STEADY-STATE PERFORMANCE ANALYSIS OF THE DISTRIBUTED RLS ALGORITHM

Gonzalo Mateos and Georgios B. Giannakis

Dept. of ECE and Digital Technology Center, Univ. of Minnesota,
200 Union Street SE, Minneapolis, MN 55455, USA

ABSTRACT

The recursive least-squares (RLS) algorithm offers both reduced complexity and limited memory requirements when it comes to learning from data acquired sequentially in time. The focus of this paper is on analyzing the performance of a *distributed* recursive least-squares (D-RLS) algorithm, suitable for *online* learning from network data. A steady-state mean-square error (MSE) analysis of D-RLS is conducted, by studying a stochastically-driven 'averaged' system that approximates the D-RLS dynamics asymptotically in time. For observations that are linearly related to the time-invariant parameter vector sought, the simplifying independence setting assumptions facilitate deriving accurate closed-form expressions for the MSE limiting values. The problems of mean and MSE-sense stability of D-RLS are also investigated, and easily-checkable sufficient conditions are derived under which a steady-state is attained. Computer simulations demonstrate that the upshot of the analysis extends accurately to the pragmatic setting where the underlying network processes exhibit temporal correlations.

Index Terms— Distributed learning, online learning, RLS algorithm, performance analysis.

1. INTRODUCTION

The explosion of *network data* has created renewed interest in the field of distributed signal and information processing over graphs, calling for collaborative solutions that enable real-time estimation of stationary signals as well as reduced-complexity tracking of nonstationary network processes. In this context, the focus of this paper is on analyzing the performance of a *distributed* recursive least-squares (D-RLS) algorithm, suitable for *online* learning from network data [9]. In D-RLS a two-step iterative process takes place towards consenting on the desired global exponentially-weighted least-squares estimator (EWLSE) [1, 2]: network agents carry out reduced-complexity tasks locally, and exchange messages with one-hop neighbors to consent on the network-wide estimates adaptively (Section 2). Network data acquired in real time enrich the estimation process and learn the unknown statistics on-the-fly.

A detailed stability and MSE steady-state (s.s.) *performance analysis* is conducted for D-RLS. Evaluating the performance of (centralized) online learning algorithms is a challenging problem in its own right; prior art from an adaptive filtering vantage point is surveyed in e.g., [1, 2], and the extensive list of references therein. On top of that, a networked setting introduces unique challenges in the analysis such as heterogeneous spatio-temporal data profiles and multiple sources of randomness, a consequence of e.g., unmodeled complex dynamics and imperfect communication links. The approach pursued here capitalizes on an 'averaged' error-form representation of the local recursions comprising D-RLS, as a global dynamical system described by a stochastic difference-equation derived in Section 3.2. Somehow related approaches were adopted in [3] and [4]. Other noteworthy analysis techniques include the energy-conservation methodology in [5], [2, p. 287], and stochastic averaging [1, p. 229]. For performance analysis of *distributed* online learning algorithms, the former has been applied in e.g., [6], while the latter can be found in [7].

The covariance matrix of the resulting state is shown to encompass all the information needed to evaluate the relevant network-wide and per-agent performance metrics (Section 3.3). For observations that are linearly related to the time-invariant parameter vector sought, the simplifying independence setting assumptions [1, pg. 110], [2, pg. 448] are key enablers towards deriving accurate closed-form expressions for the mean-square deviation and excess-MSE s.s. values (Section 4.2). Stability in the mean- and MSE-sense are also investigated, revealing easily-checkable sufficient conditions under which a s.s. is attained. Numerical tests corroborating the theoretical findings are presented in Section 5.

Notation: Operators $\otimes$, $(.)^T$, $(.)^\dagger$, $\lambda_{\max}(.)$, $\text{tr}(.)$, $\text{diag}(.)$, $\text{bdiag}(.)$, $E[.]$, will denote Kronecker product, transposition, matrix pseudo-inverse, spectral radius, matrix trace, diagonal matrix, block diagonal matrix, and expectation, respectively. For both vectors and matrices, $\|.\|$ will stand for the $2-$norm. The $n \times n$ identity matrix will be represented by $\mathbf{I}_n$, while $\mathbf{1}_n$ will denote the $n \times 1$ vector of all ones and $\mathbf{1}_{n\times m} := \mathbf{1}_n\mathbf{1}_m^T$. Similar notation will be adopted for matrices of all zeros.

Work was supported by an AFOSR MURI grant no. FA 9550-10-1-0567.

978-1-4673-1026-0/12/$31.00 ©2012 IEEE

2. DISTRIBUTED RECURSIVE LEAST-SQUARES

Consider a network of interconnected agents $\{1,\ldots,J\} := \mathcal{J}$, naturally modeled as an undirected connected graph with associated Laplacian matrix $\mathbf{L}$. Agent $j \in \mathcal{J}$ is capable of performing some local computations, as well as exchanging messages with its directly connected neighbors in $\mathcal{N}_j \subseteq \mathcal{J}$. Different from [3, 4] and [8], the present network model accounts explicitly for imperfect exchanges of information among agents. Specifically, messages received at agent j from agent i at discrete-time instant t are corrupted by a zero-mean additive noise vector $\boldsymbol{\eta}_j^i(t)$, assumed temporally and spatially uncorrelated. The communication noise covariance matrices are denoted by $\mathbf{R}_{\eta_j} := E[\boldsymbol{\eta}_j^i(t)(\boldsymbol{\eta}_j^i(t))^T]$, $j \in \mathcal{J}$.

