

Energy-Efficient Distributed Recursive Least Squares Learning with Coarsely Quantized Signals

Alireza Danaee*, Rodrigo C. de Lamare^{*,†}, and Vitor H. Nascimento[‡]

* Centre for Telecommunications Studies, Pontifical Catholic University of Rio de Janeiro, Brazil

† Department of Electronic Engineering, University of York, United Kingdom

‡ Department of Electronic Systems Engineering, University of São Paulo, Brazil

danaee.alireza@gmail.com, delamare@cetuc.puc-rio.br, vitor@lps.usp.br

This work was supported in part by the ELIOT project, FAPESP 2018/12579-7 and ANR-18-CE40-0030

Abstract—In this work, we present an energy-efficient distributed learning framework using coarsely quantized signals for Internet of Things (IoT) networks. In particular, we develop a distributed quantization-aware recursive least squares (DQA-RLS) algorithm that can learn parameters in an energy-efficient fashion using signals quantized with few bits while requiring a low computational cost. Numerical results assess the DQA-RLS algorithm against existing techniques for a distributed parameter estimation task where IoT devices operate in a peer-to-peer mode.

Index Terms—distributed learning, energy-efficient signal processing, adaptive algorithms, coarse quantization

I. INTRODUCTION

Distributed signal processing algorithms are of great relevance for statistical inference in wireless networks and applications such as wireless sensor networks (WSNs) [1] and the Internet of Things (IoT) [2]. In fact, distributed signal processing techniques deal with the extraction of information from data collected at nodes that are distributed over a geographical area. In this context, for each node a set of neighbor nodes collects and processes their local information, and transmits their estimates to a specific node. Then, each specific node combines the collected information together with its local estimate to generate improved estimates.

Prior work on distributed signal processing techniques has studied protocols for exchanging information [3]–[5], adaptive learning algorithms [6], the exploitation of sparse measurements [7] and topology adaptation [8]. Even though there have been many studies that have evaluated the need for data exchange and signaling among nodes as well as their computational complexity, prior work on energy-efficient techniques is rather limited and there is no distributed learning algorithm devised to deal with coarsely quantized signals.

In this context, energy-efficient signal processing techniques have gained a great deal of interest in the last decade or so due to their ability to save energy and promote sustainable development of electronic systems and devices. Electronic devices often exhibit an energy consumption that is strongly dependent on the analog-to-digital converters (ADCs) and the number of bits used to represent digital samples [9]. This is of central importance to devices that are battery operated and wireless networks that must keep the energy consumption to a low level for sustainability reasons. In particular, prior work

on energy efficiency has reported many contributions in signal processing for communications and electronic systems that operate with coarsely quantized signals [10]–[12].

In this work, we propose an energy-efficient distributed learning framework using low-resolution ADCs and signals for IoT networks. In particular, we devise a distributed quantization-aware recursive least squares (DQA-RLS) algorithm that can learn parameters in an energy-efficient way using signals quantized with few bits with a low computational cost. Simulations assess the proposed DQA-RLS algorithm against existing techniques for a distributed parameter estimation task where IoT devices operate in a peer-to-peer mode.

This paper is structured as follows: Section 2 introduces the signal model and states the problem. Section 3 details the proposed DQA-RLS algorithms, whereas section 4 shows and discusses the results of simulations and section 5 draws the conclusions of this work.

II. SIGNAL MODEL AND PROBLEM STATEMENT

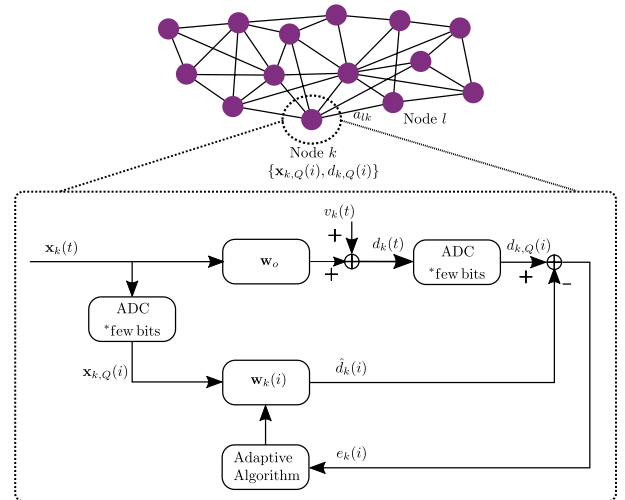


Fig. 1. A distributed adaptive IoT network

We consider an IoT network consisting of N nodes or agents which runs distributed signal processing techniques to perform the desired tasks, as depicted in Fig. 1. The model adopted considers a desired signal $d_k(i)$, at each time i described by

$$d_k(i) = \mathbf{w}_o^H \mathbf{x}_k(i) + v_k(i), \quad k = 1, 2, \dots, N, \quad (1)$$

where $\mathbf{w}_o \in \mathbb{C}^{M \times 1}$ is the parameter vector that the agents must estimate, $\mathbf{x}_k(i) \in \mathbb{C}^{M \times 1}$ is the regressor at node k , and $v_k(i)$ represents Gaussian noise with zero mean and variance $\sigma_{v,k}^2$ at each node k . We also consider the Adapt-then-Combine (ATC) diffusion rule as a more effective scheme than other previously reported schemes such as incremental and consensus [3], [4].

