Distributed Energy-Aware Diffusion Least Mean Squares: Game-Theoretic Learning

Omid Namvar Gharehshiran, Vikram Krishnamurthy, *Fellow, IEEE*, and George Yin, *Fellow, IEEE*

***Abstract*—This paper presents a game-theoretic approach to node activation control in parameter estimation via diffusion least mean squares (LMS). Nodes cooperate by exchanging estimates over links characterized by the connectivity graph of the network. The energy-aware activation control is formulated as a noncooperative repeated game where nodes autonomously decide when to activate based on a utility function that captures the trade-off between individual node's contribution and energy expenditure. The diffusion LMS stochastic approximation is combined with a game-theoretic learning algorithm such that the overall energy-aware diffusion LMS has two timescales: the fast timescale corresponds to the game-theoretic activation mechanism, whereby nodes distributively learn their optimal activation strategies, whereas the slow timescale corresponds to the diffusion LMS. The convergence analysis shows that the parameter estimates weakly converge to the true parameter across the network, yet the global activation behavior along the way tracks the set of correlated equilibria of the underlying activation control game.**

***Index Terms*—Adaptive networks, correlated equilibrium, diffusion LMS, distributed estimation, game theory, stochastic approximation.**

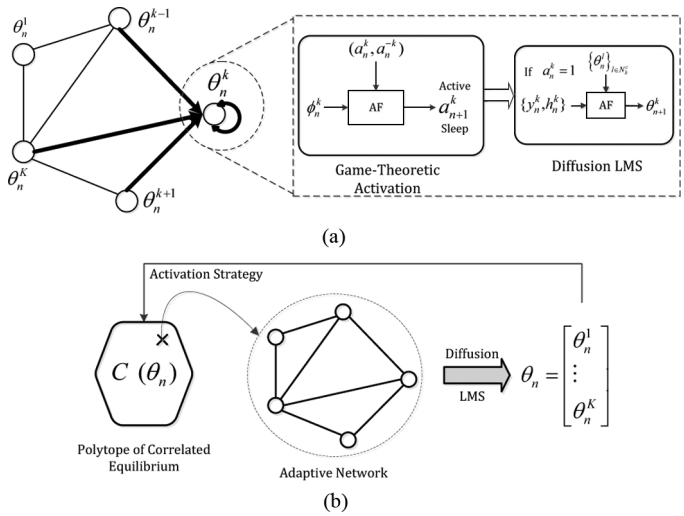

Fig. 1. Energy-aware diffusion LMS: (a) Local behavior, (b) Global behavior.

I. Introduction

THIS paper deals with a novel game-theoretic mechanism by which nodes learn how to activate diffusion least mean squares (LMS) [1], [2]. The diffusion LMS aims for a group of nodes to estimate/track an unknown common parameter by taking noisy measurements and relying solely on in-network processing. Due to limited communication capability, nodes can exchange information only with neighbors determined by the connectivity graph. At successive times, each node: (i) exchanges estimates with neighbors and fuses the collected data via a pre-specified combiner function [3]; (ii) uses the fused data and local measurements to refine its estimate via an LMS-type adaptive filter. Performing data fusion on the same timescale as the measurements enables the adaptive network to respond in real-time to temporal and spatial evolution of the data statistics [1], [4]. Due to measurement noise suppression through ensemble averaging, such diffusion LMS schemes improve the estimation performance, yet yield savings in computation, communication and energy expenditure.

Can one make such adaptive networks even more power-efficient by allowing each node to activate diffusion LMS only when its contribution outweighs the activation cost? Consistent with the decentralized essence of the diffusion LMS, we resort to distributed activation mechanisms whereby nodes autonomously make activation decision based on optimizing a utility function. This utility function captures the trade-off between the "value" of the data that activation of each node provides and the costs associated with it. Nodes do not take measurements and update estimates if they choose to sleep, nor do they exchange and fuse neighboring estimates. Benefits from developing such smart data sensing protocols are threefold: Staying sleep yields savings in energy, communication and computational resources. Naturally, there exists a trade-off between the convergence rate and how aggressively nodes activate the diffusion LMS. This allows the network controller to adjust nodes' utilities such that the desired estimation/convergence expectations (suited to the particular application) are met.

The intertwined role of adaptation and learning, crucial to the self-configuration of nodes, makes game-theoretic learning an appealing theoretical framework, wherein sophisticated global behavior can be guaranteed as the long-run outcome of nodes locally groping for optimality. In this paper, we devise an energy-aware diffusion LMS by equipping nodes with a game-theoretic learning algorithm that prescribes nodes when to activate; see Fig. 1(a).

Manuscript received November 10, 2012; revised February 25, 2013; accepted May 30, 2013. Date of publication June 05, 2013; date of current version September 11, 2013. The guest editor coordinating the review of this manuscript and approving it for publication was Prof. Sean Meyn.

O. N. Gharehshiran and V. Krishnamurthy are with the Department of Electrical and Computer Engineering, University of British Columbia, Vancouver V6T 1Z4, Canada (e-mail: omidn@ece.ubc.ca; vikramk@ece.ubc.ca).

G. Yin is with the Department of Mathematics, Wayne State University, Detroit, Michigan MI 48202 USA (e-mail: gyin@math.wayne.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSTSP.2013.2266318

Main Results: Our main results in this paper are as follows:

1) An Ordinary Differential Equation Approach for Stability Analysis of Diffusion LMS: Let $\boldsymbol{\theta}_n \triangleq \mathrm{col}\{\theta_n^1, \ldots, \theta_n^K\}$ denote the vector of estimates for all nodes at time n and $\overline{\theta}$ represent the true parameter. We show here that, by properly rescaling the periods at which measurements and data fusion take place, one can reformulate the diffusion LMS algorithm proposed in [1], which reads

$$\begin{aligned} \theta_n^k &= \boldsymbol{c}\boldsymbol{\theta}_{n-1} + \mu^k h_n^{k\prime}\big[y_n^k - h_n^k \boldsymbol{c}\boldsymbol{\theta}_{n-1}\big], \\ \boldsymbol{c} &= (c_1, \ldots, c_K),\ 0 \le c_k \le 1, \text{ and } \sum_{k=1}^{K} c_k = 1, \end{aligned} \tag{1}$$

as a classical stochastic approximation algorithm with small step-size of the form

$$\theta_n^k = \theta_{n-1}^k + \varepsilon g\big(\boldsymbol{\theta}_{n-1}, y_n^k\big), \qquad 0 < \varepsilon \ll 1. \tag{2}$$

In (1), $(\cdot)'$ denotes the transpose operator. This new formulation allows us to use the ordinary differential equation (ODE) method [5], [6] (perhaps the most powerful technique in the analysis of stochastic approximation or recursive stochastic algorithms) to show stability of diffusion LMS by affirming asymptotic stability of the origin for an ODE associated with the deviation error $\widetilde{\boldsymbol{\theta}}_n = \boldsymbol{\theta}_n - \overline{\boldsymbol{\theta}}$. This in turn implies asymptotic consistency of the small step-size diffusion LMS algorithm in (2). This approach brings about several advantages: (i) simpler derivation of known results can be obtained for the diffusion LMS; (ii) one can consider models with correlated noise (as opposed to [1], [2], [4]); and (iii) the weak convergence methods [6, Chapter 8] can be applied in the case of time-varying true parameter.

2) Game-Theoretic Formulation for the Energy-Aware Activation Control: Energy-aware activation of the diffusion LMS is formulated as a noncooperative repeated game with neighborhood structure where nodes repeatedly face the decision as whether to activate or sleep. Associated with each decision, there corresponds a reward/penalty that captures the trade-off between the "potential value" of the node contribution and its energy cost. The set of *correlated equilibria* [7] is introduced as the solution for this game.

3) Game-Theoretic Learning Mechanism for Energy-Aware Diffusion LMS: A novel two time-scale stochastic approximation algorithm is proposed that combines the diffusion LMS (slow timescale) with a game-theoretic learning algorithm (fast timescale) for activation control; see Fig. 1(a). The game-theoretic activation mechanism is essentially a non-linear adaptive filter of regret-matching type [8], [9] that prescribes the node whether to "activate" or "sleep" according to a randomized policy being updated based on perceived utilities. The proposed energy-aware diffusion LMS is scalable, robust to delays, and requires minimal message passing among nodes. Our results show that the proposed algorithm is asymptotically consistent yet the global activation behavior along the way tracks the set of *correlated equilibria* [7] of the underlying activation control game; see Fig. 1(b). Correlated equilibrium is a generalization of Nash equilibrium and describes a condition of competitive optimality. It is, however, more preferable for online adaptive learning in distributed systems with tight computation/energy constraints due to structural and computational simplicity; see Section III-B for a discussion.

The convergence analysis is carried out borrowing techniques from stochastic averaging theory [5], [6], two-timescale systems [6] and stability of differential inclusions. It is shown that the limiting process representing the global behavior of the network is a singularly perturbed coupled system of a differential inclusion and an ODE. Intuitively, the activation mechanism (fast timescale) views the diffusion LMS (slow timescale) as quasi-static while the diffusion LMS views the activation mechanism as almost equilibrated; see Fig. 1(b).

4) Numerical Example: Finally, simulation results verify theoretical findings and illustrate the performance of the proposed scheme, in particular, the trade-off between the performance metrics and energy savings.

Literature: The recent past has witnessed a growing interest in employing game-theoretic methods for designing self-configurable adaptive networks [10]. The literature on energy saving schemes for sensor networks mostly focuses on how to transmit information, rather than how to sense the environment; see [11]. There are only few works (e.g., [12], [13]) that propose game-theoretic methods for energy-efficient data acquisition. However, none explicitly study direct interaction of such energy saving schemes with the parameter estimation algorithm. This differentiates our work.