The network infrastructure is utilized to estimate a parameter vector $\mathbf{s}_0 \in \mathbb{R}^{p\times 1}$ in a distributed fashion and subject to the single-hop communication constraints, by resorting to the least-squares (LS) criterion. Per time instant $t = 0,1,\ldots$, each agent acquires a regression vector $\mathbf{h}_j(t) \in \mathbb{R}^{p\times 1}$ and a scalar observation $x_j(t)$, both assumed zero-mean without loss of generality. Given new data sequentially acquired, a pertinent approach to online learning is the EWLSE [2, 3, 8]

$$\hat{\mathbf{s}}_{\text{ewls}}(t) := \arg\min_{\mathbf{s}} \sum_{\tau=0}^{t}\sum_{j=1}^{J} \lambda^{t-\tau}\left[x_j(\tau) - \mathbf{h}_j^T(\tau)\mathbf{s}\right]^2 \quad (1)$$

where $\lambda \in (0,1]$ is a forgetting factor. A strictly-convex term $\lambda^t\delta\|\mathbf{s}\|^2$ is typically included in (1) for regularization purposes, where δ is a large positive constant. Note that in forming $\hat{\mathbf{s}}_{\text{ewls}}(t)$ the entire history of data $\{x_j(\tau),\mathbf{h}_j(\tau)\}_{\tau=0}^{t}$, $\forall\ j \in \mathcal{J}$ is incorporated in the online estimation process. When $\lambda < 1$, past data are exponentially discarded thus enabling tracking of nonstationary network processes.

The (centralized) estimator (1) is not amenable for distributed implementation since the global variable $\mathbf{s}$ couples the per-agent summands. A distributed algorithm was put forth in [9] by reformulating the EWLSE into an equivalent constrained form, which can be minimized in a distributed fashion by resorting to the alternating-minimization algorithm (AMA) [10]. The algorithmic construction details can be found in [9, Sec. II]. Accounting for additive communication noise that corrupts the exchanges of Lagrange multipliers $\{\mathbf{v}_j^{j'}\}_{j\in\mathcal{N}_j}^{j'\in\mathcal{N}_j}$ and local estimates $\{\mathbf{s}_j\}_{j\in\mathcal{J}}$ through the vectors $\bar{\boldsymbol{\eta}}_j^{j'}(t)$ and $\boldsymbol{\eta}_j^{j'}(t)$, respectively, the per agent tasks comprising the AMA-based D-RLS algorithm are given by [9]

$$\mathbf{v}_j^{j'}(t) = \mathbf{v}_j^{j'}(t-1) + \frac{c}{2}\left[\mathbf{s}_j(t) - (\mathbf{s}_{j'}(t) + \boldsymbol{\eta}_j^{j'}(t))\right] \quad (2)$$

$$\boldsymbol{\Phi}_j^{-1}(t+1) = \lambda^{-1}\boldsymbol{\Phi}_j^{-1}(t) - \frac{\lambda^{-1}\boldsymbol{\Phi}_j^{-1}(t)\mathbf{h}_j(t+1)\mathbf{h}_j^T(t+1)\boldsymbol{\Phi}_j^{-1}(t)}{\lambda + \mathbf{h}_j^T(t+1)\boldsymbol{\Phi}_j^{-1}(t)\mathbf{h}_j(t+1)} \quad (3)$$

Algorithm 1 : D-RLS at agent j

Arbitrarily initialize $\mathbf{s}_j(0)$ and $\{\mathbf{v}_j^{j'}(-1)\}_{j'\in\mathcal{N}_j}$.
for $t = 0, 1, \ldots$ **do**
  Transmit $\mathbf{s}_j(t)$ to neighbors in $\mathcal{N}_j$.
  Update $\{\mathbf{v}_j^{j'}(t)\}_{j'\in\mathcal{N}_j}$ using (2).
  Transmit $\mathbf{v}_j^{j'}(t)$ to each $j' \in \mathcal{N}_j$.
  Update $\boldsymbol{\Phi}_j(t+1)$ and $\boldsymbol{\psi}_j(t+1)$ using (3) and (4).
  Update $\mathbf{s}_j(t+1)$ using (5).
end for

$$\boldsymbol{\psi}_j(t+1) = \lambda\boldsymbol{\psi}_j(t) + \mathbf{h}_j(t+1)x_j(t+1) \quad (4)$$

$$\mathbf{s}_j(t+1) = \boldsymbol{\Phi}_j^{-1}(t+1)\boldsymbol{\psi}_j(t+1) - \frac{1}{2}\boldsymbol{\Phi}_j^{-1}(t+1)\sum_{j'\in\mathcal{N}_j}\left[\mathbf{v}_j^{j'}(t) - (\mathbf{v}_{j'}^{j}(t) + \bar{\boldsymbol{\eta}}_j^{j'}(t))\right]. \quad (5)$$

Note that $j' \in \mathcal{N}_j$ in the dual-ascent iterations (2), while $c > 0$ is a constant step-size. In addition, the per-agent exponentially-weighted data (cross-) correlations are $\boldsymbol{\psi}_j(t) := \sum_{\tau=0}^{t}\lambda^{t-\tau}\mathbf{h}_j(\tau)x_j(\tau)$ and $\boldsymbol{\Phi}_j(t) := \sum_{\tau=0}^{t}\lambda^{t-\tau}\mathbf{h}_j(\tau)\mathbf{h}_j^T(\tau)$.

Recursions (2)-(5) are tabulated as Algorithm 1, which also details the inter-agent communications of multipliers and local estimates taking place only within neighborhoods. If the inter-agent links can be rendered error-free, results in [11] show that D-RLS can be further simplified to reduce the communication overhead and memory storage requirements.

3. ANALYSIS PRELIMINARIES

3.1. Analysis scope: assumptions and approximations

The challenges in evaluating the performance of classical (centralized) LMS and RLS filters are well documented [2, 1], and results for RLS are less common and typically involve simplifying approximations. What is more, the distributed setting studied in this paper introduces unique challenges in the analysis. These include network data and multiple sources of additive noise, a consequence of unmodeled dynamics, imperfect data acquisition and communication links.