As shown in Fig. 1, because the measurement data at each node and the unknown system are analog and each agent processes the local data $\{d_k(i), \mathbf{x}_k(i)\}$ digitally, we need two ADCs in each agent. One concern is that as the number of agents increases, the energy consumption will grow too much when using high-resolution ADCs for each agent. This motivates us to quantize signals using few bits. Therefore, the problem we are interested in solving in this work is how to design energy-efficient distributed learning algorithms that can cost-effectively operate with coarsely quantized signals.

III. PROPOSED DQA-RLS ALGORITHM

Let $\mathbf{x}_{k,Q} = Q_b(\mathbf{x}_k)$ denote the b -bits quantized output of an ADC at node k , described by a set of $2^b + 1$ thresholds $\mathcal{T}_b = \{\tau_0, \tau_1, \dots, \tau_{2^b}\}$, such that $-\infty = \tau_0 < \tau_1 < \dots < \tau_{2^b} = \infty$, and the set of 2^b labels $\mathcal{L}_b = \{l_0, l_1, \dots, l_{2^b-1}\}$ where $l_p \in (\tau_p, \tau_{p+1}]$, for $p \in [0, 2^b - 1]$ [10]. Let us assume that $\mathbf{x}_k \sim \mathcal{CN}(\mathbf{0}, \mathbf{R}_{x_k})$ where $\mathbf{R}_{x_k} \in \mathbb{C}^{M \times M}$ is the covariance matrix of \mathbf{x}_k . We now use Bussgang's theorem [13] to derive a model for the quantized vector $\mathbf{x}_{k,Q}$, which we later use to derive our DQA-RLS algorithm. Employing Bussgang's theorem, $\mathbf{x}_{k,Q}$ can be decomposed as

$$\mathbf{x}_{k,Q} = \mathbf{G}_{k,b} \mathbf{x}_k + \mathbf{q}_k \quad (2)$$

where the quantization distortion \mathbf{q}_k is uncorrelated with \mathbf{x}_k , and $\mathbf{G}_{k,b} \in \mathbb{R}^{M \times M}$ is a diagonal matrix described by

$$\mathbf{G}_{k,b} = \text{diag}(\mathbf{R}_{x_k})^{-\frac{1}{2}} \sum_{j=0}^{2^b-1} \frac{l_j}{\sqrt{\pi}} \left[\exp(-\tau_j^2 \text{diag}(\mathbf{R}_{x_k})^{-1}) - \exp(-\tau_{j+1}^2 \text{diag}(\mathbf{R}_{x_k})^{-1}) \right] \quad (3)$$

Note that, as a simplifying approximation, we also apply this signal decomposition to the desired signal, $d_{k,Q}$, which is the output of the second ADC in the system, and for the particular case that $\mathbf{R}_{x_k} = \mathbb{E}[\mathbf{x}_k \mathbf{x}_k^H] = \sigma_{x,k}^2 \mathbf{I}_M$, the matrix $\mathbf{G}_{k,b}$ becomes $g_{k,b} \mathbf{I}_M$. However, to minimize the mean square error (MSE) between \mathbf{x}_k and $\mathbf{x}_{k,Q}$, we need to characterize the probability density function (PDF) of \mathbf{x}_k to find the optimal quantization labels. Because choosing these labels based on such PDF is ineffective in practice (since the PDFs are difficult to estimate), we assume the regressor $\mathbf{x}_k(i)$ is Gaussian, then adapt the approach in [10] and approximate the thresholds and labels as follows:

- 1) We generate an auxiliary Gaussian random variable with unit variance and then use the Lloyd-Max algorithm [14], [15] to find a set of thresholds $\tilde{\mathcal{T}}_b = \{\tilde{\tau}_1, \dots, \tilde{\tau}_{2^b-1}\}$ and labels $\tilde{\mathcal{L}}_b = \{\tilde{l}_0, \dots, \tilde{l}_{2^b-1}\}$ that minimize the MSE between the unquantized and the quantized signals.
- 2) We wrap up the set of thresholds $\tilde{\mathcal{T}}_b$ by adding $\tau_0 = -\infty$ and $\tau_{2^b} = \infty$ to the $\tilde{\mathcal{T}}_b$.

- 3) We rescale the labels such that the variance of the auxiliary random variable is 1. To do this, we multiply each label in the set $\tilde{\mathcal{L}}_b$ by

$$\alpha = \frac{1}{\sqrt{2 \sum_{j=0}^{2^b-1} \tilde{l}_j^2 (\Phi(\sqrt{2\tau_{j+1}^2}) - \Phi(\sqrt{2\tau_j^2}))}} \quad (4)$$

to produce a set of suboptimal labels $\mathcal{L}_b = \alpha \tilde{\mathcal{L}}_b$, where $\Phi(\cdot)$ is the cumulative distribution function (CDF) of a standard Gaussian random variable.

We compute these thresholds and labels offline and use them to build the diagonal matrix $\mathbf{G}_{k,b}$ for the proposed DQA-RLS algorithm in what follows.

