition, from Assumption 1 and Assumption 2 \( E[U_i^T]f(e_i^2)U_i^T \) in (40) are calculated, it can be expressed as
\[ E[U_i^T]f(e_i^2)U_i^T = G \otimes I_L \] (41)

where \( G \equiv diag(\sigma_{w_1}^2 \sigma_{v_1}^2, \sigma_{w_2}^2 \sigma_{v_2}^2, \ldots, \sigma_{w_N}^2 \sigma_{v_N}^2). \)

Then, under Assumption 1, Assumption 2 and Assumption 3, the focus is to calculate several expectations in (40), including \( E[g[w_{i-1}]g[w_{i-1}]], E[\tilde{w}_{i-1}f(e_i^2)f(e_i^2)]\tilde{w}_{i-1}^TP_i^T \)
\( E[U_i^T]f(e_i^2)f(e_i^2)]w_i^T U_i^T \) and \( E[\tilde{w}_{i-1}f(e_i^2)\tilde{w}_{i-1}] \). Given that \( E[w_{i-1}f(e_i^2)g[w_{i-1}] \) and \( E[\tilde{w}_{i-1}f(e_i^2)]\tilde{w}_{i-1}] \) can be rewritten as \( w_{opt}E(f(e_i^2)g[w_{i-1}] \) and \( \tilde{w}_{opt}E(g[w_{i-1}] \) - \( E[w_{i-1}f(e_i^2)g[w_{i-1}] \) using Approximation 1 and Approximation 2.

V. COMPUTER SIMULATIONS
In this section, the proposed LP-DLE2 and PZA-VSDLSE algorithms are simulated for the sparse system identification. The adaptive filter and the unknown system are assumed to have the same number of taps. Consider the proposed algorithms in the adaptive network, the 10 log_{10} MSD_{net,i} is to test the performances of those algorithms [27], [28]. In this work, the uniform rule is used in the simulations, which is defined as \( \mu_{i,k} = 1/|N_k| \) for all \( I \). The results of all tests are obtained by averaging 100 independent trails.

A. SYNTHETIC SPARE SYSTEM
We define the network with 20 nodes, shown in Fig. 2 [27]. The parameter vector of the unknown time-varying system \( w_o \) has \( L = 16 \) taps. Initially, only one coefficient of the unknown system is set to 1, and the others are set to 0. So, the sparse ratio of the unknown system is 1/16. Notably, the position of its only one coefficient is random and the other coefficients are equal. In this section, both the Gaussian inputs and colored inputs are considered to examine the algorithms. However, different nodes have different inputs and background noises, where the variance of the Gaussian inputs and background noises are shown in Fig. 3. The colored inputs are generated by passing the Gaussian inputs through a first-order system \( H_1(z) = 1/1 - 0.3e^{-1}. \)

![FIGURE 2. Network topology of 20 nodes.](image-url)
robustness of the proposed algorithms. The unknown system is the same as that defined in Section 5.1.

Experiment 1. In the first set of simulations, the Gaussian inputs are considered where the background noise is an independent white Gaussian noise whose variances are given in Fig. 3. To achieve the results of the experiment easily, we test the proposed ATC-type algorithms and its CTA-type algorithms respectively. Then, the proposed LP-DLE2 and the PZA-VSIDLSE algorithms are compared with the diffusion LSE algorithm and the reweighted ZA-DLMS algorithm. In order to fairly compare all algorithms, all the algorithms were initialized in such a way that they have the same initial convergence rate. The filter parameters of the proposed LP-DLE2 algorithm are set as $\mu = 0.3; \beta = 5; \rho = 0.005; \alpha = 1.2$, the filter parameters of the proposed PZA-VSIDLSE algorithm are set as $\mu_{se} = 1.2; \alpha = 1.2; \alpha' = 6; \beta = 0.996; \gamma = 0.0001$ and $\rho = 0.005; \lambda = 1.5$. The MSD performance curves results are shown in Fig. 4 and Fig. 5.

Fig. 4 shows the comparison of the proposed ATC-LP-LSE2, ATC-PZA-VSIDLSE algorithms and other existing algorithms. It is observed that the steady-state misalignments of the proposed ATC-LP-LSE2 and ATC-PZA-VSIDLSE algorithms are -34 dB and -32 dB. However, the steady-state misalignments of the proposed CTA-LP-LSE2 and CTA-PZA-VSIDLSE algorithms are -27 dB and -23 dB from Fig. 5. So, the proposed algorithms have a considerable improvement compared with the conventional DLSE and the reweighted DLMS algorithms.