In order to proceed, a few modeling assumptions are introduced which delineate the scope of the ensuing stability and performance results. For all $j \in \mathcal{J}$, it is assumed that:

(a1) *Agent observations obey* $x_j(t) = \mathbf{h}_j^T(t)\mathbf{s}_0 + \epsilon_j(t)$, *where the zero-mean white noise* $\{\epsilon_j(t)\}$ *has variance* $\sigma_{\epsilon_j}^2$;

(a2) Vectors $\{\mathbf{h}_j(t)\}$ *are spatio-temporally white with positive-definite covariance matrix* $\mathbf{R}_{h_j}$; *and*

(a3) Vectors $\{\mathbf{h}_j(t)\}$, $\{\epsilon_j(t)\}$, $\{\boldsymbol{\eta}_j^{j'}(t)\}_{j'\in\mathcal{N}_j}$ *and* $\{\bar{\boldsymbol{\eta}}_j^{j'}(t)\}_{j'\in\mathcal{N}_j}$ *are statistically independent.*

Assumptions (a1)-(a3) comprise the widely adopted *independence setting*, for agent observations that are linearly related to the parameter vector of interest; see e.g., [1, pg. 110], [2, pg. 448]. In line with network-generated data, the statistical profiles of both regressors and the noise quantities vary across agents (space), yet they are assumed time invariant.

In the particular case of the D-RLS algorithm, a unique challenge stems from the stochastic matrices $\mathbf{\Phi}_j^{-1}(t)$ present in (5). Even obtaining $\mathbf{\Phi}_j^{-1}(t)$'s distribution or computing its expected value is a formidable task in general, due to the matrix inversion operation. For these reasons some simplifying approximations will be adopted to carry out the analysis that otherwise becomes intractable.

By definition, matrix $\mathbf{\Phi}_j(t)$ is obtained as an exponentially weighted moving average (EWMA) of local regressor outer products. The EWMA can be seen as an average modulated by a sliding window of equivalent length $1/(1-\lambda)$, which clearly grows as $\lambda \to 1$. This observation in along with (a2) and the strong law of large numbers, justifies the approximation

$$\mathbf{\Phi}_j(t) \approx E[\mathbf{\Phi}_j(t)] = \frac{\mathbf{R}_{h_j}}{1-\lambda}, \quad 0 \ll \lambda < 1 \text{ and } t \to \infty. \tag{6}$$

The expectation of $\mathbf{\Phi}_j^{-1}(t)$, on the other hand, is considerably harder to evaluate. To overcome this challenge, the following approximation will be invoked [2, 3]

$$E[\mathbf{\Phi}_j^{-1}(t)] \approx E[\mathbf{\Phi}_j(t)]^{-1} \approx (1-\lambda)\mathbf{R}_{h_j}^{-1} \tag{7}$$

for $0 \ll \lambda < 1$ and $t \to \infty$. It is a crude approximation at first sight. However, experimental evidence suggests that the approximation is sufficiently accurate for all practical purposes, when the forgetting factor approaches unity [2, p. 319].

3.2. Error-form D-RLS

The approach here to s.s. performance analysis relies on an 'averaged' error-form system representation of D-RLS in (2)-(5), where $\mathbf{\Phi}_j^{-1}(t)$ in (5) is replaced by the approximation $(1-\lambda)\mathbf{R}_{h_j}^{-1}$, for sufficiently large t.

Towards obtaining such error-form representation, introduce the local estimation errors $\{\mathbf{y}_{1,j}(t) := \mathbf{s}_j(t) - \mathbf{s}_0\}_{j=1}^J$ and multiplier-based quantities $\{\mathbf{y}_{2,j}(t) := \frac{1}{2}\sum_{j' \in \mathcal{N}_j}(\mathbf{v}_j^{j'}(t-1) - \mathbf{v}_{j'}^{j}(t-1))\}_{j=1}^J$. It turns out that a convenient global state to describe the spatio-temporal dynamics of D-RLS is $\mathbf{y}(t) := [\mathbf{y}_1^T(t)\ \mathbf{y}_2^T(t)]^T = [\mathbf{y}_{1,1}^T(t) \ldots \mathbf{y}_{1,J}^T(t)\ \mathbf{y}_{2,1}^T(t) \ldots \mathbf{y}_{2,J}^T(t)]^T \in \mathbb{R}^{2Jp}$. In addition, to concisely capture the effects of both observation and communication noise on the estimation errors across the network, define the $Jp \times 1$ noise supervectors $\boldsymbol{\epsilon}(t) := \sum_{\tau=0}^{t} \lambda^{t-\tau}[\mathbf{h}_1^T(\tau)\epsilon_1(\tau) \ldots \mathbf{h}_J^T(\tau)\epsilon_J(\tau)]^T$ and $\bar{\boldsymbol{\eta}}(t) := [\bar{\boldsymbol{\eta}}_1^T(t) \ldots \bar{\boldsymbol{\eta}}_J^T(t)]^T$. Vectors $\{\bar{\boldsymbol{\eta}}_j(t)\}_{j=1}^J$ represent the aggregate noise corrupting the multipliers received by agent j at time instant t, and are given by $\bar{\boldsymbol{\eta}}_j(t) := \sum_{j' \in \mathcal{N}_j} \bar{\boldsymbol{\eta}}_j^{j'}(t)/2$. Their respective covariance matrices are easily computable under (a2)-(a3). For instance,

$$\mathbf{R}_{\boldsymbol{\epsilon}}(t) := \left(\frac{1-\lambda^{2(t+1)}}{1-\lambda^2}\right) \text{bdiag}(\mathbf{R}_{h_1}\sigma_{\epsilon_1}^2, \ldots, \mathbf{R}_{h_J}\sigma_{\epsilon_J}^2)$$