A. Derivation of DQA-RLS

We consider $\mathbf{x}_k(t)$ and $d_k(t)$ as the analog input and output of the unknown system \mathbf{w}_o at node k . Let $\mathbf{x}_k(i)$ and $d_k(i)$ denote the digital versions of $\mathbf{x}_k(t)$ and $d_k(t)$, and $\mathbf{x}_{k,Q}(i)$ and $d_{k,Q}(i)$ denote the coarsely quantized versions of $\mathbf{x}_k(i)$ and $d_k(i)$, respectively. We assume that the input signal at each node is Gaussian with zero mean and covariance matrix $\mathbf{R}_{x_k} = E[\mathbf{x}_k \mathbf{x}_k^H] = \sigma_{x,k}^2 \mathbf{I}_M$ for $k = 1, 2, \dots, N$. We can now write $d_{k,Q}(i)$ as

$$\begin{aligned} d_{k,Q}(i) &= Q(d_k(i)) = g_{k,b}(i) d_k(i) + q_k(i) \\ &= g_{k,b}(i) (\mathbf{w}_o^H \mathbf{x}_k(i) + v_k(i)) + q_k(i) \\ &= g_{k,b}(i) \mathbf{w}_o^H \mathbf{x}_k(i) + \hat{q}_k(i), \end{aligned} \quad (5)$$

where $\hat{q}_k(i) = g_{k,b}(i) v_k(i) + q_k(i)$ and $g_{k,b}(i)$ is built from an estimate of \mathbf{R}_{x_k} given by $\hat{\mathbf{R}}_{x_k} = \mathbf{x}_k \mathbf{x}_k^H$ [16], as it depends on the choice of the input vector \mathbf{x}_k due to (1). Because the adaptive algorithm receives a quantized signal, $\mathbf{x}_{k,Q}$, and the signal is assumed to be wide-sense stationary, at each time instant, we estimate $\sigma_{x,k}^2$ using the variance of the received input, $\sigma_{x_{k,Q}}^2$ and the distortion factor of the b -bit quantization, $\rho_{k,b}$, such that $\sigma_{x,k}^2 \approx \sigma_{x_{k,Q}}^2 + \rho_{k,b}$, where $\rho_{k,b} \approx \frac{\pi\sqrt{3}}{2} 2^{-2b}$ [11] for a Gaussian signal using non-uniform quantization in order to obtain the scalar $g_{k,b}(i)$.

Let us consider a network of N nodes distributed over an area as in Fig. 1. At time i , we collect the quantized desired signal and noise samples into vectors $\mathbf{d}_{i,Q}$ and \mathbf{v}_i , and the quantized input regressors into a matrix $\mathbf{X}_{i,Q}$ as follows

$$\begin{aligned} \mathbf{X}_i &= \text{col}\{\mathbf{x}_1^T(i), \dots, \mathbf{x}_N^T(i)\} \quad (N \times M) \\ \mathbf{X}_{i,Q} &= \text{col}\{\mathbf{x}_{1,Q}^T(i), \dots, \mathbf{x}_{N,Q}^T(i)\} \quad (N \times M) \\ \mathbf{d}_{i,Q} &= \text{col}\{d_{1,Q}(i), \dots, d_{N,Q}(i)\} \quad (N \times 1) \\ \mathbf{v}_i &= \text{col}\{v_1(i), \dots, v_N(i)\} \quad (N \times 1) \\ \mathbf{g}_{i,b} &= \text{col}\{g_{1,b}(i), \dots, g_{N,b}(i)\} \quad (N \times 1). \end{aligned} \quad (6)$$

If \mathbf{v}_i^* denotes the complex conjugate transpose of \mathbf{v}_i , we can write down the covariance matrix of the noise vector as follows

$$\mathbf{R}_v = \mathbb{E}[\mathbf{v}_i \mathbf{v}_i^*] = \text{diag}\{\sigma_{v_1}^2, \dots, \sigma_{v_N}^2\} \quad (N \times N). \quad (7)$$

Now we collect these data from time 0 to time i as follows

$$\begin{aligned}\mathcal{X}_i &= \text{col}\{\mathbf{X}_i, \dots, \mathbf{X}_0\}, \\ \mathcal{X}_{i,Q} &= \text{col}\{\mathbf{X}_{i,Q}, \dots, \mathbf{X}_{0,Q}\}, \\ \mathcal{D}_{i,Q} &= \text{col}\{\mathbf{d}_{i,Q}, \dots, \mathbf{d}_{0,Q}\}, \\ \mathcal{V}_i &= \text{col}\{\mathbf{v}_i, \dots, \mathbf{v}_0\}, \\ \mathcal{G}_{i,b} &= \text{col}\{\mathbf{g}_{i,b}, \dots, \mathbf{g}_{0,b}\},\end{aligned}\quad (8)$$

and write down $\mathcal{R}_{v,i} = \mathbb{E}[\mathcal{V}_i \mathcal{V}_i^*]$. In order to devise a learning algorithm based on (5), it is convenient to first define $\hat{d}_k(i) = g_{k,b}(i) \mathbf{w}_k^H(i-1) \mathbf{x}_{k,Q}(i)$. Then, we estimate \mathbf{w}_o by solving the weighted, regularized least squares problem given by

$$\min_{\mathbf{w}} \|\mathbf{w} - \bar{\mathbf{w}}\|_{\Pi_i}^2 + \|\mathcal{D}_{i,Q} - \mathcal{G}_{i,b} \mathbf{w}^H \mathcal{X}_{i,Q}\|_{\mathcal{W}_i}^2 \quad (9)$$

