ROBUST DIFFUSION RECURSIVE LEAST SQUARES ESTIMATION WITH SIDE INFORMATION FOR NETWORKED AGENTS

Yi Yu *†, *Haiquan Zhao**, *Rodrigo C. de Lamare* †‡ *and Yuriy Zakharov* †

* *School of Electrical Engineering, Southwest Jiaotong University, Chengdu, China*
† *Department of Electronic Engineering, University of York, York YO10 5DD, U.K.*
‡ *CETUC, PUC-Rio, Rio de Janeiro 22451-900, Brazil*
yuyi_xyuan@163.com, hqzhao_swjtu@126.com, rcdl500@ohm.york.ac.uk, yury.zakharov@york.ac.uk

ABSTRACT

This work develops a robust diffusion recursive least squares algorithm to mitigate the performance degradation often experienced in networks of agents in the presence of impulsive noise. This algorithm minimizes an exponentially weighted least-squares cost function subject to a time-dependent constraint on the squared norm of the intermediate estimate update at each node. With the help of side information, the constraint is recursively updated in a diffusion strategy. Moreover, a control strategy for resetting the constraint is also proposed to retain good tracking capability when the estimated parameters suddenly change. Simulations show the superiority of the proposed algorithm over previously reported techniques in various impulsive noise scenarios.

Index Terms— Diffusion cooperation, distributed algorithms, impulsive noise, robust recursive least squares.

1. INTRODUCTION

In the last decade, distributed adaptive algorithms for estimating parameters of interest over wireless sensor networks with multiple nodes (or agents) have attracted significant attention, due to their performance advantages and robustness [1]. The core idea is that each node performs adaptive estimation in cooperation with its neighboring nodes. Distributed adaptive algorithms have been applied to many problems, e.g., frequency estimation in power grid [2] and spectrum estimation [3]. According to the cooperation strategy of interconnected nodes, existing algorithms can be categorized as the incremental [4], consensus [5,6], and diffusion [7–9] types. The diffusion protocol is the most popular [5] because it does not require a Hamiltonian cycle path as does the incremental type [4]; it is stable and has a better estimation performance than the consensus type [5]. Several diffusion-based distributed algorithms have been proposed such as the diffusion least mean square (dLMS) algorithm [7], diffusion recursive least squares (dRLS) algorithm [8], and their modifications [10–13].

In practice, measurements at the network nodes can be corrupted by impulsive noise [14]. Impulsive noise has the property that its occurence probability is small and magnitude is typically much higher than the nominal measurement. It is well-known that impulsive noise deteriorates significantly the performance of many algorithms in the single-agent case. In addition, for distributed algorithms in the multi-agent case, impulsive noise can also propagate over the entire network due to the exchange of information among nodes. To reduce the effects of impulsive noise, many robust distributed algorithms have been proposed [15–18]. Some algorithms, e.g., the diffusion sign error LMS (dSE-LMS) [15], are based on using the instantaneous gradient-descent method to minimize an individual robust criterion. In [16], a robust variable weighting coefficients dLMS (RVWC-dLMS) algorithm was developed, which only considers the data and intermediate estimates from nodes not affected by impulsive noise; this is based on a judgement whether impulsive noise samples occur or not. However, these robust algorithms have slow convergence, especially for colored input signals at nodes.

RLS-based algorithms have a good decorrelating property for colored input signals, thereby providing fast convergence. In this paper, therefore, we present a robust dRLS (R-dRLS) algorithm for distributed estimation over networks disturbed by impulsive noise. The R-dRLS algorithm minimizes a local exponentially weighted least-squares (LS) cost function subject to a time-dependent constraint on the squared norm of the intermediate estimate at each node. Unlike the framework in [19], we consider here a multi-agent scenario with the diffusion strategy. Furthermore, in order to equip the R-dRLS algorithm with the ability to withstand sudden changes in the environment, we also propose a diffusion-based distributed nonstationary control (DNC) method. This paper is organized as follows. In Section 2, the estimation problem in the network is described. In Section 3, the proposed algorithm is derived. In Section 4, results of simulation in impulsive noise scenarios are presented. Finally, conclusions are given in Section 5.