while the structure of $\mathbf{R}_{\bar{\boldsymbol{\eta}}} := E[\bar{\boldsymbol{\eta}}(t)\bar{\boldsymbol{\eta}}^T(t)]$ can be found in [9, App. E]. In addition, introduce the $p(\sum_{j=1}^J |\mathcal{N}_j|) \times 1$ vector $\boldsymbol{\eta}(t) := \left[\{(\boldsymbol{\eta}_{j'}^1(t))^T\}_{j' \in \mathcal{N}_1} \ldots \{(\boldsymbol{\eta}_{j'}^J(t))^T\}_{j' \in \mathcal{N}_J}\right]^T$, which comprises the receiver noise terms corrupting transmissions of local estimates across the whole network at time instant t, and define $\mathbf{R}_{\boldsymbol{\eta}} := E[\boldsymbol{\eta}(t)\boldsymbol{\eta}^T(t)]$.

Finally, let $\mathbf{L}_c := (c/2)\mathbf{L} \otimes \mathbf{I}_p \in \mathbb{R}^{Jp \times Jp}$ be a matrix capturing network topology through the (scaled) graph Laplacian matrix $\mathbf{L}$, and define $\mathbf{R}_{h,\lambda}^{-1} := (1-\lambda)\text{bdiag}(\mathbf{R}_{h_1}^{-1}, \ldots, \mathbf{R}_{h_J}^{-1})$. It is now possible to state the following important lemma[1].

Lemma 1: *Let (a1) and (a2) hold. Then for $t \geq t_0$ with t_0 sufficiently large while $0 \ll \lambda < 1$, the global state $\mathbf{y}(t)$ approximately evolves according to*

$$\mathbf{y}(t+1) = \textit{bdiag}(\mathbf{I}_{Jp}, \mathbf{L}_c)\mathbf{z}(t+1) + \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} \bar{\boldsymbol{\eta}}(t) + \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1}(\mathbf{P}_\alpha - \mathbf{P}_\beta) \\ \mathbf{P}_\beta - \mathbf{P}_\alpha \end{bmatrix} \boldsymbol{\eta}(t). \tag{8}$$

The inner state $\mathbf{z}(t) := [\mathbf{z}_1^T(t)\ \mathbf{z}_2^T(t)]^T$ is arbitrarily initialized at time t_0, and updated according to

$$\mathbf{z}(t+1) = \mathbf{\Psi}\mathbf{z}(t) + \mathbf{\Psi}\begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1}(\mathbf{P}_\alpha - \mathbf{P}_\beta) \\ \mathbf{C} \end{bmatrix} \boldsymbol{\eta}(t-1) + \mathbf{\Psi}\begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} \bar{\boldsymbol{\eta}}(t-1) + \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} \boldsymbol{\epsilon}(t+1) \tag{9}$$

and the $2Jp \times 2Jp$ transition matrix $\mathbf{\Psi}$ consists of the blocks $[\mathbf{\Psi}]_{11} = [\mathbf{\Psi}]_{12} = -\mathbf{R}_{h,\lambda}^{-1}\mathbf{L}_c$ and $[\mathbf{\Psi}]_{21} = [\mathbf{\Psi}]_{22} = \mathbf{L}_c\mathbf{L}_c^\dagger$. Matrix $\mathbf{C}$ is chosen such that $\mathbf{L}_c\mathbf{C} = \mathbf{P}_\beta - \mathbf{P}_\alpha$, where the structure of the time-invariant matrices $\mathbf{P}_\alpha$ and $\mathbf{P}_\beta$ can be found in [9, App. E].

The desired state $\mathbf{y}(t)$ is obtained as a rank-deficient linear transformation of the inner state $\mathbf{z}(t)$, plus a stochastic offset due to the effects of communication noise. A linear, time-invariant, first-order difference equation describes the dynamics of $\mathbf{z}(t)$, and hence of $\mathbf{y}(t)$, via the algebraic transformation in (8). The time-invariant nature of the transition matrix $\mathbf{\Psi}$ is due to the approximations $\mathbf{\Phi}_j^{-1}(t) \approx \mathbf{R}_{h,\lambda}^{-1}$, $j \in \mathcal{J}$, particularly accurate for large enough $t > t_0$. Examination of (9) reveals that the evolution of $\mathbf{z}(t)$ is driven by three stochastic input processes: i) communication noise $\boldsymbol{\eta}(t-1)$ affecting the transmission of local estimates; ii) communication noise $\bar{\boldsymbol{\eta}}(t-1)$ contaminating the Lagrange multipliers; and iii) observation noise within $\boldsymbol{\epsilon}(t+1)$.

[1] Proofs are omitted here due to lack of space, but can be found in [9]

Table 1. Evaluation of local and global figures of merit from $\mathbf{R}_y(t)$

	MSD	**EMSE**	**MSE**
Local	$\text{tr}([\mathbf{R}_y(t)]_{11,j})$	$\text{tr}(\mathbf{R}_{h_j}[\mathbf{R}_y(t-1)]_{11,j})$	$\text{tr}(\mathbf{R}_{h_j}[\mathbf{R}_y(t-1)]_{11,j}) + \sigma^2_{\epsilon_j}$
Global	$J^{-1}\text{tr}([\mathbf{R}_y(t)]_{11})$	$J^{-1}\text{tr}(\mathbf{R}_h[\mathbf{R}_y(t-1)]_{11})$	$J^{-1}\text{tr}(\mathbf{R}_h[\mathbf{R}_y(t-1)]_{11}) + J^{-1}\sum_{j=1}^{J}\sigma^2_{\epsilon_j}$