The common solution $\mathbf{w}(i)$ is given by [17]

$$\mathbf{w}_i = \bar{\mathbf{w}} + (\Pi_i + \mathcal{X}_{i,Q}^* \mathcal{W}_i \mathcal{X}_{i,Q})^{-1} \mathcal{X}_{i,Q}^* \mathcal{W}_i (\mathcal{D}_{i,Q} - \mathcal{G}_{i,b} \bar{\mathbf{w}}^H \mathcal{X}_{i,Q}) \quad (10)$$

where $\Pi_i > 0$ and $\mathcal{W}_i > 0$ are the regularization and the weighting Hermitian matrices. An exponentially weighted version of (9) can be derived choosing

$$\mathcal{W}_i = \mathcal{R}_{v,i}^{-1} \Lambda_i \quad \text{and} \quad \Pi_i = \lambda^{i+1} \Pi, \quad (11)$$

where $0 < \lambda < 1$, $\Pi > 0$ and $\Lambda_i \triangleq \text{diag}\{\mathbf{I}_N, \lambda \mathbf{I}_N, \dots, \lambda^i \mathbf{I}_N\}$. Usually, $\Pi = \delta^{-1} \mathbf{I}_M$ where $\delta > 0$ is large. Often, in least squares estimation, when the noise variances $\sigma_{v_k}^2$ are unknown, the weighting matrix \mathcal{W}_i is simply replaced by $\mathcal{W}_i = \Lambda_i$.

Choosing $\bar{\mathbf{w}} = 0$ the estimation problem (9) will be

$$\mathbf{w}(i) = \arg \min_{\mathbf{w}} \left\{ \lambda^{i+1} \|\mathbf{w}\|_{\Pi}^2 + \sum_{j=0}^i \lambda^{i-j} \sum_{l=1}^N \frac{|d_{l,Q}(j) - g_{k,b}(i) \mathbf{w}_k^H(i-1) \mathbf{x}_{k,Q}(i)|^2}{\sigma_{v_l}^2} \right\}. \quad (12)$$

We then reformulate the global least squares problem in (12) to a local least squares problem as follows:

$$\mathbf{h}_k(i) = \arg \min_{\mathbf{w}} \left\{ \lambda^{i+1} \|\mathbf{w}\|_{\Pi}^2 + \sum_{j=0}^i \lambda^{i-j} \sum_{l=1}^N \frac{c_{lk} |d_{l,Q}(j) - g_{k,b}(i) \mathbf{w}_k^H(i-1) \mathbf{x}_{k,Q}(i)|^2}{\sigma_{v_l}^2} \right\}, \quad (13)$$

for weighting coefficients c_{lk} such that

$$c_{lk} = 0 \text{ if } l \notin \mathcal{N}_k \text{ and } c_{lk} > 0 \text{ if } l \in \mathcal{N}_k.$$

The coefficients c_{lk} can be incorporated into the weighting matrix of (9) by replacing \mathcal{W}_i with

$$\mathcal{W}_{k,i} = \mathcal{R}_{v,i}^{-1} \Lambda_i \text{diag}\{\mathbf{C}_k, \mathbf{C}_k, \dots, \mathbf{C}_k\} \quad (14)$$

where $\mathbf{C}_k = \text{diag}\{\mathbf{C}_{e_k}\}$ of size $N \times N$, and \mathbf{e}_k is the $N \times 1$ vector with a unity entry in position k and zeros elsewhere.

When there is no regularization ($\Pi = 0$), the solution $\mathbf{h}_k(i)$ for (10) is described by

$$\mathbf{h}_k(i) = (\mathcal{X}_{i,Q}^* \mathcal{W}_{k,i} \mathcal{X}_{i,Q})^{-1} \mathcal{X}_{i,Q}^* \mathcal{W}_{k,i} \mathcal{D}_{i,Q} \quad (15)$$

where

$$\mathcal{D}_{i,Q} = Q(\mathbf{w}_o^H \mathcal{X}_i + \mathcal{V}_i) = \mathcal{G}_{i,b} \mathbf{w}_o^H \mathcal{X}_i + \hat{\mathcal{V}}_i. \quad (16)$$

We replace \mathcal{W}_i in (10) with $\mathcal{W}_{k,i}$, $\Pi_i = \lambda^{i+1} \Pi$, and $\bar{\mathbf{w}} = 0$ to obtain the solution to (13). It can be formulated as

$$\mathbf{h}_k(i) = \mathbf{P}_k(i) \mathcal{X}_{i,Q}^* \mathcal{W}_{k,i} \mathcal{D}_{i,Q} \quad (17)$$

where

$$\mathbf{P}_k(i) = (\lambda^{i+1} \Pi + \mathcal{X}_{i,Q}^* \mathcal{W}_{k,i} \mathcal{X}_{i,Q})^{-1}. \quad (18)$$