Experiment 2. In this example, computer simulations are conducted to verify the performance of the proposed algorithm for AR (1) input signal that is colored signal. As for the colored inputs are the same as that in Section 4.1. Under the same filter parameters, the step size of the proposed LP-LSE2 and PZA-VSIDLSE algorithms are $\mu = 0.18$ and $\mu_{se} = 0.8$. The MSD curves for the ATC-type and the CTA-type are shown in Fig. 6 and Fig. 7.

Under the colored inputs, the steady-state misalignments of the proposed ATC-LP-LSE2 and ATC-PZA-VSIDLSE algorithms are -35 dB and -34 dB from Fig. 6. Moreover, the
steady-state misalignments of those CTA-type algorithms are -31 dB and -29 dB in Fig. 7. It is clearly observed that the convergence performance of the LP-DLE2 and PZA-VSIDLSE algorithms are not severely disturbed by colored signal, and the proposed LP-DLE2 and PZA-VSIDLSE algorithms achieve improved performance.

Experiment 3. In this work, the impulsive noise is also added to the output and the desired signal $d_k(i)$ is disturbed with a signal-to-interference ratio of -10 dB at nodes $k$. The impulsive noise is modelled by the Bernoulli-Gaussian (BG) distribution, i.e., $\vartheta_k(n) = c_k(n)A_k(n)$, where $A_k(n)$ is a zero-mean white Gaussian random sequence with variance $\sigma^2_{A_k}$ and $c_k(n)$ is a Bernoulli process with the probability density function defined by $P(c_k(n) = 1) = Pr_k$ and $P(c_k(n) = 0) = 1 - Pr_k$. In this test, the value of $Pr_k$ is set to 0.01. Under the same filter parameters, the step size of the proposed LP-DLE2 and PZA-VSIDLSE algorithms are $\mu_{c2} = 0.3$ and $\mu_{se} = 1.15$. The results of the ATC-type and the CTA-type are depicted in Fig. 8 and Fig. 9.

Moreover, the steady-state misalignments of those CTA-type algorithms are -28 dB and -25 dB in Fig. 9. It is clearly seen that the LP-DLE2 and PZA-VSIDLSE algorithms have better robustness than the other algorithms and achieve improved performance.

VI. CONCLUSION

In this paper, a class of diffusion zero attracting stochastic gradient algorithms with exponentiated error cost functions are proposed by incorporating the zero attractors and the variable step size method into the conventional diffusion cost function with exponentiated error. Due to the use of the zero-attractor and the variable scaling factor, the proposed algorithms provide improved performance with the colored signal in the sparse system. In addition, an $l_p$-norm penalty and a polynomial zero-attractor provide better performance compared with the existing zero-attracting algorithms. Moreover, the method of variable step size is incorporated into the proposed PZA-VSIDLSE algorithm which is superior to the conventional DLSE algorithm. Simulations are implemented to verify the performances of the LP-DLE2 and PZA-VSIDLSE algorithms for the sparse system identification. The results of the experiments show that the proposed algorithms have good convergence performance and robustness.

REFERENCES

[1] J. Chen and A. H. Sayed, “Diffusion adaptation strategies for distributed optimization and learning over networks,” IEEE Trans. Signal Process., vol. 60, no. 8, pp. 4289–4305, Aug. 2012.

[2] L. Shi and H. Zhao, “Two diffusion proportionate sign subband adaptive filtering algorithms,” Circuits, Syst., Signal Process., vol. 36, no. 10, pp. 4242–4259, Oct. 2017.

[3] R. Abdoolie, B. Champagne, and A. H. Sayed, “Diffusion adaptation over multi-agent networks with wireless link impairments,” IEEE Trans. Mobile Comput., vol. 15, no. 6, pp. 1362–1376, Jun. 2016.

[4] C. G. Lopes and A. H. Sayed, “Incremental adaptive strategies over distributed networks,” IEEE Trans. Signal Process., vol. 55, no. 8, pp. 4064–4077, Aug. 2007.

[5] C. G. Lopes and A. H. Sayed, “Diffusion least-squares over adaptive networks: Formulation and performance analysis,” IEEE Trans. Signal Process., vol. 56, no. 7, pp. 3122–3136, Jul. 2008.
[6] F. S. Cattivelli and A. H. Sayed, “Diffusion strategies for distributed Kalman filtering and smoothing,” IEEE Trans. Autom. Control, vol. 55, no. 9, pp. 2069–2084, Sep. 2010.

[7] Y. Li and M. Hamamura, “Zero-attracting variable-step-size least mean square algorithms for adaptive sparse channel estimation,” Int. J. Adapt. Control Signal Process., vol. 29, no. 9, pp. 1189–1206, Sep. 2015.