In the literature, energy conservation arguments and error variance analysis are widely used to prove stability for diffusion LMS and its generalizations; see [14] for an exposition. This paper, for the first time, re-derives such stability results using the well-known ODE approach. The game-theoretic learning algorithm also differs from [8], [9] in that nodes only observe neighbors' actions. Nodes are, in fact, oblivious to the existence of other nodes except neighbors in the network, yet the global behavior converges to the polytope of approximate correlated equilibria. In contrast to [8], [9] which assume a static game model, the proposed learning algorithm allows evolution of the game in the same timescale as the regret-matching algorithm [15]. This is crucial since the activation control game is parameterized by instantaneous estimates of the nodes across the network.

Organization: The rest of this paper is organized as follows: Section II formulates the parameter estimation problem, presents the diffusion LMS, and studies its asymptotic stability via the ODE method. Section III then provides an overview of the design principles for the energy-aware diffusion LMS. In Section IV, we introduce the activation control game, present and elaborate on the proposed energy-aware diffusion LMS algorithm and define the global performance quantities. In Section VI, we give a detailed convergence analysis for the proposed algorithm. Finally, numerical examples are provided in Section VII before the concluding remarks in Section VIII.

II. Diffusion LMS

This section describes parameter estimation via diffusion least mean squares (LMS) and shows how one can reformulate the diffusion LMS algorithm (1), proposed in [1], as a stochastic approximation algorithm of the form (2).

This new formulation enables employing the powerful ordinary differential equation (ODE) method [6] to analyze stability. Section II-A describes the parameter estimation problem in linear regression model. Section II-B describes the standard diffusion LMS [1]. Section II-C then reformulates it as a stochastic approximation algorithm of the form (2). This sets the stage for Section II-D that proves asymptotic consistency by means of the ODE method.

A. Centralized Parameter Estimation Problem

Consider a set of K nodes $\mathcal{K} = \{1, \ldots, K\}$ spread over some geographic region with the common objective to perform real-time parameter estimation in the local regression model:

$$y_n^k = h_n^k \overline{\theta} + v_n^k, \quad k = 1, \ldots, K. \tag{3}$$

Here, $\overline{\theta}$ is the $M \times 1$ unknown parameter, $\{h_n^k\}$ the sequence of $1 \times M$ random vector of regressors locally accessible to each node k, y_n^k the scalar local measurements and $\{v_n^k\}$ the sequence of zero-mean local measurement noise uncorrelated with $\{h_n^k\}$. Such linear models are well-suited to approximate input-output relations for many practical applications [16]. The centralized parameter estimation problem in the linear least mean square (LMS) sense can then be formulated as follows: Let

$$\begin{array}{ll} H_n^c \triangleq \mathrm{col}\{h_n^1, \ldots, h_n^K\} & (K \times M) \\ Y_n \triangleq \left[y_n^1, \ldots, y_n^K\right]^T & (K \times 1) \end{array}.$$

The network of nodes then seeks $\overline{\theta}$ that solves

$$\min_{\theta} \ \mathbb{E}\left\|Y_n - H_n^c \theta\right\|^2, \tag{4}$$

where $\mathbb{E}$ and $\|\cdot\|$ denote the expectation and Euclidean norm operators, respectively. Define the correlation $R_h^c \triangleq \mathbb{E}\, H_n^{c\prime} H_n^c$ and cross-correlation $R_{yh}^c \triangleq \mathbb{E}\, H_n^{c\prime} Y_n$. Recall that $(\cdot)'$ denotes the transpose operator[1] and note the second-order moment $R_h^k \triangleq \mathbb{E}\, h_n^{k\prime} h_n^k$ is allowed to vary across nodes. It is well-known that the optimal solution $\overline{\theta}$ satisfies the orthogonality condition $\mathbb{E}\, H_n^{c\prime}(H_n^c \overline{\theta} - Y_n) = 0$, which can also be expressed by the solution to the normal equations [16]

$$R_h^c \overline{\theta} = R_{yh}^c. \tag{5}$$

The goal is to develop a *distributed* stochastic approximation algorithm for parameter estimation that ensures the sequence of parameter estimates θ_n^k converge to (or track) the true parameter $\overline{\theta}$ ($\overline{\theta}_n$). The diffusion LMS [1] is one such algorithm that adopts a peer-to-peer diffusion protocol to implement cooperation among individual nodes. This cooperation leads to savings in communications and energy resources and enables the network to respond in real-time to the temporal and spatial variations in the statistical profile of the data [1]. We thus focus on diffusion LMS as a decentralized solution to approximate $\overline{\theta}$ in (5) in the rest of this paper.

[1]For complex-valued signals, $(\cdot)'$ is replaced with $(\cdot)^*$ that denotes complex conjugate-transpose.

B. Standard Diffusion LMS

It is instructive to start with the diffusion LMS algorithm proposed in [1]. We then continue to reformulate it as a stochastic approximation of the form (2). Before proceeding further, we spell out the conditions imposed on the measurement model (3) in [1]:

(C1) The sequence $\{h_n^k, v_n^k\}$ is temporally white and spatially independent, i.e.,

$$\begin{aligned} \mathbb{E} h_n^{k\prime} h_{n'}^l &= R_d^k \cdot \delta_{nn'} \cdot \delta_{kl}, \\ \mathbb{E} v_n^{k\prime} v_{n'}^l &= \sigma_{v,k}^2 \cdot \delta_{nn'} \cdot \delta_{kl}, \end{aligned} \tag{6}$$

where R_d^h is positive-definite and δ_{ij} is the Kronecker delta function: $\delta_{ij} = 1$ if $i = j$ and 0 otherwise. The noise sequence $\{v_n^k\}$ is further uncorrelated with the regression data $\{h_{n'}^l\}$ for all $\{n, n', k, l\}$, i.e.,

$$\mathbb{E} h_{n'}^{l\prime} v_n^k = 0. \tag{7}$$

The diffusion LMS is simply an LMS-type adaptive filter with a cooperation strategy that adopts a peer-to-peer diffusion protocol. Due to limited communication capabilities, each node k can only communicate with neighbors determined by the connectivity graph of the network $\mathcal{G} = (\mathcal{E}, \mathcal{V})$. Here, $\mathcal{G}$ is an undirected graph, where $\mathcal{V} = \mathcal{K}$ is the set of nodes and

$$(k, l) \in \mathcal{E} \text{ if there exists a link between nodes } k \text{ and } l. \tag{8}$$

We assume the graph $\mathcal{G}$ is simple, i.e., contains no self loops. The *open* and *closed neighborhoods* for each node k are then defined by $\mathcal{N}_k \triangleq \{l \in \mathcal{V}; (k, l) \in \mathcal{E}\}$ and $\mathcal{N}_k^c \triangleq \mathcal{N}_k^c \cup \{k\}$, respectively. Accordingly, we define the degree of node k as $d_{\mathcal{G}}(k) \triangleq \left|\mathcal{N}_k^c\right|$, where $|\cdot|$ denotes the cardinality operator.

Remark 2.1: For simplicity of presentation, we assume the connectivity graph $\mathcal{G}$ is fixed and strongly connected, i.e., there exists a path between each pair of nodes. Intermittent link/node failures can be captured by a random graph model where the probability that two nodes are connected is simply the probability of successful transmission times the indicator function that shows the two nodes are neighbors in the underlying fixed graph. Motivated by wireless sensor network applications, the link failures can be spatially correlated (due to the interference among wireless communication channels), however, are independent over time. In this case, mean connectedness of the random graph is sufficient, i.e., $\lambda_2(\overline{L}) > 0$, where $\overline{L} \triangleq \mathbb{E} L_n$ and L_n denotes the Laplacian of the random graph process $\mathcal{G}_n$; see [17] for details.

Nodes exchange estimates over the graph $\mathcal{G}$ at each period. Each node k then fuses the local estimates $\{\theta_n^l\}_{l \in \mathcal{N}_k^c}$ by means of a linear combiner:

$$\phi_n^k = \sum_{l \in \mathcal{N}_k^c} c_{kl} \theta_n^l. \tag{9}$$

Note that, since each node k has a different neighborhood, the fusion rule (9) helps fuse data across the network into node k.

The coefficients c_{kl} give rise to a stochastic matrix $C = [c_{kl}]$ satisfying

$$\begin{aligned} 0 \le\ c_{kl} \le 1, \quad \forall k, l \in \mathcal{K}, \\ C\mathbb{1}_K = \mathbb{1}_K, \quad \text{and} \quad C' = C, \end{aligned} \tag{10}$$

where $\mathbb{1}_K = [1, \ldots, 1]_{K\times 1}$. Each row k of C simply gives the weights by which each node k fuses the local estimates $\{\theta_n^l\}_{l\in\mathcal{N}_k^c}$. Therefore, it is referred to as the *weight matrix*.

The fused estimates are then promptly fed back into the local LMS-type adaptive filter. More precisely, the diffusion LMS requires each node to run a stochastic approximation algorithm of the form

$$\begin{aligned} \theta_n^k &= \phi_{n-1}^k + \mu h_n^{k\prime}\left[y_n^k - h_n^k \phi_{n-1}^k\right], \\ \phi_{n-1}^k &= \sum_{l\in\mathcal{N}_k^c} c_{kl}\theta_{n-1}^l, \end{aligned} \tag{11}$$

with local step-sizes $\mu > 0$. The estimate at each node is thus a function of both its temporal data as well as the spatial data across the neighbors. This enables the adaptive network to respond in real-time to the temporal and spatial variations in the statistical profile of the data [1]. It is well-known that, to ensure mean-square stability, the step-size must be sufficiently small and satisfy $0 < \mu < 2/\lambda_{\max}(R_h^k)$[1], [2], [14]. In view of (11), however, making the step-size too small leads to performing data assimilation and fusion of neighboring estimates on different timescales, where the former is run on the slow timescale and the latter constitutes the fast timescale.