To form the recursion, we compute $\mathbf{P}_k(i)$ from $\mathbf{P}_k(i-1)$ considering

$$\begin{aligned}\mathbf{P}_k^{-1}(i) &= \lambda[\lambda^i \Pi + \mathcal{X}_Q(i-1)^* \mathcal{W}_k(i-1) \mathcal{X}_Q(i-1)] + \\ &\quad \mathcal{X}_{i,Q}^* \mathbf{R}_v^{-1} \mathbf{C}_k \mathbf{X}_{i,Q} \\ &= \lambda \mathbf{P}_k^{-1}(i-1) + \mathcal{X}_{i,Q}^* \mathbf{R}_v^{-1} \mathbf{C}_k \mathbf{X}_{i,Q} \\ &= \lambda \mathbf{P}_k^{-1}(i-1) + \sum_{l=1}^N \frac{c_{lk}}{\sigma_{v_l}^2} \mathbf{x}_l(i) \mathbf{x}_l^*(i).\end{aligned} \quad (19)$$

To implement the recursion with reduced complexity, we use a series of rank-one updates as follows

$$\mathbf{P}_k^0(i) \leftarrow \lambda^{-1} \mathbf{P}_k(i-1)$$

For $l = 1$ to N , repeat

$$\mathbf{P}_k^l(i) \leftarrow \left[(\mathbf{P}_k^{l-1}(i))^{-1} + \frac{c_{lk}}{\sigma_{v_l}^2} \mathbf{x}_l(i) \mathbf{x}_l^*(i) \right]^{-1} \quad (20)$$

end

$$\mathbf{P}_k(i) \leftarrow \mathbf{P}_k^N(i).$$

The matrices $\mathbf{P}_k^l(i)$ denote the intermediate results after every rank-one update and are needed for only neighbor nodes where $c_{lk} \neq 0$. Using the matrix inversion lemma, we can write (20) as follows

$$\mathbf{P}_k(i) \leftarrow \lambda^{-1} \mathbf{P}_k(i-1)$$

For every $l \in \mathcal{N}_k$ repeat

$$\mathbf{P}_k(i) \leftarrow \mathbf{P}_k(i) - \frac{c_{lk} \mathbf{P}_k(i) \mathbf{x}_l(i) \mathbf{x}_l^*(i) \mathbf{P}_k(i)}{\sigma_{v_l}^2 + c_{lk} \mathbf{x}_l^*(i) \mathbf{P}_k(i) \mathbf{x}_l(i)} \quad (21)$$

end.

To build a recursion for the update of (17), we collect all measurements and regressors from all nodes up to time $i-1$, from nodes 1 to l and denote the intermediate matrices as

$$\begin{aligned}\mathcal{D}_{i,Q}^l &= \begin{bmatrix} d_{l,Q}(i) \\ \vdots \\ d_{1,Q}(i) \\ \mathcal{D}_{i-1,Q}^l \end{bmatrix}, \quad \mathcal{X}_{i,Q}^l = \begin{bmatrix} \mathbf{x}_{l,Q}(i) \\ \vdots \\ \mathbf{x}_{1,Q}(i) \\ \mathcal{X}_{i-1,Q}^l \end{bmatrix} \\ \mathcal{W}_{k,i}^l &= \begin{bmatrix} \frac{c_{lk}}{\sigma_{v_l}^2} & & & \\ & \ddots & & \\ & & \frac{c_{1k}}{\sigma_{v_1}^2} & \\ & & & \lambda \mathcal{W}_{k,i-1}^l \end{bmatrix}.\end{aligned} \quad (22)$$

This allows us to write the intermediate estimates $\mathbf{h}_k^l(i)$ as

$$\mathbf{h}_k^l(i) = \mathbf{P}_k^l(i) (\mathcal{X}_{i,Q}^l)^* \mathcal{W}_{k,i}^l \mathcal{D}_{i,Q}^l, \quad (23)$$

where the $\mathbf{P}_k^l(i)$ is defined in (20), and can also be written as

$$\mathbf{P}_k^l(i) = [\lambda^{i+1} \mathbf{\Pi} + (\mathcal{X}_{i,Q}^l)^* \mathcal{W}_{k,i}^l \mathcal{D}_{i,Q}^l]^{-1}. \quad (24)$$

It holds that $\mathbf{h}_k(i-1) = \mathbf{h}_k^N(i-1) \triangleq \mathbf{h}_k^0(i)$. Let j denote the smallest index such that $c_{jk} \neq 0$. We can calculate $\mathbf{h}_k^j(i)$ from $\mathbf{h}_k^0(i)$ as follows