2. PROBLEM FORMULATION

Let us consider a network that has N nodes distributed over some region in space, where a link between two nodes means that they can communicate directly with each other. The neighborhood of node k is denoted by $\mathcal{N}_k$, i.e., a set of all nodes connected to node k including itself. The cardinality of $\mathcal{N}_k$ is denoted by n_k. At every time instant $i \geq 0$, every node k observes a data regressor vector $\boldsymbol{u}_{k,i}$ of size $M \times 1$ and a scalar measurement $d_k(i)$, related as:

$$d_k(i) = \boldsymbol{u}_{k,i}^T \boldsymbol{w}^o + v_k(i), \tag{1}$$

where the superscript T denotes the transpose, $\boldsymbol{w}^o$ is a parameter vector of size $M \times 1$, and $v_k(i)$ is the additive noise at node k. The regressors $\boldsymbol{u}_{k,i}$ and $\boldsymbol{u}_{l,j}$ are spatially independent for $k \neq l$. The

This work was partly supported by National Science Foundation of P.R. China (Grant: 61271340, 61571374 and 61433011), and the work of Y. Yu was also supported by China Scholarship Council Funding. The work of Y. Zakharov was partly supported by the UK Engineering and Physical Sciences Research Council (EPSRC) through Grants EP/P017975/1 and EP/R003297/1. (*Corresponding author: Haiquan Zhao*)

additive noises $v_k(i)$ and $v_l(j)$ are spatially and temporally independent for $k \neq l$ and $i \neq j$. Moreover, any $\boldsymbol{u}_{k,i}$ is independent of any $v_l(j)$. The model in (1) is widely used in many applications [1, 20].

The task is to estimate $\boldsymbol{w}^o$ using the available data collected at nodes, i.e., $\{\boldsymbol{u}_{k,i}, d_k(i)\}_{k=1}^N$. For this purpose, the global LS-based estimation problem is described as [8]:

$$\boldsymbol{w}_i = \arg\min_{\boldsymbol{w}} \left\{ \lambda^{i+1}\delta\|\boldsymbol{w}\|_2^2 + \sum_{j=0}^{i} \lambda^{i-j} \sum_{k=1}^{N} \left(d_k(j) - \boldsymbol{u}_{k,i}^T \boldsymbol{w} \right)^2 \right\}, \tag{2}$$

where $\|\cdot\|_2$ denotes the l_2-norm of a vector, $\delta > 0$ is a regularization constant, and λ is the forgetting factor. The dRLS algorithm solves (2) in a distributed manner [8]. In practice, $v_k(i)$ may contain impulsive noise, severely corrupting the measurement $d_k(i)$. With such noise processes, the algorithms obtained from (2), e.g., the dRLS algorithm, would fail to work.

3. PROPOSED DISTRIBUTED ALGORITHM

3.1. Derivation of R-dRLS algorithm

We focus here on the adapt-then-combine (ATC) implementation of the diffusion strategy, which has been shown to outperform the combine-then-adapt (CTA) implementation[1] [5]. Following the ATC-diffusion strategy [7, 8], i.e., performing first the adaptation step and then the combination step, the R-dRLS algorithm will be derived in the sequel.

We start with the adaptation step. Every node k, at time instant i, finds an intermediate estimate $\boldsymbol{\psi}_{k,i}$ of $\boldsymbol{w}^o$ by minimizing the individual local cost function:

$$\begin{aligned} J_k(\boldsymbol{\psi}_{k,i}) =& \|\boldsymbol{\psi}_{k,i} - \boldsymbol{w}_{k,i-1}\|_{\boldsymbol{Q}_{k,i}}^2 \\ &+ [d_k(i) - \boldsymbol{u}_{k,i}^T \boldsymbol{\psi}_{k,i}]^2, \end{aligned} \tag{3}$$

with $\boldsymbol{Q}_{k,i} = \boldsymbol{R}_{k,i} - \boldsymbol{u}_{k,i}\boldsymbol{u}_{k,i}^T$, where

$$\begin{aligned} \boldsymbol{R}_{k,i} \triangleq& \lambda^{i+1}\delta\boldsymbol{I} + \sum_{j=0}^{i} \lambda^{i-j} \boldsymbol{u}_{k,j}\boldsymbol{u}_{k,j}^T \\ =& \lambda\boldsymbol{R}_{k,i-1} + \boldsymbol{u}_{k,i}\boldsymbol{u}_{k,i}^T \end{aligned} \tag{4}$$

is the time-averaged correlation matrix for the regression vector at node k, $\boldsymbol{w}_{k,i-1}$ is an estimate of $\boldsymbol{w}^o$ at node k at time instant $i-1$, and $\boldsymbol{I}$ is the identity matrix. Notice that the form $\|\boldsymbol{x}\|_{\boldsymbol{Q}}^2 \triangleq \boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x}$ in (3) defines the Riemmanian distance [21] between vectors $\boldsymbol{\psi}_{k,i}$ and $\boldsymbol{w}_{k,i-1}$. Setting the derivative of $J_k(\boldsymbol{\psi}_{k,i})$ with respect to $\boldsymbol{\psi}_{k,i}$ to zero, we obtain

$$\boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i-1} + \boldsymbol{P}_{k,i}\boldsymbol{u}_{k,i}e_k(i), \tag{5}$$