3.3. Performance Metrics

When it comes to performance evaluation of adaptive algorithms, it is customary to consider as figures of merit the so-called MSE, excess mean-square error (EMSE) and mean-square deviation (MSD) [2], [1]. In the present setup for distributed online learning, it is pertinent to address both global (network-wide) and local (per-agent) performance [6]. After recalling the definitions of the local a priori error $e_j(t) := x_j(t) - \mathbf{h}_j^T(t)\mathbf{s}_j(t-1)$ and local estimation error $\mathbf{y}_{1,j}(t) := \mathbf{s}_j(t) - \mathbf{s}_0$, the per-agent performance metrics are defined as $\text{MSE}_j(t) := E[e_j^2(t)]$, $\text{EMSE}_j(t) := E[(\mathbf{h}_j^T(t)\mathbf{y}_{1,j}(t-1))^2]$, and $\text{MSD}_j(t) := E[\|\mathbf{y}_{1,j}(t)\|^2]$. Their global counterparts are defined as the respective averages across agents, e.g., $\text{MSE}(t) := J^{-1}\sum_{j=1}^{J} E[e_j(t)^2]$ and so on.

Next, it is shown that it suffices to evaluate the state covariance matrix $\mathbf{R}_y(t) := E[\mathbf{y}(t)\mathbf{y}^T(t)]$ in order to assess the aforementioned performance metrics. Under (a1) it is possible to write $e_j(t) = -\mathbf{h}_j^T(t)\mathbf{y}_{1,j}(t-1) + \epsilon_j(t)$. Because $\mathbf{y}_{1,j}(t-1)$ is independent of the zero-mean $\{\mathbf{h}_j(t), \epsilon_j(t)\}$ under (a1)-(a3), from the previous relationship between the a priori and estimation errors one finds that $\text{MSE}_j(t) = \text{EMSE}_j(t) + \sigma^2_{\epsilon_j}$. Hence, it suffices to focus on the evaluation of $\text{EMSE}_j(t)$, through which $\text{MSE}_j(t)$ can also be determined under the assumption that the observation noise variances are known, or can be estimated for that matter. If $\mathbf{R}_{y_{1,j}}(t) := E[\mathbf{y}_{1,j}(t)\mathbf{y}_{1,j}^T(t)]$ denotes the j-th local error covariance matrix, then $\text{MSD}_j(t) = \text{tr}(\mathbf{R}_{y_{1,j}}(t))$; and under (a1)-(a3), a simple manipulation yields $\text{EMSE}_j(t) = \text{tr}(\mathbf{R}_{h_j}\mathbf{R}_{y_{1,j}}(t-1))$. To derive corresponding formulas for the global performance metrics, let $\mathbf{R}_{y_1}(t) := E[\mathbf{y}_1(t)\mathbf{y}_1^T(t)]$ denote the global error covariance matrix, and define $\mathbf{R}_h := E[\mathbf{R}_h(t)] = \text{bdiag}(\mathbf{R}_{h_1}, \ldots, \mathbf{R}_{h_J})$. It follows that $\text{MSD}(t) = J^{-1}\text{tr}(\mathbf{R}_{y_1}(t))$, and $\text{EMSE}(t) = J^{-1}\text{tr}(\mathbf{R}_h\mathbf{R}_{y_1}(t-1))$.

It is now apparent that $\mathbf{R}_y(t)$ indeed provides all the information needed to evaluate the performance of the D-RLS algorithm. For instance, the global error covariance matrix $\mathbf{R}_{y_1}(t)$ corresponds to the $Jp \times Jp$ upper left submatrix of $\mathbf{R}_y(t)$, which is denoted by $[\mathbf{R}_y(t)]_{11}$. Further, the j-th $p \times p$ diagonal submatrix of $[\mathbf{R}_y(t)]_{11}$ is exactly $\mathbf{R}_{y_{1,j}}(t)$, and is likewise denoted by $[\mathbf{R}_y(t)]_{11,j}$. In a nutshell, deriving a closed-form expression for $\mathbf{R}_y(t)$ enables the evaluation of all performance metrics of interest, the subject of Section 4.2.

4. STABILITY AND STEADY-STATE PERFORMANCE ANALYSIS

In this section, stability and s.s. performance analyses are conducted for the D-RLS algorithm outlined in Section 2. Because recursions (2)-(5) are stochastic in nature, stability will be assessed both in the mean and in the MSE-sense.

4.1. Mean Stability

Based on Lemma 1, it follows that D-RLS achieves consensus in the mean sense on the parameter $\mathbf{s}_0$.

Proposition 1: *Under (a1)-(a3) and for* $0 \ll \lambda < 1$, *the D-RLS algorithm achieves consensus in the mean, i.e.,* $\lim_{t\to\infty} E[\mathbf{y}_{1,j}(t)] = \mathbf{0}_p, \quad \forall\, j \in \mathcal{J}$ *provided the step-size is chosen such that*

$$0 < c < \frac{4}{(1-\lambda)\lambda_{\max}(\mathbf{R}_h^{-1}(\mathbf{L}\otimes\mathbf{I}_p))}. \qquad (10)$$

When $0 \ll \lambda < 1$, (10) is actually not restrictive at all since a $1-\lambda$ factor is present in the denominator. When λ is close to one, any practical choice of $c > 0$ will result in asymptotically unbiased local estimates. Also note that (10) depends on the network topology through $\mathbf{L}_c$.

4.2. MSE Stability and Steady-State Performance

In order to assess the s.s. MSE performance of the D-RLS algorithm, the figures of merit in Table 1 will be evaluated here. To this end, it suffices to derive a closed-form expression for the global estimation error covariance matrix $\mathbf{R}_{y_1}(t) := E[\mathbf{y}_1(t)\mathbf{y}_1^T(t)]$, as already argued in Section 3.3.