$$\begin{aligned} \mathbf{h}_k^j(i) &= \mathbf{P}_k^j(i) \left[\lambda (\mathcal{X}_{i,Q}^0)^* \mathcal{W}_{k,i}^0 \mathcal{D}_{i,Q}^0 + \frac{c_{jk}}{\sigma_{v_j}^2} \mathbf{x}_{j,Q}(i) d_{j,Q}(i) \right] \\ &= \underbrace{\lambda \mathbf{P}_k^0(i) (\mathcal{X}_{i,Q}^0)^* \mathcal{W}_{k,i}^0 \mathcal{D}_{i,Q}^0}_{\mathbf{h}_k(i-1)} + \frac{c_{jk}}{\sigma_{v_j}^2} \mathbf{P}_k^0(i) \mathbf{x}_{j,Q}(i) \\ &\quad \left(1 - \frac{c_{jk} \mathbf{x}_j^*(i) \mathbf{P}_k^0(i) \mathbf{x}_j(i)}{\sigma_{v_j}^2 + c_{jk} \mathbf{x}_j^*(i) \mathbf{P}_k^0(i) \mathbf{x}_j(i)} \right) d_{j,Q}(i) \\ &\quad - \frac{c_{jk} \mathbf{x}_j^*(i) \mathbf{P}_k^0(i) \mathbf{x}_j(i)}{\sigma_{v_j}^2 + c_{jk} \mathbf{x}_j^*(i) \mathbf{P}_k^0(i) \mathbf{x}_j(i)} \underbrace{\lambda \mathbf{P}_k^0(i) (\mathcal{X}_{i,Q}^0)^* \mathcal{W}_{k,i}^0 \mathcal{D}_{i,Q}^0}_{\mathbf{h}_k(i-1)} \\ &= \mathbf{h}_k(i-1) + \frac{c_{jk} \mathbf{P}_k^0(i) \mathbf{x}_j(i)}{\sigma_{v_j}^2 + c_{jk} \mathbf{x}_j^*(i) \mathbf{P}_k^0(i) \mathbf{x}_j(i)} (d_{j,Q}(i) \\ &\quad - g_{k,b}(i) \mathbf{h}_k^H(i-1) \mathbf{x}_j(i)). \end{aligned} \quad (25)$$

We derive recursions for $\mathbf{h}_k^l(i)$ from $\mathbf{h}_k^{l-1}(i)$ for $l = j + 1, \dots, N$ in a similar way as

$$\begin{aligned} \mathbf{h}_k^l(i) &= \mathbf{P}_k^l(i) \left[(\mathcal{X}_{i,Q}^{l-1})^* \mathcal{W}_{k,i}^{l-1} \mathcal{D}_{i,Q}^{l-1} + \frac{c_{lk}}{\sigma_{v_l}^2} \mathbf{x}_{l,Q}(i) d_{l,Q}(i) \right] \\ &= \mathbf{h}_k^{l-1}(i-1) + \frac{c_{lk} \mathbf{P}_k^{l-1}(i) \mathbf{x}_l(i)}{\sigma_{v_l}^2 + c_{lk} \mathbf{x}_l^*(i) \mathbf{P}_k^{l-1}(i) \mathbf{x}_l(i)} [d_{l,Q}(i) \\ &\quad - g_{k,b}(i) (\mathbf{h}_k^{l-1})^H(i-1) \mathbf{x}_l(i)]. \end{aligned} \quad (26)$$

To simplify the recursion, we drop the super-indexes j as we only need to consider values of l which $c_{lk} \neq 0$. Then regarding all time instants up to time i , we combine (21), (25) and (26) to build the recursion to solve (13) as in Table I.

We improve the estimation by exchanging the estimates between neighbor nodes in the diffusion fashion using a weighted average of the estimates of the nodes as in the last step in table I. The combination coefficients of neighbor nodes on node k , a_{lk} , are chosen such that

$$a_{lk} = 0 \text{ if } l \notin \mathcal{N}_k, a_{lk} > 0 \text{ if } l \in \mathcal{N}_k, \text{ and } \sum_{l \in \mathcal{N}_k} a_{lk} = 1.$$

B. Complexity and Energy Consumption

At each time instant, to compute $g_{k,b}(i)$ from (3), $2^{b+1} + 1$ multiplications, $2^b - 1$ additions, $2^b + 1$ divisions, and 2^b exponentiations are needed. Note that we compute $g_{k,b}(i)$ online, since this is more appropriate to deal with non-stationary input data. However, one can compute $\mathbf{G}_{k,b}$ offline having the covariance matrix of the input signal as in (3). To compute $\mathbf{h}_k(i)$ in table I, one more multiplication than in DRLS [5] is needed. Therefore, DQA-RLS performs a few more operations ($\approx O(2^b)$) than DRLS. However, the extra complexity in the DQA-RLS algorithm allows the system to work in a more energy-efficient way. In order to assess the power savings by