where $e_k(i) = d_k(i) - \boldsymbol{u}_{k,i}^T\boldsymbol{w}_{k,i-1}$ stands for the output error at node k and $\boldsymbol{P}_{k,i} \triangleq \boldsymbol{R}_{k,i}^{-1}$. Using the matrix inversion lemma [20], we have

$$\boldsymbol{P}_{k,i} = \frac{1}{\lambda}\left(\boldsymbol{P}_{k,i-1} - \frac{\boldsymbol{P}_{k,i-1}\boldsymbol{u}_{k,i}\boldsymbol{u}_{k,i}^T\boldsymbol{P}_{k,i-1}}{\lambda + \boldsymbol{u}_{k,i}^T\boldsymbol{P}_{k,i-1}\boldsymbol{u}_{k,i}} \right), \tag{6}$$

where $\boldsymbol{P}_{k,i}$ is initialized as $\boldsymbol{P}_{k,0} = \delta^{-1}\boldsymbol{I}$. Since $\boldsymbol{w}_{k,i-1} = \boldsymbol{R}_{k,i-1}^{-1}\boldsymbol{z}_{k,i-1}$, where $\boldsymbol{z}_{k,i} = \lambda\boldsymbol{z}_{k,i-1} + \boldsymbol{u}_{k,i}d_k(i)$, (5) means that every node k performs an RLS update. However, with the update (5), the adverse effect of an impulsive noise sample at time instant i will propagate through nodes via $e_k(i)$. This effect can last for many iterations. To make the algorithm robust in impulsive noise scenarios, we propose to minimize (3) under the following constraint:

$$\|\boldsymbol{\psi}_{k,i} - \boldsymbol{w}_{k,i-1}\|_2^2 \leq \xi_k(i-1), \tag{7}$$

where $\xi_k(i-1)$ is a positive bound. This constraint is employed to enforce the squared norm of the update of the intermediate estimate not to exceed the amount $\xi_k(i-1)$ regardless of the type of noise (possibly, impulsive noise), thereby guaranteeing the robustness of the algorithm. If (5) satisfies (7), i.e.,

$$\|\boldsymbol{g}_{k,i}\|_2|e_k(i)| \leq \sqrt{\xi_k(i-1)}, \tag{8}$$

where $\boldsymbol{g}_{k,i} \triangleq \boldsymbol{P}_{k,i}\boldsymbol{u}_{k,i}$ represents the Kalman gain vector, then (5) is a solution of the above constrained minimization problem. On the other hand, if (8) is not satisfied (usually in the case of appearance of impulsive noise samples), i.e., $\|\boldsymbol{g}_{k,i}\|_2|e_k(i)| > \sqrt{\xi_k(i-1)}$, we propose to replace the update (5) by a normalized form to satisfy the constraint (7), which is described by

$$\boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i-1} + \sqrt{\xi_k(i-1)}\frac{\boldsymbol{g}_{k,i}}{\|\boldsymbol{g}_{k,i}\|_2}\mathrm{sign}(e_k(i)), \tag{9}$$

where $\mathrm{sign}(\cdot)$ is the sign function. Consequently, combining (5), (8) and (9), we obtain the adaptation step for each node k as:

$$\boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i-1} + \min\left[\frac{\sqrt{\xi_k(i-1)}}{\|\boldsymbol{g}_{k,i}\|_2|e_k(i)|},\ 1 \right] \boldsymbol{g}_{k,i}e_k(i). \tag{10}$$

At the combination step, the intermediate estimates $\psi_{m,i}$ from the neigborhood $m \in \mathcal{N}_k$ of node k are linearly weighed, yielding a more reliable estimate $\boldsymbol{w}_{k,i}$ [1]:

$$\boldsymbol{w}_{k,i} = \sum_{m\in\mathcal{N}_k} c_{m,k}\boldsymbol{\psi}_{m,i}, \tag{11}$$

where the combination coefficients $\{c_{m,k}\}$ are non-negative, and satisfy:

$$\sum_{m\in\mathcal{N}_k} c_{m,k} = 1, \text{ and } c_{m,k} = 0 \text{ if } m \notin \mathcal{N}_k. \tag{12}$$

Note that node k assigns a weight $c_{m,k}$ to the intermediate estimate $\boldsymbol{\psi}_{m,i}$ received from its neighbor node m. In general, $\{c_{m,k}\}$ are determined by a static rule (e.g., the Metropolis rule [22] that we adopt in this paper) which keeps them constant in the estimation, or an adaptive rule [22].