Observe from the upper $Jp \times 1$ block of $\mathbf{y}(t+1)$ in (8) that $\mathbf{y}_1(t+1) = \mathbf{z}_1(t+1) + \mathbf{R}_{h,\lambda}^{-1}[\bar{\boldsymbol{\eta}}(t) + (\mathbf{P}_\alpha - \mathbf{P}_\beta)\boldsymbol{\eta}(t)]$. Under (a3), $\mathbf{z}_1(t+1)$ is independent of $\{\bar{\boldsymbol{\eta}}(t), \boldsymbol{\eta}(t)\}$; hence,

$$\mathbf{R}_{y_1}(t) = \mathbf{R}_{z_1}(t) + \mathbf{R}_{h,\lambda}^{-1}\mathbf{R}_{\bar{\boldsymbol{\eta}}}\mathbf{R}_{h,\lambda}^{-1} + \mathbf{R}_{h,\lambda}^{-1}(\mathbf{P}_\alpha - \mathbf{P}_\beta)\mathbf{R}_{\boldsymbol{\eta}}(\mathbf{P}_\alpha - \mathbf{P}_\beta)^T\mathbf{R}_{h,\lambda}^{-1} \qquad (11)$$

which prompts one to obtain $\mathbf{R}_z(t) := E[\mathbf{z}(t)\mathbf{z}^T(t)]$. Specifically, the goal is to extract its upper-left $Jp \times Jp$ matrix block $[\mathbf{R}_z(t)]_{11} = \mathbf{R}_{z_1}(t)$. To this end, define the vectors

$$\bar{\boldsymbol{\eta}}_\lambda(t) := \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp\times Jp} \end{bmatrix} \bar{\boldsymbol{\eta}}(t)$$

$$\boldsymbol{\eta}_\lambda(t) := \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1}(\mathbf{P}_\alpha - \mathbf{P}_\beta) \\ \mathbf{C} \end{bmatrix} \boldsymbol{\eta}(t) \qquad (12)$$

with respective covariance matrices $\mathbf{R}_{\bar{\boldsymbol{\eta}}_\lambda} := E[\bar{\boldsymbol{\eta}}_\lambda(t)\bar{\boldsymbol{\eta}}_\lambda^T(t)]$ and $\mathbf{R}_{\boldsymbol{\eta}_\lambda} := E[\boldsymbol{\eta}_\lambda(t)\boldsymbol{\eta}_\lambda^T(t)]$. Also recall that $\boldsymbol{\epsilon}(t)$ depends on the entire history of regressors up to time instant t. Starting from (9) and capitalizing on the independence setting assumptions (a2)-(a3), it is straightforward to obtain a first-order matrix recursion to update $\mathbf{R}_z(t)$ as

$$\begin{aligned}\mathbf{R}_z(t) &= \boldsymbol{\Psi}\mathbf{R}_z(t-1)\boldsymbol{\Psi}^T + \boldsymbol{\Psi}\mathbf{R}_{\bar{\boldsymbol{\eta}}_\lambda}\boldsymbol{\Psi}^T \\ &+ \boldsymbol{\Psi}\mathbf{R}_{\boldsymbol{\eta}_\lambda}\boldsymbol{\Psi}^T + \begin{bmatrix}\mathbf{R}_{h,\lambda}^{-1}\\ \mathbf{0}_{Jp\times Jp}\end{bmatrix}\mathbf{R}_{\boldsymbol{\epsilon}}(t)\begin{bmatrix}\mathbf{R}_{h,\lambda}^{-1}\\ \mathbf{0}_{Jp\times Jp}\end{bmatrix}^T \\ &+ \boldsymbol{\Psi}\mathbf{R}_{z\boldsymbol{\epsilon}}(t)\begin{bmatrix}\mathbf{R}_{h,\lambda}^{-1}\\ \mathbf{0}_{Jp\times Jp}\end{bmatrix}^T + \left(\boldsymbol{\Psi}\mathbf{R}_{z\boldsymbol{\epsilon}}(t)\begin{bmatrix}\mathbf{R}_{h,\lambda}^{-1}\\ \mathbf{0}_{Jp\times Jp}\end{bmatrix}^T\right)^T\end{aligned} \tag{13}$$

where the cross-correlation $\mathbf{R}_{z\boldsymbol{\epsilon}}(t) := E[\mathbf{z}(t-1)\boldsymbol{\epsilon}^T(t)]$ is recursively updated as

$$\mathbf{R}_{z\boldsymbol{\epsilon}}(t) = \lambda\boldsymbol{\Psi}\mathbf{R}_{z\boldsymbol{\epsilon}}(t-1) + \lambda\begin{bmatrix}\mathbf{R}_{h,\lambda}^{-1}\\ \mathbf{0}_{Jp\times Jp}\end{bmatrix}\mathbf{R}_{\boldsymbol{\epsilon}}(t-1). \tag{14}$$

The main result of this section pertains to MSE stability of the D-RLS algorithm, and provides a checkable sufficient condition under which the global error covariance matrix has bounded entries as $t \to \infty$. Recall that a matrix is termed stable when its spectral radius is strictly less than one.

Proposition 2: *Under (a1)-(a3) and for $0 \ll \lambda < 1$, the D-RLS algorithm is MSE stable, i.e., $\lim_{t\to\infty}\mathbf{R}_{y_1}(t)$ has bounded entries, provided that $c > 0$ is chosen so that $\boldsymbol{\Psi}$ is a stable matrix.*

Proposition 2 asserts that the D-RLS algorithm is stable in the MSE-sense, even when the exchanges of information among agents are imperfect. While most distributed adaptive estimation works have only looked at ideal inter-agent links, others have adopted diminishing step-sizes to mitigate the undesirable effects of communication noise [12]. This approach however, limits their applicability to stationary environments. Remarkably, the D-RLS algorithm exhibits robustness to noise when using a constant step-size c.