TABLE I
PSEUDO CODE OF DQA-RLS ALGORITHM

Initializations: $\mathbf{w}_k(-1) = 0$ and $\mathbf{P}_k(-1) = \mathbf{\Pi}^{-1}$ for each node k
At each time instant i and node k repeat
$g_{k,b}(i) = \frac{1}{\sqrt{\sigma_{x_k}^2}} \sum_{j=0}^{2^b-1} \frac{l_j}{\sqrt{\pi}} (e^{-\frac{\tau_j^2}{\sigma_{x_k}^2}} - e^{-\frac{\tau_{j+1}^2}{\sigma_{x_k}^2}})$
$\mathbf{h}_k(i) = \mathbf{w}_k(i-1)$
$\mathbf{P}_k(i) = \lambda^{-1} \mathbf{P}_k(i-1)$
for all $l \in \mathcal{N}_k$
$\mathbf{h}_k(i) \leftarrow \mathbf{h}_k(i) + \frac{c_{lk} \mathbf{P}_k(i) \mathbf{x}_l(i) [d_{l,Q}(i) - g_{k,b}(i) \mathbf{h}_k^H(i) \mathbf{x}_l(i)]}{\sigma_{v_l}^2 + c_{lk} \mathbf{x}_l^*(i) \mathbf{P}_k(i) \mathbf{x}_l(i)}$
$\mathbf{P}_k(i) \leftarrow \mathbf{P}_k(i) - \frac{c_{lk} \mathbf{P}_k(i) \mathbf{x}_l(i) \mathbf{x}_l^*(i) \mathbf{P}_k(i)}{\sigma_{v_l}^2 + c_{lk} \mathbf{x}_l^*(i) \mathbf{P}_k(i) \mathbf{x}_l(i)}$
end
for every node k repeat
$\mathbf{w}_k(i) = \sum_{l \in \mathcal{N}_k} a_{lk} \mathbf{h}_l(i)$

low resolution quantization, let us consider a network with N nodes in which each node uses two ADCs. The power consumption of each ADC is $P_{ADC}(b) = cB2^b$ [18], where B is the bandwidth (related to the sampling rate), b is the number of quantization bits of the ADC, and c is the power consumption per conversion step. Therefore, the total power consumption of the ADCs in the network is

$$P_{ADC,T}(b) = 2NcB2^b \quad (\text{watts}). \quad (27)$$

Fig. 2 shows an example of the total power consumption of ADCs in a narrowband IoT (NB-IoT) network running diffusion adaptation consisting of 20 nodes with bandwidth $B = 200$ kHz [19] and considering the energy consumption per conversion step of each ADC, $c = 494$ fJ, as in [20].

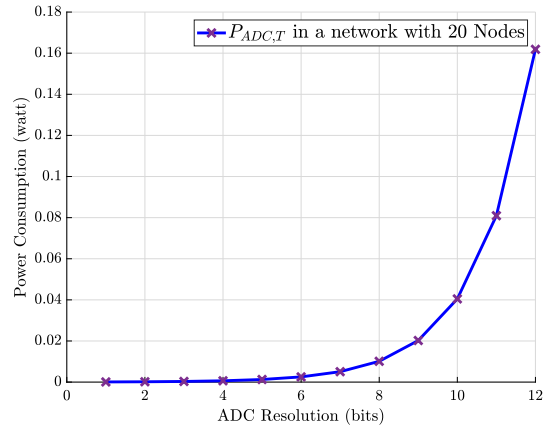
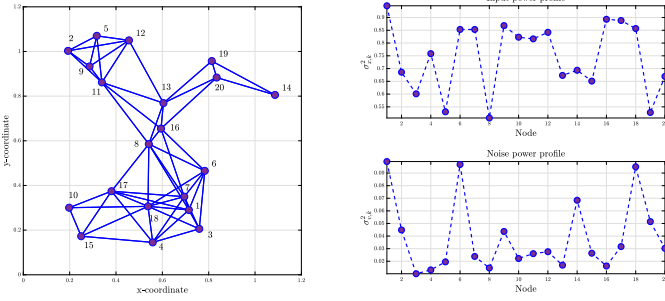


Fig. 2. Power consumption of the ADCs in an adaptive IoT network.

IV. SIMULATION RESULTS

In this section, we assess the estimation performance of the DQA-RLS algorithm for a system identification setup in a network with $N = 20$ nodes. The impulse response of the unknown system has $M = 8$ taps, is generated randomly and normalized to one. The input signals $\mathbf{x}_k(i)$ at each node are

generated by a white Gaussian noise process with variance $\sigma_{x,k}^2$ and quantized using Lloyd-Max quantization scheme to generate $\mathbf{x}_{k,Q}(i)$. The noise samples of each node are drawn from a zero mean white Gaussian process with variance $\sigma_{v,k}^2$. Fig. 3 plots the network structure and the node profiles.



(a) Distributed network structure (b) Input and noise variances
Fig. 3. A wireless network with $N = 20$ nodes.

The simulated mean-square deviation (MSD) learning curves are obtained by ensemble averaging over 100 independent trials. The combining coefficients a_{lk} are computed by the Metropolis rule and $\lambda = 0.98$. Fig. 4 shows the MSD results obtained from simulations for DRLS and DQA-RLS using different numbers of bits. Curve 1 shows the standard DRLS performance assuming full resolution ADCs to perform system identification. Curves 2, 4 and 6 show the MSD evolution of the standard DRLS with signals coarsely quantized with $b=1, 2$ and 3 bits, respectively. Curves 3, 5 and 7 show the MSD performance of the proposed DQA-RLS algorithm that improves the MSD performance for coarsely quantized signals. The performance of the proposed DQA-RLS algorithm is closer to the DRLS while its energy consumption is over 90% less than that of the standard DRLS with full resolution (see Fig. 2).