It is evident that the bound $\xi_k(i)$ controls the robustness of the algorithm against impulsive noise and influences its dynamic behavior, so choosing its value properly is of fundamental importance. To this end, motivated by the single-agent case in [19], $\xi_k(i)$ is adjusted recursively based on the diffusion strategy as:

$$\begin{aligned} \zeta_k(i) =& \beta\xi_k(i-1) + (1-\beta)\|\boldsymbol{\psi}_{k,i} - \boldsymbol{w}_{k,i-1}\|_2^2 \\ =& \beta\xi_k(i-1) + (1-\beta)\min[\|\boldsymbol{g}_{k,i}\|_2^2 e_k^2(i), \xi_k(i-1)], \\ \xi_k(i) =& \sum_{m\in\mathcal{N}_k} c_{m,k}\zeta_m(i), \end{aligned} \tag{13}$$

where β is a forgetting factor, $0 < \beta \lesssim 1$. In (13), at every node k, $\xi_k(i)$ can be initialized as $\xi_k(0) = E_c\sigma_{d,k}^2/(M\sigma_{u,k}^2)$, where E_c is

[1] In fact, the CTA version is obtained by reversing the adaptation step and combination step in the ATC version.

Table 1. Proposed R-dRLS Algorithm with the DNC Method.

Parameters: $0 < \beta \lesssim 1$, λ, δ and E_c (R-dRLS); ϱ and t_{th} (DNC)

Initialization: $\boldsymbol{w}_{k,0} = \boldsymbol{0}$, $\boldsymbol{P}_{k,0} = \delta^{-1}\boldsymbol{I}$ and $\xi_k(0) = E_c \frac{\sigma^2_{d,k}}{M\sigma^2_{u,k}}$ (R-dRLS)

$\Theta_{\text{old},k} = \Theta_{\text{new},k} = 0$, $V_t = \varrho M$, and $V_d = 0.75V_t$ (DNC)

R-dRLS algorithm:

$e_k(i) = d_k(i) - \boldsymbol{u}^T_{k,i}\boldsymbol{w}_{k,i-1}$

$\boldsymbol{P}_{k,i} = \frac{1}{\lambda}\left(\boldsymbol{P}_{k,i-1} - \frac{\boldsymbol{P}_{k,i-1}\boldsymbol{u}_{k,i}\boldsymbol{u}^T_{k,i}\boldsymbol{P}_{k,i-1}}{\lambda + \boldsymbol{u}^T_{k,i}\boldsymbol{P}_{k,i-1}\boldsymbol{u}_{k,i}}\right)$

$\boldsymbol{g}_{k,i} = \boldsymbol{P}_{k,i}\boldsymbol{u}_{k,i}$

$\boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i-1} + \min\left[\frac{\sqrt{\xi_k(i-1)}}{\|\boldsymbol{g}_{k,i}\|_2|e_k(i)|},\ 1\right]\boldsymbol{g}_{k,i}e_k(i)$

$\boldsymbol{w}_{k,i} = \sum_{m\in\mathcal{N}_k} c_{m,k}\boldsymbol{\psi}_{m,i}$

DNC method:

Step 1: to compute $\Delta_k(i)$

if $i = nV_t$, $n = 0, 1, 2, \ldots$

$\boldsymbol{a}^T_{k,i} = \mathcal{O}\left(\left[\frac{e^2_k(i)}{\|\boldsymbol{u}_{k,i}\|^2_2}, \frac{e^2_k(i-1)}{\|\boldsymbol{u}_{k,i-1}\|^2_2}, \ldots, \frac{e^2_k(i-V_t+1)}{\|\boldsymbol{u}_{k,i-V_t+1}\|^2_2}\right]\right)$

$\Theta_{\text{new},k} = \sum_{m\in\mathcal{N}_k} c_{m,k}\frac{\boldsymbol{a}^T_{m,i}\boldsymbol{e}}{V_t - V_d}$

$\Delta_k(i) = \frac{\Theta_{\text{new},k} - \Theta_{\text{old},k}}{\xi_k(i-1)}$

end

Step 2: to reset $\xi_k(i)$

if $\Delta_k(i) > t_{\text{th}}$

$\zeta_k(i) = \xi_k(0)$, $\boldsymbol{P}_{k,i} = \boldsymbol{P}_{k,0}$

elseif $\Theta_{\text{new},k} > \Theta_{\text{old},k}$

$\zeta_k(i) = \xi_k(i-1) + (\Theta_{\text{new},k} - \Theta_{\text{old},k})$

else

$\zeta_k(i) = \beta\xi_k(i-1) + (1-\beta)\min\left[\|\boldsymbol{g}_{k,i}\|^2_2 e^2_k(i),\ \xi_k(i-1)\right]$

end

$\xi_k(i) = \sum_{m\in\mathcal{N}_k} c_{m,k}\zeta_m(i)$

$\Theta_{\text{old},k} = \Theta_{\text{new},k}$

a positive integer, and $\sigma^2_{d,k}$ and $\sigma^2_{u,k}$ are powers of signals $d_k(i)$ and $\boldsymbol{u}_{k,i}$, respectively. The proposed algorithm is shown in Table 1.