The local updates (11) together give rise to a global state-space representation. Define

$$\begin{aligned} Y_n &\triangleq \left[y_n^1, \ldots, y_n^K\right]^T && (K\times 1), \\ V_n &\triangleq [v_n^1, \ldots, v_n^K]^T && (K\times 1), \\ \overline{\boldsymbol{\theta}} &\triangleq \left[\mathrm{col}\{I_M\overline{\theta}, \ldots, I_M\overline{\theta}\}\right] && (KM\times 1), \\ H_n &\triangleq \left[\mathrm{diag}\{h_n^1, \ldots, h_n^K\}\right] && (K\times KM), \end{aligned}$$

where I_M denotes the $M\times M$ identity matrix. Then, the linear measurement model (3) can be written as

$$Y_n = H_n\overline{\boldsymbol{\theta}} + V_n. \tag{12}$$

Further, let

$$\begin{aligned} S(\mu) &\triangleq \mu I_{KM} && (KM\times KM), \\ \boldsymbol{\theta}_n &\triangleq \mathrm{col}\{\theta_n^1, \ldots, \theta_n^K\} && (KM\times 1). \end{aligned}$$

The global diffusion LMS update can then be written in a more compact state-space form

$$\boldsymbol{\theta}_n = F\boldsymbol{\theta}_{n-1} + SH_n'\big(Y_n - H_nF\boldsymbol{\theta}_{n-1}\big), \tag{13}$$

where

$$F \triangleq C\otimes I_M \quad (KM\times KM), \tag{14}$$

and $\otimes$ denotes the Kronecker product. Further, C is a stochastic matrix as defined in (10).

C. Revised Diffusion LMS

Here, the objective is to reformulate the standard diffusion LMS (11) as a classical stochastic approximation algorithm $\theta_n^k = \theta_{n-1}^k + \varepsilon g\big(\boldsymbol{\theta}_{n-1}, y_n^k\big)$ with small step-size $0 < \varepsilon \ll 1$. The advantage of this new formulation is threefold: (i) both data assimilation and fusion takes place on the same timescale; (ii) simpler derivation of the known results can be obtained by employing the powerful ordinary differential equation method [5], [6]; (iii) one can use weak convergence methods [6, Chapter 8] to show how responsive the diffusion LMS is to the time-variations of the true parameter. To proceed, we first need to revise the diffusion protocol.

It is instructive to start by commenting on the diffusion protocol in the standard diffusion LMS, given by (9). Consider the data fusion term in the global state-space model (13), i.e.,

$$\boldsymbol{\theta}_n = F\boldsymbol{\theta}_{n-1}, \tag{15}$$

and suppose Δ seconds elapse between two successive iterations. Then, the discrete-time iterates (15) can be conceived as a discretization of the ODE

$$\frac{d\boldsymbol{\theta}}{dt} = \widehat{F}\boldsymbol{\theta}, \tag{16}$$

where $e^{\widehat{F}\Delta} = F$, i.e., $\widehat{F}$ is the matrix logarithm of F normalized by Δ. Here, Δ can be large. Since $\widehat{F} = \lim_{\Delta\to 0}(\exp(\widehat{F}\Delta) - I)/\Delta$,

$$\widehat{F}\mathbb{1}_{KM} = \mathbf{0}, \quad \text{and} \quad f_{ij} \ge 0 \ \text{ for } \ j \ne i. \tag{17}$$

Having the same properties as the generator of a continuous-time Markov chain, we refer to $\widehat{F}$ as the *generator* of the stochastic matrix F. Another way to look at the issue is based on a Taylor expansion argument. To this end, taking Taylor expansion and retaining the first two terms, we have $\exp(\Delta\widehat{F}) = I + \Delta\widehat{F} + o(\Delta)$, where $o(\Delta)\to 0$ as $\Delta\to 0$.

Before proceeding, we note that, to differentiate between the two diffusion LMS algorithms, we denote the step-size of the revised diffusion LMS by ε. It is well-known that, to analyze a stochastic approximation algorithm with a small step-size ε, an interpolated process in intervals of ε is studied to determine the limiting process described by an ODE [6]. Inspired by this idea and to ensure that the data fusion takes place at the same timescale as the assimilation of measurements, we then re-discretize (16) with sampling period ε to obtain the new diffusion protocol

$$\boldsymbol{\theta}_n = B\boldsymbol{\theta}_{n-1}, \tag{18}$$

where

$$B = e^{\widehat{F}\varepsilon} \approx I_{KM} + \varepsilon\,\widehat{F}. \tag{19}$$

In light of the above discussion, the local updates of the revised diffusion LMS can be expressed as

$$\begin{aligned} \theta_n^k &= \theta_{n-1}^k + \varepsilon\left[\varphi_{n-1}^k + h_n^{k\prime}\left[y_n^k - h_n^k\phi_{n-1}^k\right]\right], \\ \varphi_{n-1}^k &= \sum_{l\in\mathcal{N}_k^c} q_{kl}\theta_{n-1}^l,\ \phi_{n-1}^k = \sum_{l\in\mathcal{N}_k^c} w_{kl}\theta_{n-1}^l, \end{aligned} \tag{20}$$

where the new weight matrix $W = [w_{kl}]$ for the diffusion protocol is defined as

$$\begin{gathered} W = I_K + \varepsilon Q, \\ |q_{ij}| \leq 1, \ \ \forall i,j, \ \ q_{ij} \geq 0, \ \ \text{for} \ \ i \neq j, \\ Q\mathbb{1}_K = \mathbf{0}, \ \ \text{and} \ \ Q' = Q. \end{gathered} \tag{21}$$

Note that W is dominated by the diagonal elements representing the weights that nodes put on their own estimate. This is in contrast with the weight matrix C. This is essentially because smaller step-sizes require smaller fusion weights to ensure fusion and measurements are performed at the same timescale.

Remark 2.2: The small step-size diffusion LMS recursion (20) is essentially the same as the standard diffusion LMS, i.e.,

$$\begin{aligned} \theta_n^k &= \phi_{n-1}^k + \varepsilon h_n^{k\prime} \left[y_n^k - h_n^k \phi_{n-1}^k \right], \\ \phi_{n-1}^k &= \sum_{l \in \mathcal{N}_k^c} w_{kl} \theta_{n-1}^l. \end{aligned} \tag{22}$$

It thus incurs no additional computational cost. The formulation in (20) is only needed for our convergence analysis using the ODE method.

Accordingly, the global recursion for the revised diffusion LMS can be written as

$$\boldsymbol{\theta}_n = \boldsymbol{\theta}_{n-1} + S(\varepsilon) \left[\widehat{B} \boldsymbol{\theta}_{n-1} + H_n{}' (Y_n - H_n B \boldsymbol{\theta}_{n-1}) \right], \tag{23}$$

where

$$\begin{aligned} B &\triangleq W \otimes I_M \quad (KM \times KM), \\ \widehat{B} &\triangleq Q \otimes I_M \quad (KM \times KM). \end{aligned} \tag{24}$$

Using a small step-size allows us to relax the independency assumptions (C1) on the sequence $\{h_n^k, v_n^k\}$. Let $\mathcal{F}_n^k$ denote the σ-algebra generated by $\{h_\tau^k, y_\tau^k, \tau < n\}$, and denote the conditional expectation with respect to $\mathcal{F}_n^k$ by $\mathbb{E}_n^k$. We impose the following conditions on the sequence $\{h_n^k, v_n^k\}$ for the revised diffusion LMS:

(C2) For all $k \in \mathcal{K}$:

(i)

$$\left| \sum_{\tau=n}^{\infty} \mathbb{E}_n^k h_\tau^{k\prime} v_\tau^k \right| \leq K. \tag{25}$$

(ii) There exists a symmetric and positive-definite matrix $R_h^k \in \mathbb{R}^{M \times M}$ such that $R_h^k \triangleq \mathbb{E} h_n^{k\prime} h_n^k$, and

$$\left| \sum_{\tau=n}^{\infty} \mathbb{E}_n^k h_\tau^{k\prime} h_\tau^k - R_h^k \right| \leq K. \tag{26}$$

The above condition allows us to work with correlated signals whose remote past and distant future are asymptotically independent. With suitable regularity conditions (such as uniform integrability), a large class of processes $\{h_n, v_n\}$ can be considered. Examples of correlated processes that satisfy (C2) include:

1) Finite-state Markov chains: Assuming the Markov chain $\{x_n\}$ has memory p,

$$\begin{aligned} &\mathbb{E}\left[x_n | x_{n-1}, x_{n-2}, \ldots, x_1\right] \\ &\quad = \mathbb{E}\left[x_n | x_{n-1}, x_{n-2}, \ldots, x_{n-p}\right], \quad \text{for n} > \text{p}, \end{aligned} \tag{27}$$

where $\{x_n\} \in \mathcal{M}_s$ and $\mathcal{M}_s$ is finite.

2) Moving average noise $MA(p)$:

$$x_n = u_n + \sum_{\tau=1}^{p} \gamma_\tau u_{n-\tau}, \tag{28}$$

where $\{u_\tau\}$ are (Gaussian) white noise error terms.

3) Stationary autoregressive processes $AR(p)$

$$x_n = \sum_{i=1}^{p} a_i x_{n-i} + u_n, \tag{29}$$

if the roots of the polynomial

$$z^p + a^1 z^{p-1} + a_p = 0, \tag{30}$$

lie inside the unit circle.

Note further that, to obtain the desired result, the distributions of the sequences $\{h_n^k\}$ and $\{v_n^k\}$ need not be known.