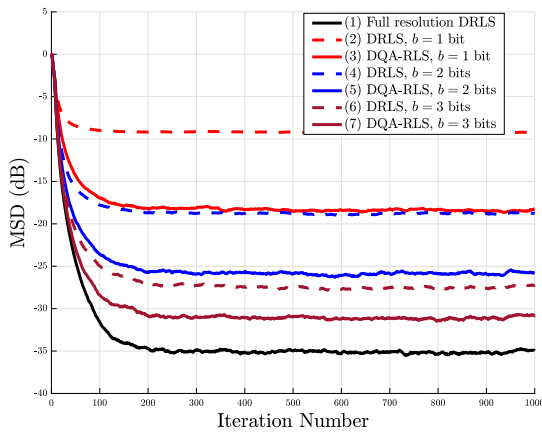


Fig. 4. The MSD curves for the DRLS and DQA-RLS algorithms.

V. CONCLUSION

In this paper, we have proposed an energy-efficient framework for distributed learning and developed the DQA-RLS algorithm for adaptive IoT networks. We have also investigated the DQA-RLS algorithm using low resolution ADCs. The

proposed DQA-RLS algorithm has comparable computational complexity to the standard DRLS algorithm while it greatly reduces the power consumption of the ADCs in the network. Simulations have shown the good performance of DQA-RLS as compared to the standard DRLS for coarsely quantized signals.

REFERENCES

- [1] J. B. Predd, S. B. Kulkarni, and H. V. Poor, "Distributed learning in wireless sensor networks," *IEEE Signal Processing Magazine*, vol. 23, no. 4, pp. 56–69, 2006.
- [2] M. M. Rana, W. Xiang, and E. Wang, "Iot-based state estimation for microgrids," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 1345–1346, 2018.
- [3] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.
- [4] 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.
- [5] 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.
- [6] S. Xu, R. C. de Lamare, and H. V. Poor, "Distributed estimation over sensor networks based on distributed conjugate gradient strategies," *IET Signal Processing*, vol. 10, no. 3, pp. 291–301, 2016.
- [7] T. G. Miller, S. Xu, R. C. de Lamare, V. H. Nascimento, and Y. Zakharov, "Sparsity-aware distributed conjugate gradient algorithms for parameter estimation over sensor networks," in *2015 49th Asilomar Conference on Signals, Systems and Computers*. IEEE, 2015, pp. 1556–1560.
- [8] 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.
- [9] R. H. Walden, "Analog-to-digital converter survey and analysis," *IEEE Journal on selected areas in communications*, vol. 17, no. 4, pp. 539–550, 1999.
- [10] S. Jacobsson, G. Durisi, M. Coldrey, U. Gustavsson, and C. Studer, "Throughput analysis of massive MIMO uplink with low-resolution adcs," *IEEE Transactions on Wireless Communications*, vol. 16, no. 6, pp. 4038–4051, 2017.
- [11] A. Mezghani, M.-S. Khoufi, and J. A. Nossek, "A modified MMSE receiver for quantized MIMO systems," *Proc. ITG/IEEE WSA, Vienna, Austria*, pp. 1–5, 2007.
- [12] L. T. N. Landau, M. Dörpinghaus, R. C. de Lamare, and G. P. Fettweis, "Achievable rate with 1-bit quantization and oversampling using continuous phase modulation-based sequences," *IEEE Transactions on Wireless Communications*, vol. 17, no. 10, pp. 7080–7095, Oct 2018.
- [13] J. J. Bussgang, "Crosscorrelation functions of amplitude-distorted gaussian signals. research lab. electron," *MIT, Cambridge, MA, USA, Tech. Rep*, 1952.
- [14] S. Lloyd, "Least squares quantization in PCM," *IEEE transactions on information theory*, vol. 28, no. 2, pp. 129–137, 1982.
- [15] J. Max, "Quantizing for minimum distortion," *IRE Transactions on Information Theory*, vol. 6, no. 1, pp. 7–12, 1960.
- [16] Y. Li, C. Tao, G. Seco-Granados, A. Mezghani, A. L. Swindlehurst, and L. Liu, "Channel estimation and performance analysis of one-bit massive MIMO systems," *IEEE Transactions on Signal Processing*, vol. 65, no. 15, pp. 4075–4089, 2017.
- [17] Ali H Sayed, *Fundamentals of adaptive filtering*, John Wiley & Sons, 2003.
- [18] O. Orhan, E. Erkip, and S. Rangan, "Low power analog-to-digital conversion in millimeter wave systems: Impact of resolution and bandwidth on performance," in *2015 Information Theory and Applications Workshop (ITA)*. IEEE, 2015, pp. 191–198.
- [19] R. Ratasuk, B. Vejlgaard, N. Mangalvedhe, and A. Ghosh, "Nb-iot system for m2m communication," in *2016 IEEE wireless communications and networking conference*. IEEE, 2016, pp. 1–5.
- [20] H. Chung, A. Rylyakov, Z. T. Deniz, J. Bulzacchelli, G. Wei, and D. Friedman, "A 7.5-gs/s 3.8-enob 52-mw flash adc with clock duty cycle control in 65nm cmos," in *2009 Symposium on VLSI Circuits*. IEEE, 2009, pp. 268–269.