Remark: As can be seen from (10), the operation mode of the proposed algorithm is twofold. At time instant i, if $\|\boldsymbol{g}_{k,i}\|^2_2 e^2_k(i) \leq \xi_k(i-1)$, the RLS update is performed; if not, the RLS update is normalized to have a norm of value $\xi_k(i-1)$. At the early iterations, the values of $\xi_k(i)$ can be high compared to $\|\boldsymbol{g}_{k,i}\|^2_2 e^2_k(i)$ so that the algorithm will behave as the dRLS algorithm, providing a fast convergence. Whenever an impulsive noise sample appears, due to its significant magnitude, the algorithm will work as an dRLS update multiplied by a very small 'step size' scaling factor given by $\sqrt{\xi_k(i-1)}/(\|\boldsymbol{g}_{k,i}\|_2|e_k(i)|)$, thus suppressing the negative influence of impulsive noise on the estimation [23, 24] and reducing the error propagation effect. The algorithm robustness to impulsive noise is further maintained due to decreasing $\xi_k(i)$ over the iterations. This algorithm can be considered as an improved dRLS algorithm with an additional 'step size' scaling factor which is time-varying and lies between 1 and $\sqrt{\xi_k(i-1)}/(\|\boldsymbol{g}_{k,i}\|_2|e_k(i)|)$, as can be observed in (10).

3.2. DNC Method

Although the decreasing values of the sequence $\{\xi_k(i)\}$ prompt the R-dRLS algorithm more robust against impulsive noises, the algorithm also loses its tracking capability for a sudden change of the unknown vector $\boldsymbol{w}^o$. To improve the tracking capability, referring to the single-agent scenario [25], we also develop a diffusion-based DNC method summarized in Table 1. The DNC method includes two steps.

Firstly, a variable $\Delta_k(i)$ at node k is computed once for every V_t iterations, to judge whether the unknown vector has a change or not. In this step, $\boldsymbol{a}^T_{k,i} = \mathcal{O}\left(\left[\frac{e^2_k(i)}{\|\boldsymbol{u}_{k,i}\|^2_2}, \frac{e^2_k(i-1)}{\|\boldsymbol{u}_{k,i-1}\|^2_2}, \ldots, \frac{e^2_k(i-V_t+1)}{\|\boldsymbol{u}_{k,i-V_t+1}\|^2_2}\right]\right)$ with $\mathcal{O}(\cdot)$ denoting the ascending arrangement for its arguments, and $\boldsymbol{e} = [1, \ldots, 1, 0, \ldots, 0]^T$ is a vector whose first $V_t - V_d$ elements set to one, where V_d is a positive integer with $V_d < V_t$. Thus, the product $\boldsymbol{a}^T_{k,i}\boldsymbol{e}$ can reduce the effect of outliers (e.g., impulsive noise samples) when computing $\Delta_k(i)$. Typically, for both V_t and V_d, good choices are $V_t = \varrho M$ with $\varrho = 1 \sim 3$ and $V_d = 0.75V_t$ [25]. Note that, for larger occurence probability of impulsive noise, the value of $V_t - V_d$ should be decreased to discard the impulsive noise samples.

Secondly, if $\Delta_k(i) > t_{\text{th}}$, where t_{th} is a predefined threshold, meaning a change of $\boldsymbol{w}^o$ has occured, then we need to reset $\xi_k(i)$ to its initial value $\xi_k(0)$. More importantly, $\boldsymbol{P}_{k,i}$ is also re-initialized with $\boldsymbol{P}_{k,0}$. It is worth noting that since the parameters γ, N_w, ϱ, and t_{th} are not affected by each other, their choices are simplified.

4. SIMULATION RESULTS

Simulation examples are presented for a diffusion network with $N = 20$ nodes. The vector $\boldsymbol{w}^o$ to be estimated has a length of $M = 16$ and a unit norm; it is generated randomly from a zero-mean uniform distribution. To evaluate the tracking capability, $\boldsymbol{w}^o$ changes to $-\boldsymbol{w}^o$ in the middle of iterations. The input regressor $\boldsymbol{u}_{k,i}$ has a shifted structure, i.e., $\boldsymbol{u}_{k,i} = [u_k(i), u_k(i-1), \ldots, u_k(i-M+1)]^T$ [4, 26], where $u_k(i)$ is colored and generated by a second-order autoregressive system:

$$u_k(i) = 1.6u_k(i-1) - 0.81u_k(i-2) + \epsilon_k(i),$$

where $\epsilon_k(i)$ is a zero-mean white Gaussian process with variance $\sigma^2_{\epsilon,k}$ shown in Fig. 1(a) for all the nodes. We employ the averaged network mean square deviation (MSD) to assess the performance of the algorithm, i.e., $\text{MSD}_{\text{net}}(i) = \frac{1}{N}\sum_{k=1}^{N} E\{\|\boldsymbol{w}^o - \boldsymbol{w}_{k,i}\|^2_2\}$, where $E\{\cdot\}$ denotes the expectation. Usually, the impulsive noise can be described by either the Bernoulli-Gaussian (BG) process [15–17] or the α-Stable process [21,27]. We consider both the cases. All results are the average over 200 independent trials[2].