Remark 1: Because the 'average' system representation of $\mathbf{y}(t)$ relies on an approximation that becomes increasingly accurate as $\lambda \to 1$ and $t \to \infty$, so does the covariance recursion for $\mathbf{R}_y(t)$ derived in (11). For this reason, the scope of the MSE performance analysis of this paper is limited to the *steady-state* behavior of the D-RLS algorithm.

Remark 2: The following steps enable evaluation of the s.s. MSE performance of D-RLS. First, one can solve for $\mathbf{R}_{z\boldsymbol{\epsilon}}(\infty)$ using $\mathbf{R}_{\boldsymbol{\epsilon}}(\infty)$ in (14). Plugging the result into (13) one obtains the s.s. covariance matrix of the forcing terms in (13). It is then possible to evaluate $\mathbf{R}_z(\infty)$, by solving (through vectorization) for the fixed point of (13) as $t \to \infty$. Matrix $\mathbf{R}_{z_1}(\infty)$ can be extracted from the upper-left $Jp \times Jp$ matrix block of $\mathbf{R}_z(\infty)$, and the desired global error covariance matrix $\mathbf{R}_{y_1}(\infty) = [\mathbf{R}_y(\infty)]_{11}$ becomes available via (11). Closed-form evaluation of the MSE(∞), EMSE(∞) and MSD(∞) for every agent $j \in \mathcal{J}$ is now possible given $\mathbf{R}_{y_1}(\infty)$, by resorting to the formulae in Table 1.

Before closing this section, an alternative notion of stochastic stability that readily follows from Proposition 2 is established here. Specifically, it is possible to show that the global error norm $\|\mathbf{y}_1(t)\|$ remains most of the time in a finite interval, i.e., errors are weakly stochastic bounded (WSB) [1, pg. 110]. This WSB stability guarantees that for any $\theta > 0$, $\exists\zeta > 0$ such that $\Pr[\|\mathbf{y}_1(t)\| < \zeta] = 1 - \theta$ uniformly in time.

Corollary 1: *Under (a1)-(a3) and for $0 \ll \lambda < 1$, if $c > 0$ is chosen so that $\boldsymbol{\Psi}$ is a stable matrix, then the D-RLS algorithm yields estimation errors which are WSB; i.e.,*$\lim_{\zeta\to\infty}\sup_{t\geq t_0}\Pr[\|\mathbf{y}_1(t)\| \geq \zeta] = 0$.

In words, Corollary 1 ensures that with overwhelming probability, local agent estimates remain inside a ball with finite radius, centered at $\mathbf{s}_0$.

5. NUMERICAL TESTS

Computer simulations are carried out here to corroborate the analytical results of Section 4.2. Even though based on simplifying assumptions and approximations, the usefulness of the analysis is justified since the predicted s.s. MSE figures of merit accurately match the empirical D-RLS limiting values. Interestingly, when $\lambda \to 1$ the upshot of the analysis under the independence setting assumptions is shown to extend accurately to the pragmatic scenario whereby agents acquire time-correlated data. For $J = 15$ agents, a connected network is generated as a realization of the random geometric graph model on the unit-square, with communication range $r = 0.3$. To model imperfect inter-agent links, additive white Gaussian noise (AWGN) with variance $\sigma_\eta^2 = 0.5$ is added at the receiving end.

With $p = 4$, observations obey a linear model [cf. (a1)] with sensing AWGN of spatial variance profile $\sigma_{\epsilon_j}^2 = 10^{-3}\alpha_j$, where $\alpha_j \sim \mathcal{U}[0,1]$ (are uniformly distributed) and i.i.d.. Regression vectors $\mathbf{h}_j(t) := [h_j(t)\dots h_j(t-p+1)]^T$ have a shift structure, and entries which evolve according to first-order stable autoregressive processes $h_j(t) = (1-\rho)\beta_j h_j(t-1) + \sqrt{\rho}\omega_j(t)$ for all $j \in \mathcal{J}$. Parameters are selected as $\rho = 5 \times 10^{-1}$, $\beta_j \sim \mathcal{U}[0,1]$ i.i.d. in space, and the driving white noise $\omega_j(t) \sim \mathcal{U}[-\sqrt{3}\sigma_{\omega_j}, \sqrt{3}\sigma_{\omega_j}]$ with spatial variance profile given by $\sigma_{\omega_j}^2 = 2\gamma_j$ with $\gamma_j \sim \mathcal{U}[0,1]$ and i.i.d.. Accordingly, the local covariance matrices $\mathbf{R}_{h_j}$ have a Toeplitz structure. Observe that the data is temporally correlated, implying that (a2) does not hold here. The time-invariant parameter vector sought is $\mathbf{s}_0 = \mathbf{1}_p$. For all experimental performance curves obtained by running the algorithms, the ensemble averages are approximated via sample averaging over 200 runs of the experiment.

With all-zero initializations, $\lambda = 0.99$, $c = 0.1$ and $\delta = 100$ for the D-RLS algorithm, Fig. 1 depicts the network

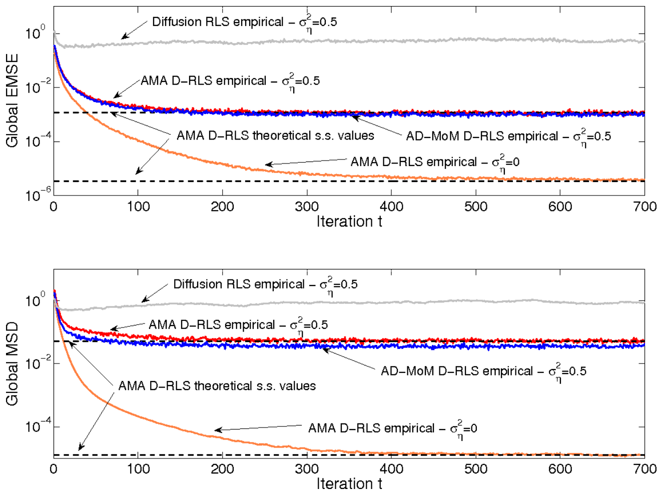

Fig. 1. Global s.s. performance when $\lambda = 0.99$. D-RLS is run with ideal links and when communication noise with variance $\sigma_\eta^2 = 0.5$ is present. Comparisons with the AD-MoM-based D-RLS and diffusion RLS algorithms are shown as well.