D. Convergence Analysis via Ordinary Differential Equation Method

As is well known since the 1970s, the limiting behavior of a stochastic approximation algorithm (such as the diffusion LMS algorithm) is typically captured by a system of deterministic ODEs—indeed, this is the basis of the widely used "ODE Method" for convergence analysis of stochastic approximation algorithms—see [6]. Here, we proceed with the convergence analysis of the diffusion LMS (20) via the ODE method.

Define the tracking error $\widetilde{\boldsymbol{\theta}}_n = \boldsymbol{\theta}_n - \overline{\boldsymbol{\theta}}$. Substituting this in the global diffusion LMS recursion (23) yields

$$\widetilde{\boldsymbol{\theta}}_n = \widetilde{\boldsymbol{\theta}}_{n-1} + S(\varepsilon) \left[\widehat{B} \widetilde{\boldsymbol{\theta}}_{n-1} + H_n{}' \left(V_n - H_n B \widetilde{\boldsymbol{\theta}}_{n-1} \right) \right]. \tag{31}$$

In (31), we used $\widehat{B}\mathbb{1} = \mathbf{0}$; see (21) and (24). We take the ODE approach and define the piecewise constant continuous-time interpolated process associated with $\widetilde{\boldsymbol{\theta}}_n$ as:

$$\widetilde{\boldsymbol{\theta}}^{\varepsilon}(\cdot) = \widetilde{\boldsymbol{\theta}}_n, \quad \text{for} \quad t \in [n\varepsilon, (n+1)\varepsilon). \tag{32}$$

Then, $\widetilde{\boldsymbol{\theta}}^{\varepsilon}(\cdot) \in D([0,\infty) : \mathbb{R}^M)$, the space of functions that are defined in $[0,\infty)$ taking values in $\mathbb{R}^M$, and that are right continuous and have left limits with the use of the Skorohod topology (see [6, p. 228]). Such a topology allows we "wiggle the space and time a bit".

Before proceeding further, let us recall a definition. Weak convergence is a generalization of convergence in distribution to a function space. More precisely, let Z_n and Z be $\mathbb{R}^r$-valued random vectors. We say Z_n *converges weakly* to Z ($Z_n \Rightarrow Z$) if for any bounded and continuous function $f(\cdot)$, $Ef(Z_n) \to Ef(Z)$ as $n \to \infty$. We refer the interested reader to [6, Chapter 7] for further details on weak convergence and related matters. The following theorem provides us with the evolution of the tracking error. It shows that $\widetilde{\boldsymbol{\theta}}_n$ evolves dynamically so that its trajectories follows an ODE. Since the ODE is asymptotically stable, the errors decay exponentially fast to 0 as time grows.

Theorem 2.1: Consider the interpolated process $\widetilde{\boldsymbol{\theta}}^{\varepsilon}(\cdot)$, defined in (32), associated with the diffusion LMS iterates (20). The following assertions hold:

1) Under (C2), $\widetilde{\boldsymbol{\theta}}^{\varepsilon}(\cdot)$ converges weakly to $\widetilde{\boldsymbol{\theta}}(\cdot)$ as $\varepsilon \to 0$ such that the limit is the solution to the ODE

$$\frac{d\widetilde{\boldsymbol{\theta}}(t)}{dt} = \big(\widehat{B} - R_h^d\big)\widetilde{\boldsymbol{\theta}}(t), \quad t \geq 0, \quad \widetilde{\boldsymbol{\theta}}(0) = \widetilde{\boldsymbol{\theta}}_0, \tag{33}$$

where $R_h^d \triangleq \mathrm{diag}\{R_h^1, \ldots, R_h^K\}$ is a $KM \times KM$ matrix, and $\widehat{B}$ is defined in (24).

2) System (33) is globally asymptotically stable.
3) Denote by $\{q(\varepsilon)\}$ any sequence of real numbers satisfying $q(\varepsilon) \to \infty$ as $\varepsilon \to 0$. Then, $\widetilde{\boldsymbol{\theta}}^{\varepsilon}(\cdot + q(\varepsilon))$ converges weakly to $\mathbf{0}$ as $\varepsilon \to 0$.

Proof: For a more rigorous statement of the theorem and proof, see Appendix A. ∎

The above theorem simply asserts that as $\varepsilon \to 0$, the interpolated process of individual node's parameter estimates $\theta^{k,\varepsilon}(\cdot)$ converges weakly to the true parameter $\overline{\theta}$ across the network. Therefore, the diffusion LMS algorithm is consistent. Statement 1) deals with the case that $\varepsilon \to 0$ and n is large, but εn remains bounded, whereas Statement 3) illustrates the asymptotic properties as $\varepsilon \to 0$ followed by $t \to \infty$. Roughly, it indicates that as $\varepsilon \to 0$, $n \to \infty$, and $\varepsilon n \to \infty$, $\widetilde{\boldsymbol{\theta}}_n \to 0$.

Remark 2.3: In case of a slowly time-varying true parameter $\overline{\theta}_n$, it can be shown that, if $\overline{\theta}_n$ evolves on the same timescale as the adaptation rate ε of the diffusion LMS, i.e., the true parameter dynamics are matched to the diffusion LMS stochastic approximation algorithm, $\theta^{k,\varepsilon}(\cdot)$ weakly tracks its variations.

III. Design Principles for an Energy-Aware Diffusion LMS

This section provides a qualitative overview of the design principles deployed in devising an energy-aware diffusion LMS. In Section III-A, we present a big picture of the design process. Given the noncooperative game-theoretic framework used, Section III-B comments on the significance of the correlated equilibrium (as the solution to this game) and its advantages over the well-known Nash equilibrium in adaptive signal processing.

A. Overview of the Approach

Apart from scalability and stabilizing effect on the network [1], deploying cooperative schemes such as the diffusion protocol (9) can lead to energy savings by allowing nodes to sleep while their neighbors are active. Taking into account the spatial-temporal correlation among nodes' measurements, nodes can enter a sleep mode when the energy cost of acquiring data outweighs its value. We take a decentralized solution since it facilitates self-configuration of nodes as well as network robustness and scalability. We equip nodes with pre-configured (or configurable) utility functions and simple adaptive filtering algorithms. The adaptive filters control local functionality only, with minimal message passing, thereby reducing intra-node communication and associated energy costs. While individual nodes posses limited communication and processing capability, it is their coordinated behavior that leads to the manifestation of rational global behavior. The proposed formulation conveys a design guideline: by carefully specifying the utility functions, these networked nodes can be made to exhibit many desired behaviors while individual nodes perform simple local tasks.

Due to the interdependence of nodes behavior, a game-theoretic approach is a natural choice to model their interaction. Each node repeatedly faces a non-cooperative game where the actions are whether to "activate" or "sleep" and, accordingly, receives a reward. When a node is active, it updates its estimate and performs fusion of its own estimate with those received from neighboring nodes, whereas inactive nodes do not update their estimates. Individual nodes rewards' depends not only on the actions of other nodes but also on their estimates of the true parameter. In this case, no pre-computed policy is given; nodes learn their activation policies through repeated play and exchanging information with neighbors. The nodes further adapt their policies as the parameter estimates across the network vary over time. The proposed activation mechanism is a simple non-linear adaptive filtering algorithm based on the recent economics literature on regret-based algorithms [18]. The global activation behavior of nodes is proved to converge to the set of approximate correlated equilibria (See Definition 5.1) of the underlying activation control game if each node individually follows the proposed activation mechanism.

Finally, we embed this activation mechanism into the diffusion LMS algorithm such that the overall energy-aware diffusion LMS forms a two-timescale stochastic approximation algorithm: the fast timescale corresponds to the game-theoretic activation mechanism and the slow timescale is the diffusion LMS. Intuitively, in such two timescale algorithms, the fast timescale will see the slow component as quasi-static while the slow component sees the fast one near equilibrium. In light of the above discussion, the main theoretical finding is that, by each node individually following the proposed energy-aware diffusion LMS algorithm: (i) the estimate at each node tracks the true evolution of the parameter of interest; (ii) at each time, the global activation behavior across the network properly tracks the polytope of approximate correlated equilibria of the underlying game; see Fig. 1(b).

B. Correlated Equilibrium Vs. Nash Equilibrium

The notion of correlated equilibrium [7] is a generalization (superset) of Nash equilibrium and describes a condition of competitive optimality among nodes. It can most naturally be viewed as imposing equilibrium conditions on the joint action profile of nodes (rather than on individual actions as in Nash equilibrium). An intuitive interpretation of correlated equilibrium is coordination in decision-making as described by the following example: Suppose, at the beginning of each period, each node k receives a private *recommendation* a^k exogenously as whether to "activate" or "sleep", where $\boldsymbol{a} = (a^1, \ldots, a^K)$ is drawn from a distribution $\boldsymbol{\pi}$. (Such a joint distribution correlates nodes' actions, hence, the name correlated equilibrium.) Although $\boldsymbol{\pi}$ is known to all nodes, each node is only given instructions for its own activation a^k. A correlated equilibrium results if every node k realizes that the recommendation a^k

is a best-response to the random estimated activity of others, provided that others, as well, follow their recommendations. This implies coordinated activity of nodes once they reach the correlated equilibria set.

The research so far in the signal processing community is mostly focused on the well-known Nash equilibrium, despite several advantages of correlated equilibrium as outlined below: (i) *Realistic:* Correlated equilibrium is realistic in multi-agent learning scenarios since the environment naturally correlates agents' actions. In contrast, Nash equilibrium assumes agents act independently which is rarely true in adaptive learning; (ii) *Structural simplicity:* The correlated equilibria set constitutes a compact convex polyhedron, whereas the Nash equilibria are isolated points at the extrema of this set [19]; (iii) *Computational simplicity:* Computing correlated equilibria only requires solving a linear feasibility problem (linear program with null objective function) that can be solved in polynomial time, whereas computing Nash equilibrium requires finding fixed points; (The activation mechanism in Algorithm 1 is simply an adaptive filtering algorithm that solves this linear feasibility problem in a distributed fashion;) (iv) *Coordination capability:* The definition of correlated equilibrium allows agents to coordinate their actions while still retaining their autonomy. This coordination leads to potentially higher payoffs than if agents take their actions independently (as required by Nash equilibrium) [7]. Finally, online adaptive procedures naturally converge to the correlated equilibria set, whereas the same is not true for Nash equilibria (the so-called law of conservation of coordination [20]).