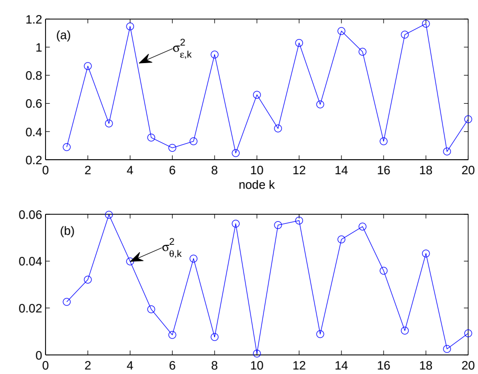

Fig. 1. Profiles of $\sigma^2_{\epsilon,k}$ and $\sigma^2_{\theta,k}$.

4.1. BG Process

The additive noise $v_k(i)$ includes the background noise $\theta_k(i)$ plus the impulsive noise $\eta_k(i)$, where $\theta_k(i)$ is zero-mean white Gaussian

[2] Here, for a fair comparison, the diffusion algorithms do not consider information exchange in the adaptation step, except the RVWC-dLMS.

noise with variance $\sigma^2_{\theta,k}$ depicted in Fig. 1(b). The impulsive noise $\eta_k(i)$ is described by the BG process, $\eta_k(i) = b_k(i) \cdot g_k(i)$, where $b_k(i)$ is a Bernoulli process with probability distribution $P[b_k(i) = 1] = p_{r,k}$ and $P[b_k(i) = 0] = 1 - p_{r,k}$, and $g_k(i)$ is a zero-mean white Gaussian process with variance $\sigma^2_{g,k}$. Here, we set $p_{r,k}$ as a random number in the range of $[0.001, 0.05]$, and $\sigma^2_{g,k} = 1000\sigma^2_{y,k}$, where $\sigma^2_{y,k}$ denotes the power of $y_k(i) = \boldsymbol{u}^T_{k,i}\boldsymbol{w}^o$. Fig. 2 compares the performance of the dRLS, dSE-LMS, and RVWC-dLMS algorithms with that of the proposed R-dRLS algorithm. Note that, the R-dRLS (no cooperation) algorithm performs an independent estimation at each node as presented in [19]. For RLS-type algorithms, we choose λ=0.995 and δ=0.01. As expected, the dRLS algorithm has a poor performance in the presence of impulsive noise. Both the dSE-LMS and RVWC-dLMS algorithms are significantly less sensitive to impulsive noise, but their convergence is slow. Apart from the robustness against impulsive noise, the proposed R-dRLS algorithm has also a fast convergence. Moreover, the proposed DNC method can retain the good tracking capability of the R-dRLS algorithm, only with a slight degradation in steady-state performance.

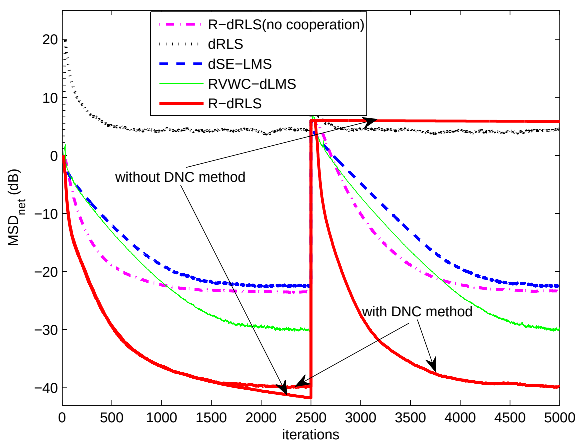

Fig. 2. Averaged network MSD performance of the algorithms in impulsive noise with BG process. Parameter setting of the algorithms (with notations from references) is as follows: μ_k=0.015 (dSE-LMS); β=0.98 and E_c=1 (R-dRLS); ϱ=3 and t_{th}=25 (DNC). For the RVWC-dLMS, the Metropolis rule [22] is also used for the combination coefficients in the adaptation step; its other parameters are L=16, α=2.58, λ=0.98 and μ_k=0.03.