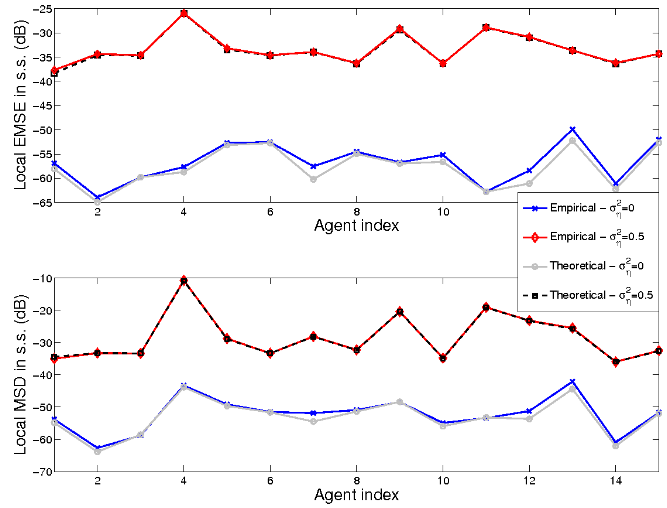

Fig. 2. Local s.s. performance evaluation when $\lambda = 0.99$. D-RLS is run with ideal links and when communication noise with variance $\sigma_\eta^2 = 0.5$ is present.

performance through the evolution of $\text{EMSE}(t)$ and $\text{MSD}(t)$ figures of merit. Even though the focus here is on noisy exchanges among agents, ideal links are also considered to assess the (expected) performance degradation due to communication noise. The s.s. limiting values found in Section 4.2 are extremely accurate, even though the simulated data does not adhere to (a2), and the results are based on simplifying approximations. Simulated error trajectory curves for the alternating-direction method of multipliers (AD-MoM)-based D-RLS [11] and diffusion RLS algorithms (with Metropolis combining weights) [3] are also included. Since $\lambda < 1$, the AD-MoM-based D-RLS algorithm demands an order of magnitude increase in terms of computational complexity per agent [9, Sec. II-B], yet its performance is comparable to that of D-RLS. Note also that in the presence of communication noise, diffusion RLS yields inaccurate and biased local estimates [4]. Similar overall conclusions can be drawn from the plots in Fig. 2, that gauge local performance by depicting $\{\text{EMSE}_j(\infty)\}_{j=1}^J$ and $\{\text{MSD}_j(\infty)\}_{j=1}^J$. The curves for the AD-MoM-based D-RLS and diffusion RLS algorithms were not included in the interest of clarity.

6. REFERENCES

[1] V. Solo and X. Kong, *Adaptive Signal Processing Algorithms: Stability and Performance*, Prentice Hall, 1995.

[2] A. H. Sayed, *Fundamentals of Adaptive Filtering*, John Wiley & Sons, 2003.

[3] 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, pp. 1865–1877, May 2008.

[4] A. Bertrand, M. Moonen, and A. H. Sayed, "Diffusion bias-compensated RLS estimation over adaptive networks," *IEEE Trans. Signal Process.*, vol. 59, pp. 5212–5224, Nov. 2011.

[5] N. R. Yousef and A. H. Sayed, "A unified approach to the steady-state and tracking analysis of adaptive filters," *IEEE Trans. Signal Process.*, vol. 49, pp. 314–324, Feb. 2001.

[6] C. G. Lopes and A. H. Sayed, "Diffusion least-mean squares over adaptive networks: Formulation and performance analysis," *IEEE Trans. Signal Process.*, vol. 56, pp. 3122–3136, July 2008.

[7] I. D. Schizas, G. Mateos, and G. B. Giannakis, "Distributed LMS for consesus-based in-network adaptive processing," *IEEE Trans. Signal Process.*, vol. 57, pp. 2365–2381, June 2009.

[8] A. H. Sayed and C. G. Lopes, "Distributed recursive least-squares over adaptive networks," in *Proc. of 40th Asilomar Conf. On Signals, Systems and Computers*, Pacific Grove, CA, Oct./Nov. 2006, pp. 233–237.

[9] G. Mateos and G. B. Giannakis, "Distributed recursive least-squares: Stability and performance analysis," *IEEE Trans. Signal Process.*, vol. 60, July 2012, to appear; see also arXiv:1109.4627v1 [cs.NI].

[10] P. Tseng, "Applications of a splitting algorithm to decomposition in convex programming and variational inequalities," *SIAM Journal on Control and Optimization*, vol. 29, pp. 119–138, Jan. 1991.

[11] G. Mateos, I. D. Schizas, and G. B. Giannakis, "Distributed recursive least-squares for consensus-based in-network adaptive estimation," *IEEE Trans. Signal Process.*, vol. 57, pp. 4583–4588, Nov. 2009.

[12] S. Kar and J. M. F. Moura, "Distributed consensus algorithms in sensor networks with imperfect communication: link failures and channel noise," *IEEE Trans. Signal Process.*, vol. 57, pp. 355–369, Jan. 2009.