In light of the above discussion, the correlated equilibrium is arguably the most natural attractive set for distributed, online adaptive algorithms (such as the one considered here).

IV. Game-Theoretic Activation Control For Diffusion LMS

We have shown in Section II (Theorem 2.1) that the diffusion LMS algorithm can be analyzed using the standard ODE approach. Therefore the powerful machinery for analysis of such stochastic approximation algorithms apply directly. For example, under conditions given in [6, Chapter 7], weak convergence can be proved for (correlated) Markovian noise. In the remainder of the paper, we add a distributed game-theoretic learning functionality for energy-aware activation of the diffusion LMS. Section IV-A formulates the game-theoretic model by which nodes make activation decisions. In Section IV-B, we present the energy-aware diffusion LMS algorithm. Finally, Section IV-C provides some intuition on the game-theoretic activation control mechanism.

A. Activation Control Game

The objective is to combine the diffusion LMS with a game-theoretic energy-aware activation mechanism which enables nodes to conserve energy by sleeping in certain periods. More precisely, each node runs local updates of the form:

$$\begin{aligned}\theta_n^k &= \theta_{n-1}^k + \varepsilon\left[\varphi_{n-1}^k + h_n^{k\prime}\left[y_n^k - h_n^k \phi_{n-1}^k\right]\right]\cdot I_{\{a_n^k=1\}},\\ \varphi_{n-1}^k &= \sum_{l\in\mathcal{N}_k^c} q_{kl}\theta_{n-1}^l,\ \phi_{n-1}^k = \sum_{l\in\mathcal{N}_k^c} w_{kl}\theta_{n-1}^l,\end{aligned} \tag{34}$$

where $I_{\{\cdot\}}$ denotes the indicator function and $a_n^k \in \{\text{"sleep"}, \text{"activate"}\}$ denotes the node's action determined by the game-theoretic learning functionality. Nodes in the sleep mode do not take measurements, nor do they fuse data from neighbors and update their estimates; see (34). Therefore, they save energy in both sensing and transmitting their estimate to neighbors. The key feature that characterizes our study as a game is the interdependence of individual node's behavior. The usefulness of a node's information, channel quality, required packet transmission energy all depend on the activity of other nodes in the network.

The problem of each node k is to successively pick action a_n^k from the set $\mathcal{A} = \{0\ (\text{sleep}), 1\ (\text{activate})\}$ to strategically optimize a utility function. This utility function captures the trade-off between energy expenditure and the "value" of node's contribution. Let $\mathcal{A}^{\mathcal{K}} = \times_{k\in\mathcal{K}}\mathcal{A}^k$ denote the space of all possible joint action profiles $\boldsymbol{a} = \left(a^1, \ldots, a^K\right)$ of all nodes. Following the common notation in game theory, $\boldsymbol{a}$ can be rearranged as $\boldsymbol{a} = \left(a^k, \boldsymbol{a}^{-k}\right)$. Let further $\mathcal{M}_{\boldsymbol{\theta}}$ denote the state-space of $\boldsymbol{\theta} = (\theta^1, \ldots, \theta^k)$ and $\mathcal{S}_k = \mathcal{K}\backslash\mathcal{N}_k^c$ represent the set of non-neighbors of node k. The utility for each node k is a bounded function $U^k : \mathcal{A}^{\mathcal{K}} \times \mathcal{M}_{\boldsymbol{\theta}} \to \mathbb{R}$ and is comprised of a *local* and a *global* term:

$$U^k\left(a^k, \boldsymbol{a}^{-k}; \boldsymbol{\theta}\right) = U_{\text{loc}}^k\left(a^k, \boldsymbol{a}^{\mathcal{N}_k}; \boldsymbol{\theta}\right) + U_{\text{glob}}^k\left(a^k, \boldsymbol{a}^{\mathcal{S}_k}\right), \tag{35}$$

where $\boldsymbol{a}^{-k} = \left(\boldsymbol{a}^{\mathcal{N}_k}, \boldsymbol{a}^{-\mathcal{N}_k^c}\right)$, $\boldsymbol{a}^{\mathcal{N}_k}$ and $\boldsymbol{a}^{\mathcal{S}_k}$ denote the joint action profile of neighbors $\mathcal{N}_k$ and non-neighbors $\mathcal{S}_k$, respectively. The utility function for each node k further depends on the instantaneous parameter estimates $\boldsymbol{\theta}_n$. The *local* utility function $U_{\text{loc}}^k(\cdot)$ captures the trade-off between the value of the measurements collected by node k and the energy and costs associated with it. We assume that each node is only capable of low-power communication[2]. The neighbors of each node are thus located within a pre-specified range. If too many of node k's neighbors activate simultaneously, excessive energy is consumed due to the spatial-temporal correlation of node measurements. That is, the data collected by node k is less valuable. Additionally, the probability of successful transmission reduces due to channel congestion. (Interchangeably, to keep success rate fixed, the required packet transmission energy should increase.) On the other hand, if too few of node k's neighbors activate, their fused estimates lack "innovation." The *local* utility of node k is then given by: [see (36), shown at the bottom of the next page], where

$$\eta^k(a^k, \boldsymbol{a}^{\mathcal{N}_k}) = a^k + \sum_{l\in\mathcal{N}_k} a^l.$$

In (36), $K_{l,1}$ and γ_l are the pricing parameters related to the 'reward' associated with data collected by node k; $K_{l,2}$ is the

[2]The ZigBee/IEEE 802.15.4 standard is currently a leading choice for low-power communication in wireless sensor networks. It employs a CSMA/CA scheme for multiple access data transmission. In networks with tight energy constraints, the non-beacon-enabled (unslotted CSMA/CA) mode is more preferable as the node receivers do not need to switch on periodically to synchronize to the beacon.

pricing parameter related to the energy costs associated with activation $E_{\rm Act}$ and broadcasting measurements $E_{\rm Tx}(\cdot)$[3]. In addition, $s(\eta^k)$ denotes the probability of successful transmission[4]. Higher activity η^k of nodes neighboring node k lowers its local utility due to: (i) reducing success of transmission attempts, and (ii) inefficient usage of energy resources (due to spatial-temporal correlation of measurements). Nodes are thus motivated to activate when majority of neighbors are in the sleep mode and/or its estimate θ^k is far from the local aggregated estimate ϕ^k.

The *global* utility function $U^k_{\rm glob}(\cdot)$ concerns the connectivity of the network and diffusion of estimates in a larger geographic region (than each node's neighborhood). Let $\mathcal{R}^k_r$ denote the set of nodes within radius r from node k excluding the neighborhood $\mathcal{N}^c_k$. Define the number of active nodes in $\mathcal{R}^k$ by

$$\zeta^k\big(a^k, \boldsymbol{a}^{\mathcal{S}_k}\big) = a^k + \sum_{l\in\mathcal{R}^k_r} a^l.$$

The *global* utility of node k is then given by

$$U^k_{\rm glob}\big(a^k, \boldsymbol{a}^{\mathcal{S}_k}\big) = \begin{cases} 0, & a^k = 0 \text{ (sleep)},\\ K_g\big(e^{-\gamma_g \zeta^k}\big), & a^k = 1 \text{ (activate)}. \end{cases} \quad (37)$$

In (37), K_g and γ_g are the pricing parameters. Higher ζ^k lowers the global utility due to: (i) less importance of node k's contribution to the diffusion of estimates across the network and keeping connectivity in $\mathcal{R}^k_r$, and (ii) reducing the success of transmission for other nodes in the neighborhood which affects global diffusion of estimates. Node k is thus motivated to activate when majority of the nodes in its geographic region $\mathcal{R}^k_r$ are in the sleep mode.

Each node k realizes $\eta^k_n = \eta^k(a^k, \boldsymbol{a}^{\mathcal{N}_k}_n)$ as a consequence of receiving estimates $\{\theta^l_n\}_{l\in\mathcal{N}_k}$ from neighbors, therefore, is able to evaluate its local utility at each period n. Nodes, however, do not observe the actions of non-neighbors; therefore, nodes do not realize $\zeta^k_n = \zeta^k(a^k_n, \boldsymbol{a}^{\mathcal{S}_k}_n)$ and are unable to evaluate global utilities. To this end, some sort of centralization is required by incorporating cluster heads that monitor activity of the nodes in their locale and deliver global utilities to the nodes; see Remark 4.1. Hereafter, $U^k_{{\rm glob},n}(a^k_n) = U^k_{\rm glob}\big(a^k_n, \boldsymbol{a}^{\mathcal{S}_k}_n\big)$ denotes the realized global utility for node k at period n by choosing action a^k_n. Note that $U^k_{{\rm glob},n}(\cdot)$ is a time-variant function due to the time-varying actions of non-neighbors $\boldsymbol{a}^{\mathcal{S}_k}_n$.