4.2. α-Stable Process

The impulsive noise is now modeled by the α-stable process with a characteristic function $\varphi(t) = \exp(-\gamma|t|^\alpha)$, where the characteristic exponent $\alpha \in (0, 2]$ describes the impulsiveness of the noise (smaller α leads to more impulsive noise samples) and $\gamma > 0$ represents the dispersion level of the noise. In particular, when $\alpha = 2$, it reduces to the Gaussian noise. It is rare to find α-stable noise with $\alpha < 1$ in practice [21, 28]. In this example, thus we set $\alpha = 1.15$ and $\gamma = 1/15$. The learning performance of the algorithms is shown in Fig. 3. Fig. 4 shows the node-wise steady-state MSD of the robust algorithms (i.e., excluding the dRLS) against impulsive noise, by averaging over 500 instantaneous MSD values in the steady-state. As can be seen from Figs. 3 and 4, the proposed R-dRLS algorithm with DNC outperforms the known robust algorithms.

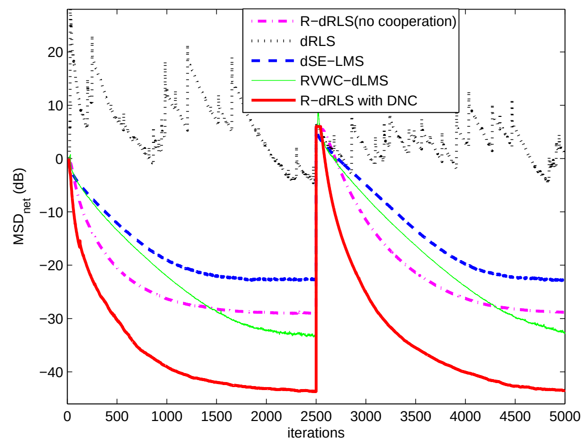

Fig. 3. Averaged network MSD performance of the algorithms in α-Stable noise. Parameter setting is the same as in Fig. 2.

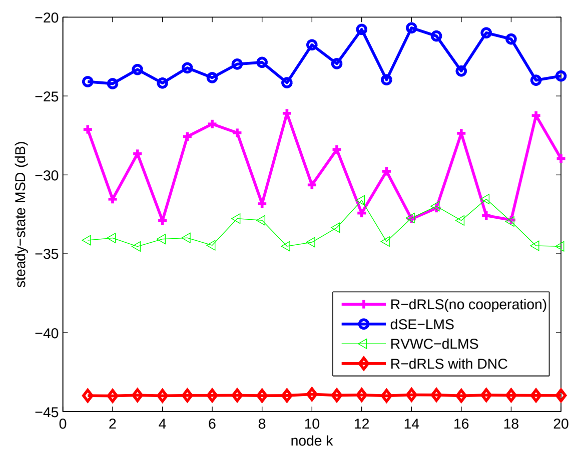

Fig. 4. Node-wise steady-state MSD of the algorithms in α-Stable noise.

5. CONCLUSION

In this paper, the R-dRLS algorithm has been proposed, based on the minimization of an individual RLS cost function with a time-dependent constraint on the squared norm of the intermediate estimate update. The constraint is dynamically adjusted based on the diffusion strategy with the help of side information. The novel algorithm not only is robust against impulsive noise, but also has fast convergence. Furthermore, to track the change of parameters of interest, a detection method (DNC method) is proposed for re-initializing the constraint. Simulation results have verified that the proposed algorithm performs better than known algorithms in impulsive noise scenarios.

6. REFERENCES

[1] A.H. Sayed, "Adaptation, learning, and optimization over networks," *Foundations and Trends in Machine Learning*, vol. 7, no. 4-5, pp. 311–801, 2014.

[2] S. Kanna, D.H. Dini, Y. Xia, S.Y. Hui, and D.P. Mandic, "Distributed widely linear kalman filtering for frequency estimation in power networks," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 1, no. 1, pp. 45–57, 2015.

[3] T.G. Miller, S. Xu, R.C. de Lamare, and H.V. Poor, "Distributed spectrum estimation based on alternating mixed discrete-continuous adaptation," *IEEE Signal Processing Letters*, vol. 23, no. 4, pp. 551–555, 2016.

[4] L. Li, J.A. Chambers, C.G. Lopes, and A.H. Sayed, "Distributed estimation over an adaptive incremental network based on the affine projection algorithm," *IEEE Transactions on Signal Processing*, vol. 58, no. 1, pp. 151–164, 2010.

[5] S.Y. Tu and A.H. Sayed, "Diffusion strategies outperform consensus strategies for distributed estimation over adaptive networks," *IEEE Transactions on Signal Processing*, vol. 60, no. 12, pp. 6217–6234, 2012.