[8] A. H. Sayed, “Diffusion adaptation over networks,” Academic Press Library in Signal Processing. Amsterdam, The Netherlands: Elsevier, 2013.

[9] A. H. Sayed, S.-Y. Tu, J. Chen, X. Zhao, and Z. J. Towfic, “Diffusion strategies for adaptation and learning over networks: An examination of distributed strategies and network behavior,” IEEE Signal Process. Mag., vol. 30, no. 3, pp. 155–171, May 2013.

[10] A. H. Sayed, “Adaptive networks,” Proc. IEEE, vol. 102, no. 4, pp. 460–497, Apr. 2014.

[11] E. Walach and B. Widrow, “The least mean fourth (LMF) adaptive algorithm and its family,” IEEE Trans. Inf. Theory, vol. IT-30, no. 2, pp. 275–283, Mar. 1984.

[12] O. Tanrikulu and A. Constantinides, “Least mean kurtosis: A novel higher-order statistics based adaptive filtering algorithm,” Electron. Lett., vol. 30, no. 3, pp. 189–190, 1994.

[13] C. Boukis, D. P. Mandic, and A. G. Constantinides, “A class of stochastic gradient algorithms with exponentiated error cost functions,” Digital Signal Process., vol. 19, no. 2, pp. 201–212, 2009.

[14] K. Shi and P. Shi, “Convergence analysis of sparse LMS algorithms with ℓ1-norm penalty based on white input signal,” IEEE Signal Process., vol. 90, no. 12, pp. 3289–3293, Dec. 2010.

[15] L. Lu, H. Zhao, W. Wang, and Y. Yu, “Performance analysis of the robust diffusion normalized least mean p-power algorithm,” IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 65, no. 12, pp. 2047–2051, Dec. 2018.

[16] J. Maheshwari and N. V. George, “Polynomial sparse adaptive algorithm,” Electron. Lett., vol. 52, no. 25, pp. 2063–2065, 2016.

[17] S. Wang, Y. Zheng, S. Duan, L. Wang, and C. K. Tse, “A class of improved least sum of exponentials algorithms,” IEEE Signal Process., vol. 128, pp. 340–349, Nov. 2016.

[18] C. Li, P. Shen, Y. Liu, and Z. Zhang, “Diffusion information theoretic learning for distributed estimation over network,” IEEE Trans. Signal Process., vol. 61, no. 16, pp. 4011–4024, Aug. 2013.

[19] 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.

[20] J. Chen and A. H. Sayed, “Distributed Pareto optimization via diffusion strategies,” IEEE J. Sel. Topics Signal Process., vol. ? , no. 2, pp. 205–220, Apr. 2013.

[21] J. Chen, C. Richard, Y. Song, and D. Brie, “Transient performance analysis of zero-attracting LMS,” IEEE Signal Process. Lett., vol. 23, no. 12, pp. 1786–1790, Dec. 2016.

[22] W. Wang, H. Zhao, and B. Chen, “Bias compensated zero attracting normalized least mean square adaptive filter and its performance analysis,” IEEE Signal Processing, vol. 143, pp. 94–105, Feb. 2018.

[23] J. Chen, C. Richard, J. C. M. Bermudez, and P. Honeine, “Nonnegative least-mean-square algorithm,” IEEE Trans. Signal Process., vol. 59, no. 11, pp. 5225–5235, Nov. 2011.

[24] G. Su, J. Jin, Y. Gu, and J. Wang, “Performance analysis of ℓ0 norm constraint least mean square algorithm,” IEEE Trans. Signal Process., vol. 60, no. 5, pp. 2223–2235, May 2012.

[25] L. Shi, H. Zhao, and Y. Zakharov, “Performance analysis of shrinkage linear complex-valued LMS algorithm,” IEEE Signal Process. Lett., vol. 26, no. 8, pp. 1202–1206, Aug. 2019.

[26] L. Shi, H. Zhao, W. Wang, and L. Lu, “Combined regularization parameter for normalized LMS algorithm and its performance analysis,” IEEE Signal Processing, vol. 126, pp. 75–82, Sep. 2019.

[27] L. Shi and H. Zhao, “Diffusion leaky zero attracting least mean square algorithm and its performance analysis,” IEEE Access, vol. 6, pp. 56911–56923, 2018.

[28] L. Lu and H. Zhao, “Diffusion leaky LMS algorithm: Analysis and implementation,” IEEE Signal Process., vol. 140, pp. 77–86, Nov. 2017.