The contribution of each node to the local parameter estimate, the success of transmissions within neighborhoods and the network connectivity all depend on the activity of other nodes in the network. Due to such interdependence, the activation control problem can be formulated as a noncooperative repeated game with neighborhood structure

$$\boldsymbol{G} = \Big(\mathcal{K}, \big(\mathcal{A}^k\big)_{k\in\mathcal{K}}, \mathcal{G}, \big(U^k\big)_{k\in\mathcal{K}}, \boldsymbol{\theta}\Big). \quad (38)$$

Here, $\mathcal{K}$ is the set of players of the game, $\mathcal{A}^k = \{0 \text{ (sleep)}, 1 \text{ (activate)}\}$ (hereafter denoted by $\mathcal{A}$) is the set of actions, U^k denotes the payoff to each player k (defined in (35)–(37)), $\mathcal{G}$ is the connectivity graph of the network, and $\boldsymbol{\theta} = (\theta^1, \ldots, \theta^K)$ is the K-tuple of nodes' estimates, each obtained locally via the diffusion LMS recursion.

Remark 4.1 (Decentralized Structure): Note that each node k only requires the cumulative activity of others in its geographic region ζ^k to evaluate its global utility. Adopting a ZigBee/IEEE 802.15.4 CSMA/CD protocol for broadcasting estimates across neighbors, ζ^k can be estimated by tracking the proportion of successful clear channel assessment (CCA) attempts. The interested reader is referred to [12, Sec. II-C], for a discussion and an algorithm for estimating node activity. This eliminates the need for inclusion of cluster heads to deliver global utilities to the nodes.

B. Energy-Aware Diffusion LMS

In light of the game-theoretic model for the activation control of diffusion LMSs, we proceed here to present the energy-aware diffusion LMS algorithm. The proposed algorithm can be simply described as a non-linear adaptive filter run in the fast timescale (step-size ρ, which governs node activation) coupled with the diffusion LMS run by nodes in the slow timescale (step-size $\varepsilon = \mathcal{O}(\rho^2)$, which carries out the parameter estimation task).

Suppose each node k has access to time realizations of $\{y^k_n, h^k_n\}$. Define $|x|^+ = \max\{0, x\}$ and let $I_{\{\cdot\}}$ denote the indicator function. Each node k runs the algorithm summarized below independently:

Algorithm 1: *Diffusion LMS with Energy-Aware Activation Control:*

Step 0) **Initialization:** Set the tunable parameter $0 < \delta < 1$ and choose the adaptation rates $0 < \varepsilon, \rho \ll 1$ such that $\varepsilon = \mathcal{O}(\rho^2)$. Initialize $\theta^k_0 = \phi^k_0 = \mathbf{0}_{M\times 1}$, $\boldsymbol{\psi}^k_0 = [1/2, 1/2]'$, $\boldsymbol{\alpha}^k_0 = \boldsymbol{\beta}^k_0 = \mathbf{0}_{2\times 2}$ for all nodes $k \in \mathcal{K}$.

For $n = 1, 2, \ldots$, repeat:

Step 1) **Node Activation:** Select action $a^k_n \sim \boldsymbol{\psi}^k_n$: [see (39), shown at the bottom of the next page]. In (39), a^k_{n-1} denotes the action picked in the last period, and ξ^k is chosen such that $\xi^k > \sum_{\substack{j\in\mathcal{A}\\ j\neq a^k_{n-1}}} |\alpha^k_{n-1}(a^k_{n-1}, j) + \beta^k_{n-1}(a^k_{n-1}, j)|^+$.

Step 2) **Diffusion LMS:**

$$\theta^k_n = \begin{cases} \theta^k_{n-1}, & a^k_n = 0,\\ \phi^k_{n-1} + \varepsilon h^{k\prime}_n \big[y^k_n - h^k_n \phi^k_{n-1}\big], & a^k_n = 1. \end{cases} \quad (40)$$

[3] $E_{\rm Tx}(\cdot)$ and $E_{\rm Act}$ are given in [12, p. 6099] for the Chipcon CC2420 transceiver chipset which implements the CSMA/CA scheme.

[4] The probability of successful transmission is given by $s(\eta^k) = \sum_{n=0}^{B_{\max}} p(\eta^k)(1-p(\eta^k))^n$, where $B_{\max}$ denotes the maximum number of back-offs; see [12, p. 6099] for details in the unslotted CSMA/CA scheme.

$$U^k_{\rm loc}\big(a^k, \boldsymbol{a}^{\mathcal{N}_k}; \boldsymbol{\theta}\big) = \begin{cases} 0, & a^k = 0 \text{ (sleep)},\\ K_{l,1}\big(1 - e^{-\gamma_l \|\theta^k - \phi^k\|^2 s(\eta^k)}\big) - K_{l,2}\big(E_{\rm Tx}(\eta^k) + E_{\rm Act}\big), & a^k = 1 \text{ (activate)} \end{cases} \quad (36)$$

Step 3) **Estimate Exchange:** If $a_n^k = 1$: (i) Transmit θ_n^k to neighbors $\mathcal{N}_k$ and collect $\{\theta_n^l\}_{l\in\mathcal{N}_k}$; (ii) Form neighbors' action profile $\boldsymbol{a}_n^{\mathcal{N}_k} = (a_n^l)_{l\in\mathcal{N}_k}$ and $(\widehat{\theta}_n^l)_{l\in\mathcal{N}_k}$ such that

$$a_n^l = \begin{cases} 1, & \text{node } k \text{ receives node } l\text{'s estimate } \theta_n^l, \\ 0, & \text{otherwise,} \end{cases} \tag{41}$$

and

$$\widehat{\theta}_n^l = \begin{cases} \text{received estimate}, & \text{if } a_n^k = 1, \\ \theta_{n-1}^l, & \text{if } a_n^k = 0. \end{cases} \tag{42}$$

If $a_n^k = 0$, go to Step 5.

Step 4) **Data Fusion:**

$$\phi_n^k = \sum_{l\in\mathcal{N}_k^c} w_{kl}\widehat{\theta}_n^l. \tag{43}$$

Step 5) **Regret Update:** For all $i, j \in \mathcal{A}$:

Step 5.1: Local-Regret Update [see (44), at the bottom of the page].

Step 5.2: Global-Regret Update [see (45), shown at the bottom of the page]. In (45), $U_{\text{glob},n}^k\big(a_n^k\big)$ denotes the realized global payoff at time n.

Step 6) **Recursion:** Set $n \leftarrow n+1$ and go to Step 1.

Remark 4.2:

Although the dynamics of the K-tuple of estimates $\boldsymbol{\theta}$, underlying the activation control game, will be used in our tracking analysis, the implementation of the game-theoretic activation mechanism does not require any explicit knowledge of these dynamics.

Extensions:

1) A useful variant of the above algorithm can be developed by allowing nodes to fuse (or even decide wether to fuse) neighbors' data while in sleep mode. More precisely, in lieu of (40), each node runs local diffusion LMS updates of the form:

$$\theta_n^k = \theta_{n-1}^k + \varepsilon\left[\varphi_{n-1}^k + {h_n^k}'\left[y_n^k - h_n^k\phi_{n-1}^k\right]\cdot a_n^k\right]. \tag{46}$$

This helps to diffuse data when nodes are power hungry in measurements.

2) While the true parameter $\overline{\theta}$, connectivity graph $\mathcal{G}$ and weight matrix W are assumed fixed in Section II, the constant step-sizes ε and ρ in Algorithm 1 make it responsive to slowly time-varying $\overline{\theta}_n$, $\mathcal{G}_n$ and W_n. A typical method for analyzing the tracking performance of an adaptive algorithm is to postulate a *hypermodel* for the time variations [5]. For exposition, suppose only $\overline{\theta}_n$ evolves with time according to a discrete-time Markov chain with finite state space $\mathcal{M}_{\overline{\theta}} = \{\overline{\theta}(1), \ldots, \overline{\theta}(S)\}$ and transition matrix

$$P^\gamma = I_S + \gamma Q_{\overline{\theta}}, \tag{47}$$

where $0 < \gamma \ll 1$ and $Q_{\overline{\theta}} = (q_{ij})$ is the generator of a continuous-time Markov chain satisfying: $q_{ij} \geq 0$ for $i \neq j$, $Q_{\overline{\theta}}\mathbb{1}_S = \mathbf{0}$ and $|q_{ij}| \leq 1$, for all $i, j \in \mathcal{M}_{\overline{\theta}}$. The small parameter γ specifies how slowly $\overline{\theta}_n$ evolves with time. Then, if the Markov chain evolves on the same timescale as the adaptation rate of the diffusion LMS updates, i.e., $\gamma = \mathcal{O}(\varepsilon)$, each node can properly track $\overline{\theta}_n$ by employing Algorithm 1. (Note that the dynamics of the hypermodel $\overline{\theta}_n$ does not explicitly enter the implementation of the algorithm—it is only used in the tracking analysis.)

3) Nodes may face random communication delays in receiving estimates from neighbors. Let $d_{lk}(n)$ denote the random delay node k faces in receiving estimate from node l at time n. Then, node k has access to $\theta_{n-d_{lk}(n)}^l$ at time n, but not θ_τ^l for $\tau > n - d_{lk}(n)$, and if it does, it does not realize that they are more recent. The results of [6, Chapter 12] ensure that the convergence analysis of Algorithm 1 in Section VI is still valid if delays are bounded or not arbitrarily large in the sense that $\big(n - d_{lk}(n)\big)/n \to 1$ almost surely.