[6] Y. Yu, R.C de Lamare, Y. Zakharov, and H. Zhao, "Distributed constrained consensus least-mean square algorithms with adjustable constraints," in *22nd International Conference on Digital Signal Processing (DSP),*. IEEE, 2017, pp. 1–5.

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

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

[9] J. Chen and A.H. Sayed, "Diffusion adaptation strategies for distributed optimization and learning over networks," *IEEE Transactions on Signal Processing*, vol. 60, no. 8, pp. 4289–4305, 2012.

[10] H.S. Lee, S.E. Kim, J.W. Lee, and W.J. Song, "A variable step-size diffusion LMS algorithm for distributed estimation.," *IEEE Trans. Signal Processing*, vol. 63, no. 7, pp. 1808–1820, 2015.

[11] S. Xu, R.C de Lamare, and H.V. Poor, "Adaptive link selection algorithms for distributed estimation," *EURASIP Journal on Advances in Signal Processing*, vol. 2015, no. 1, pp. 86, 2015.

[12] S. Xu, R.C de Lamare, and H.V. Poor, "Distributed compressed estimation based on compressive sensing," *IEEE Signal Processing Letters*, vol. 22, no. 9, pp. 1311–1315, 2015.

[13] Z. Liu, Y. Liu, and C. Li, "Distributed sparse recursive least-squares over networks.," *IEEE Trans. Signal Processing*, vol. 62, no. 6, pp. 1386–1395, 2014.

[14] K.L. Blackard, T.S. Rappaport, and C.W. Bostian, "Measurements and models of radio frequency impulsive noise for indoor wireless communications," *IEEE Journal on selected areas in communications*, vol. 11, no. 7, pp. 991–1001, 1993.

[15] J. Ni, J. Chen, and X. Chen, "Diffusion sign-error LMS algorithm: Formulation and stochastic behavior analysis," *Signal Processing*, vol. 128, pp. 142–149, 2016.

[16] D.C. Ahn, J.W. Lee, S.J. Shin, and W.J. Song, "A new robust variable weighting coefficients diffusion LMS algorithm," *Signal Processing*, vol. 131, pp. 300–306, 2017.

[17] S. Al-Sayed, A.M. Zoubir, and A.H. Sayed, "Robust distributed estimation by networked agents," *IEEE Transactions on Signal Processing*, vol. 65, no. 15, pp. 3909–3921, Aug. 2017.

[18] S. Kumar, U.K. Sahoo, A.K. Sahoo, and D.P. Acharya, "Diffusion minimum-wilcoxon-norm over distributed adaptive networks: Formulation and performance analysis," *Digital Signal Processing*, vol. 51, pp. 156–169, 2016.

[19] L.R. Vega, H. Rey, J. Benesty, and S. Tressens, "A fast robust recursive least-squares algorithm," *IEEE Transactions on Signal Processing*, vol. 57, no. 3, pp. 1209–1216, 2009.

[20] A.H. Sayed, *Adaptive filters*, John Wiley & Sons, 2011.

[21] K. Pelekanakis and M. Chitre, "Adaptive sparse channel estimation under symmetric alpha-stable noise," *IEEE Transactions on wireless communications*, vol. 13, no. 6, pp. 3183–3195, 2014.

[22] N. Takahashi, I. Yamada, and A.H. Sayed, "Diffusion least-mean squares with adaptive combiners: Formulation and performance analysis," *IEEE Transactions on Signal Processing*, vol. 58, no. 9, pp. 4795–4810, 2010.

[23] I. Song, P.G. Park, and R.W. Newcomb, "A normalized least mean squares algorithm with a step-size scaler against impulsive measurement noise," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 60, no. 7, pp. 442–445, 2013.

[24] J. Hur, I. Song, and P.G. Park, "A variable step-size normalized subband adaptive filter with a step-size scaler against impulsive measurement noise," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 64, no. 7, pp. 842–846, 2017.

[25] L.R. Vega, H. Rey, J. Benesty, and S. Tressens, "A new robust variable step-size NLMS algorithm," *IEEE Transactions on Signal Processing*, vol. 56, no. 5, pp. 1878–1893, 2008.

[26] S. Chouvardas, K. Slavakis, and S. Theodoridis, "Adaptive robust distributed learning in diffusion sensor networks," *IEEE Transactions on Signal Processing*, vol. 59, no. 10, pp. 4692–4707, 2011.

[27] L. Lu, H. Zhao, and B. Chen, "Improved-variable-forgetting-factor recursive algorithm based on the logarithmic cost for volterra system identification," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 63, no. 6, pp. 588–592, 2016.

[28] M. Shao and C.L. Nikias, "Signal processing with fractional lower order moments: stable processes and their applications," *Proceedings of the IEEE*, vol. 81, no. 7, pp. 986–1010, 1993.