C. Discussion of the Game-Theoretic Activation Mechanism

Each node k generates two regret matrices and updates them over time: (i) $\boldsymbol{\alpha}_{2\times 2}^k$, that records *weighted average local-regrets*, and (ii) $\boldsymbol{\beta}_{2\times 2}^k$, which is an unbiased estimator of the *weighted average global-regrets*. Each element $\alpha_n^k(i,j)$, $i, j \in \mathcal{A}$, gives the weighted time-averaged regret (in terms of gains and losses in local utilities) had the node selected action j every time it took action i in the past. Formally,

$$\begin{aligned}\alpha_n^k(i,j) = \rho\sum_{\tau=1}^{n}(1-\rho)^{n-\tau} \\ \times\Big[U_{\text{loc}}^k\big(j, \boldsymbol{a}_\tau^{\mathcal{N}_k}; \boldsymbol{\theta}_\tau\big) - U_{\text{loc}}^k\big(a_\tau^k, \boldsymbol{a}_\tau^{\mathcal{N}_k}; \boldsymbol{\theta}_\tau\big)\Big] I_{\{a_\tau^k = i\}}.\end{aligned} \tag{48}$$

$$\psi_n^k(i) = \begin{cases} (1-\delta)\min\left\{\frac{1}{\xi^k}\left|\alpha_{n-1}^k\left(a_{n-1}^k, i\right) + \beta_{n-1}^k\left(a_{n-1}^k, i\right)\right|^+, \frac{1}{2}\right\} + \frac{\delta}{2}, & i \neq a_{n-1}^k \\ 1 - \sum_{\substack{j\in\mathcal{A} \\ j\neq i}}\psi_n^k(j), & i = a_{n-1}^k \end{cases} \tag{39}$$

$$\alpha_n^k(i,j) = \alpha_{n-1}^k(i,j) + \rho\left[\left(U_{\text{loc}}^k\big(j, \boldsymbol{a}_n^{\mathcal{N}_k}; \boldsymbol{\theta}_n\big) - U_{\text{loc}}^k\big(a_n^k, \boldsymbol{a}_n^{\mathcal{N}_k}; \boldsymbol{\theta}_n\big)\right) I_{\{a_n^k = i\}} - \alpha_{n-1}^k(i,j)\right] \tag{44}$$

$$\beta_n^k(i,j) = \beta_{n-1}^k(i,j) + \rho\left[\left[\frac{\psi_n^k(i)}{\psi_n^k(j)} U_{\text{glob},n}^k\big(a_n^k\big) I_{\{a_n^k = j\}} - U_{\text{glob},n}^k\big(a_n^k\big) I_{\{a_n^k = i\}}\right] - \beta_{n-1}^k(i,j)\right] \tag{45}$$

However, since nodes do not observe the action profile of non-neighbors $\boldsymbol{a}^{\mathcal{S}_k}$, they are unable to perform the thought experiment to evaluate global utilities for alternative actions j as they do in (48). We thus formulate an *unbiased* estimator $\boldsymbol{\beta}_n^k$ of the *weighted time-averaged global-regrets* based only on the *realized* global utilities $\{U^k_{\text{glob},\tau}(a^k_\tau)\}$[9]. Each element $\beta^k_n(i,j)$, $i,j \in \mathcal{A}$, is formulated as:

$$\beta^k_n(i,j) = \rho \sum_{\tau=1}^{n} (1-\rho)^{n-\tau} \times \left[\frac{\psi^k_\tau(i)}{\psi^k_\tau(j)} U^k_{\text{glob},\tau}\left(a^k_\tau\right) I_{\{a^k_\tau = j\}} - U^k_{\text{glob},\tau}\left(a^k_\tau\right) I_{\{a^k_\tau = i\}} \right]. \quad (49)$$

Nodes combine $\boldsymbol{\alpha}^k_n$ and $\boldsymbol{\beta}^k_n$ to update their activation strategy for the following period. Positive overall-regrets $\alpha^k_n(i,j)+\beta^k_n(i,j)$ imply the opportunity to achieve higher utilities by switching from action i to j in future. The more positive the regret for not choosing an action, the higher is the probability that the node picks that action.

1) Computational Complexity: The computational burden of the game-theoretic activation mechanism is negligible as compared to the diffusion LMS ($O(M)$ calculations per iteration)—it does not grow with the number of nodes, nor with the dimension of the parameter estimation problem. At each iteration, each node needs to update a row in $\boldsymbol{\alpha}^k_n$, and a row and column in $\boldsymbol{\beta}^k_n$. Two multiplications, two additions, two comparisons and a table lookup (assuming random numbers are stored in a table) then suffice to select the next action (see (39)). Therefore, it is suitable for implementation in nodes with limited computational capability.

2) Exploration Vs. Exploitation: At successive periods, with probability $1-\delta$, node k chooses its consecutive action according to $|\boldsymbol{\alpha}^k_n + \boldsymbol{\beta}^k_n|^+$. With the remaining probability δ, it randomizes among all actions $\mathcal{A}$ according to a uniform distribution. The tunable "exploration" factor δ forces all actions to be chosen with a minimum frequency and is essential as nodes continuously learn their global utility functions. As will be discussed later, larger δ will lead to convergence of the nodes' global activation behavior to a larger ϵ-distance of the correlated equilibria set. Taking the minimum with $1/|\mathcal{A}|$ in (39) also ensures $\sum_{a^k \in \mathcal{A}} \psi^k_n(a^k) = 1$.

3) Inertia: The normalization factor ξ^k in (39) can be interpreted as an *inertia* parameter for switching actions [8]—higher ξ^k lowers the probability of switching actions. To avoid computing ξ^k repeatedly, one can fix $\xi^k > 2|U^k_{\max} - U^k_{\min}|$, where $U^k_{\max}$ and $U^k_{\min}$ denote the upper and lower bounds on node k's utility function, respectively. This inertia is essential in the convergence of the game-theoretic adaptive filter (44)–(45) in Algorithm 1. The rate of convergence of the adaptive filtering algorithm is closely related to ξ^k—higher ξ^k lowers the convergence rate.

4) Better-Reply Adaptive Filter: The strategy in Step 1 of Algorithm 1 reinforces all plausible actions with positive probabilities. Hence, the behavior of nodes is more of a reflex-oriented individual (better-reply strategy) than a sophisticated decision-maker who takes the most plausible action (best-reply) given its limited conception of the activation strategy of others [8].

V. Global Network Performance

This section deals with global performance analysis of the energy-aware diffusion LMS proposed in Section IV-B. We start by defining two global performance quantities in Section V-A that capture the behavior of individual nodes across the network. We then state the main theorem of this paper that reveals the global performance of Algorithm 1 in Section V-B. The detailed proofs are postponed until Section VI.

A. Global Behavior

The emergent global behavior of the network can be captured by two inter-connected discrete-time processes:

1) Network-Wide Diffusion LMS Recursion: Define:

$$M_n = \left[\text{diag}\{a^1_n I_M, \ldots, a^K_n I_M\}\right] \quad (KM \times KM).$$

Following the same lines as (23), the global diffusion LMS update associated with Algorithm 1 can be expressed in the state-space form as

$$\boldsymbol{\theta}_n = \boldsymbol{\theta}_{n-1} + S(\varepsilon) M_n \left[\widehat{B}\boldsymbol{\theta}_{n-1} + H_n{}'\left(Y_n - H_n B \boldsymbol{\theta}_{n-1}\right)\right]. \quad (50)$$

In light of (50), Algorithm 1 can be viewed as a synchronous distributed stochastic approximation algorithm [6], [21]: At each time n, some (but not necessarily all) elements of $\boldsymbol{\theta}_n$ are updated, each by different nodes spread across the network. The common clock that synchronizes the system is implicit in the game-theoretic activation mechanism employed by nodes. An asynchronous implementation of Algorithm 1 is feasible, however, is out of the scope of this paper.

2) Global Activation Behavior: The collective activation behavior of nodes, $\boldsymbol{z}_n$, is defined as the discounted *empirical frequency* of joint activation decisions of all nodes up to period n. Formally,

$$\boldsymbol{z}_n = \rho \sum_{\tau \le n} (1-\rho)^{n-\tau} \boldsymbol{e}_{\boldsymbol{a}_\tau}, \quad (51)$$

where $\boldsymbol{e}_{\boldsymbol{a}_\tau}$ denotes the 2^K-dimensional unit vector with the element corresponding to $\boldsymbol{a}_\tau$ being equal to one. In (51), ρ serves as a forgetting factor to foster adaptivity to the evolution of the local parameter estimates. That is, the effect of the old local parameter estimates on the activation strategy of nodes vanishes as the nodes repeatedly take actions. It is more convenient to rewrite $\boldsymbol{z}_n$ as a stochastic approximation equation:

$$\boldsymbol{z}_n = \boldsymbol{z}_{n-1} + \rho\left(\boldsymbol{e}_{\boldsymbol{a}_n} - \boldsymbol{z}_{n-1}\right). \quad (52)$$

The global performance analysis of the network, captured by (50) and (52), is nontrivial and quite challenging as the two global recursions are closely coupled: The estimates $\boldsymbol{\theta}_n$ affect utility of nodes, hence, their choice of action ($\boldsymbol{e}_{\boldsymbol{a}_n}$ in (52)). On the other hand, nodes' strategies $\boldsymbol{\psi}^k$ are functions of $\boldsymbol{z}_n$ (see (75) in Section VI), which in turn enter M_n in (50).

B. Main Result: From Individual to Global Behavior

Let us first define the set of *correlated ϵ-equilibria* $\mathcal{C}_\epsilon(\boldsymbol{\theta})$, which this paper focuses on as the solution concept for the activation control game:

Definition 5.1: Let $\boldsymbol{\pi}^c_{\boldsymbol{\theta}}$ denote a joint distribution on the joint action space $\mathcal{A}^{\mathcal{K}} = \times_{k \in \mathcal{K}} \mathcal{A}$, where $\pi^c_{\boldsymbol{\theta}}(\boldsymbol{a}) \ge 0$ for all $\boldsymbol{a} \in$

$\mathcal{A}^{\mathcal{K}}$ and $\sum_{\boldsymbol{a}\in\mathcal{A}^{\mathcal{K}}}\pi^{c}_{\boldsymbol{\theta}}(\boldsymbol{a})=1$. The set of *correlated ϵ-equilibria*, denoted by $\mathcal{C}_{\epsilon}(\boldsymbol{\theta})$, is the convex polytope [see (53), shown at the bottom of the page] [5].

In (53), $\pi^{c,k}_{\boldsymbol{\theta}}\left(i,\boldsymbol{a}^{-k}\right)$ denotes the probability of node k selecting action i and the rest $\boldsymbol{a}^{-k}$. The above definition simply states, when the "recommendation" $\boldsymbol{a}\sim\boldsymbol{\pi}^{c}_{\boldsymbol{\theta}}$ puts positive probability on choosing action i, node k can gain at most ϵ by deviating through choosing $j\in\mathcal{A}\backslash\{i\}$. Note that $\mathcal{C}_{\epsilon}(\boldsymbol{\theta})$ is a function of hypermodel $\boldsymbol{\theta}$—it slowly evolves as the local parameter estimates $\boldsymbol{\theta}_n$ change over time.

We now characterize the global behavior emerging by each node individually following Algorithm 1. Let $\widetilde{\boldsymbol{\theta}}=\boldsymbol{\theta}-\overline{\boldsymbol{\theta}}$. Define the piecewise constant continuous-time interpolation processes $\widetilde{\boldsymbol{\theta}}^{\varepsilon}(\cdot)$ and $\boldsymbol{z}^{\rho}(\cdot)$ associated with Algorithm 1:

$$\begin{aligned}\widetilde{\boldsymbol{\theta}}^{\varepsilon}(\cdot)&=\widetilde{\boldsymbol{\theta}}_n,\quad \text{for } t\in[n\varepsilon,(n+1)\,\varepsilon),\\ \boldsymbol{z}^{\rho}(\cdot)&=\boldsymbol{z}_n,\quad \text{for } t\in[n\rho,(n+1)\,\rho).\end{aligned}$$

The following theorem is the main result of the paper and asserts that, by following Algorithm 1: (i) the local estimates θ^k_n converge to the true parameter $\overline{\theta}$ across the network, (ii) The global activation behavior tracks the time-varying polytope of correlated ϵ-equilibria $\mathcal{C}_{\epsilon}(\boldsymbol{\theta}_n)$ of the underlying activation control game as $\boldsymbol{\theta}_n$ evolves with time.

Theorem 5.1: Consider the setting of Section IV-A and suppose every node follows Algorithm 1. Then, for each $\epsilon>0$, there exists $\widehat{\delta}(\epsilon)$ such that if $\delta<\widehat{\delta}(\epsilon)$ in Algorithm 1 (Step 1) and $\varepsilon=\mathcal{O}(\rho^2)$,

1) As $\rho\to 0$, $\boldsymbol{z}^{\rho}(\cdot)$ converges weakly to $\mathcal{C}_{\epsilon}(\boldsymbol{\theta}(\cdot))$ in the sense that:
$$d\big(\boldsymbol{z}^{\rho}(\cdot),\mathcal{C}_{\epsilon}(\boldsymbol{\theta}(\cdot))\big)=\inf_{\boldsymbol{z}\in\mathcal{C}_{\epsilon}(\boldsymbol{\theta}(\cdot))}|\boldsymbol{z}^{\rho}(\cdot)-\boldsymbol{z}|\Rightarrow 0. \tag{54}$$
2) Under (C2), $\widetilde{\boldsymbol{\theta}}^{\varepsilon}(\cdot)$ converges weakly, as $\varepsilon\to 0$, to $\widetilde{\boldsymbol{\theta}}(\cdot)$ that is a solution to
$$\frac{d\widetilde{\boldsymbol{\theta}}(t)}{dt}\in\Big\{M_{\boldsymbol{\pi}^{c}_{\boldsymbol{\theta}(t)}}\big(\widehat{B}-R^{d}_{h}\big)\widetilde{\boldsymbol{\theta}}(t);\boldsymbol{\pi}^{c}_{\boldsymbol{\theta}(t)}\in\mathcal{C}_{\epsilon}(\boldsymbol{\theta}(t))\Big\}, \tag{55}$$
where $M_{\boldsymbol{\pi}^{c}_{\boldsymbol{\theta}}}=\operatorname{diag}\{m^{1}_{\boldsymbol{\pi}^{c}_{\theta}}I_M,\ldots,m^{K}_{\boldsymbol{\pi}^{c}_{\theta}}I_M\}$ and $I_{KM}>M_{\boldsymbol{\pi}^{c}_{\boldsymbol{\theta}}}>\kappa\cdot I_{KM}$ for some $\kappa>0$.
3) System (55) is globally asymptotically stable and
$$\lim_{t\to\infty}\big\|\widetilde{\boldsymbol{\theta}}(t)\big\|=\mathbf{0}.$$

Proof: The proof is provided in detail in Section VI. ■

Remark 5.1: The exploration factor δ essentially determines how close the regrets of each node can get to zero. Larger δ enforces players to choose "non-promising" actions more frequently. This in turn leads to the global activation behavior of nodes converging to a larger ϵ-distance of the correlated equilibria set.

From the game-theoretic point of view, part 2) in Theorem 5.1 simply states that non-fully rational local behavior of individual nodes (due to utilizing a better-reply rather than a best-reply strategy) leads to sophisticated globally rational behavior, where all nodes pick actions from a common joint distribution, i.e., coordination in decision making among nodes.

VI. Proof of Theorem 5.1: Convergence Analysis of the Energy-Aware Diffusion LMS

This section is devoted to the proof of the global performance analysis of Algorithm 1, summarized in Theorem 5.1. The proof relies on the results of Section II-D which proves asymptotic consistency of the small step-size diffusion LMS. For brevity and to make it more comprehensible, we omit technical details and present only an intuitive sketch of the proof below:

Step 1: Let $\mathcal{I}_n=\{k|a^k_n=1\}$ denote the set of nodes that choose to activate at period n. One can unfold iteration n of (50) by replacing it with $|\mathcal{I}_n|$ distinct iterations, where each involves only one node $k\in\mathcal{I}_n$ updating its estimate. Relabeling iteration indices by m results in a new algorithm with the same behavior as Algorithm 1. Since each node explores with probability δ at each period, the stationary distribution places a probability of at least δ on "activation" and
$$\liminf_{m\longrightarrow\infty}\frac{1}{m}\sum_{\tau=0}^{m}a^k_{\tau}\geq\delta,\quad\forall k\in\mathcal{K}. \tag{56}$$
Equation (56) is essential in the proof of Theorem 5.1 and simply asserts that all nodes take measurements and update their estimates relatively often (however, in a competitively optimal sense described by the set of correlated ϵ-equilibria).

For convenience, take $\varepsilon=\rho^2$. Note that an interval of length δ in the ρ timescale involves δ/ρ iterates, whereas, the same number of iterations take $\delta\varepsilon/\rho$ in the ε timescale. To analyze the asymptotics of the joint process $(\boldsymbol{z}^{\rho}(\cdot),\widetilde{\boldsymbol{\theta}}^{\rho}(\cdot))$, it is instructive to first look at the singularly perturbed coupled system:
$$\begin{cases}\frac{d\widetilde{\boldsymbol{\theta}}(t)}{dt}=\rho M_{\boldsymbol{z}(t)}\big[\widehat{B}-R^{d}_{h}\big]\widetilde{\boldsymbol{\theta}}(t),\\ \frac{d\boldsymbol{z}(t)}{dt}\in\big\{\boldsymbol{\sigma}^{k}(\boldsymbol{z}(t),\boldsymbol{\theta}(t))\times\nu^{-k};\nu^{-k}\in\Delta\mathcal{A}^{-k}\big\}-\boldsymbol{z}(t),\end{cases} \tag{57}$$
in the limit $\rho\rightarrow 0$. In (57), $\widetilde{\boldsymbol{\theta}}(\cdot)=\boldsymbol{\theta}(\cdot)-\overline{\boldsymbol{\theta}}$ denotes the tracking error, and
$$M_{\boldsymbol{z}(t)}\triangleq\operatorname{diag}\{m^{1}_{\boldsymbol{z}(t)}I_M,\ldots,m^{K}_{\boldsymbol{z}(t)}I_M\}\ \ (KM\times KM),$$
where
$$I_{KM}>M_{\boldsymbol{z}(t)}>\kappa\cdot I_{KM},\qquad\kappa>0. \tag{58}$$

[5]Nash equilibrium corresponds to the special case where agents act independently. That is, $\boldsymbol{\pi}$ is a product measure: $\pi(a^1,\ldots,a^K)=\Pi_{k=1}^{K}\pi^k(a^k)$. Every Nash equilibrium is thus a correlated equilibrium. The set of correlated equilibria is nonempty, closed and contains the convex hull of Nash equilibria.

$$\mathcal{C}_{\epsilon}\big(\boldsymbol{\theta}\big)=\left\{\boldsymbol{\pi}^{c}_{\boldsymbol{\theta}}:\sum_{\boldsymbol{a}^{-k}\in\mathcal{A}^{-k}}\pi^{c,k}_{\boldsymbol{\theta}}\big(i,\boldsymbol{a}^{-k}\big)\left[U^{k}\big(j,\boldsymbol{a}^{-k};\boldsymbol{\theta}\big)-U^{k}\big(i,\boldsymbol{a}^{-k};\boldsymbol{\theta}\big)\right]\leq\epsilon,\ \forall i,j\in\mathcal{A},\forall k\in\mathcal{K}\right\} \tag{53}$$

In addition, $\nu^{-k} \in \Delta\mathcal{A}^{-k}$, where $\Delta\mathcal{A}^{-k}$ denotes the set of all probability measures over the joint activation decisions of nodes excluding node k, $\boldsymbol{\sigma}^k(\cdot)$ represents the randomized activation strategy of node k (which will be defined later in the proof). Each element $m^k_{\boldsymbol{z}(t)}$ in $M_{\boldsymbol{z}(t)}$ is the relative instantaneous rate at which each node k is being activated. More precisely, $m^k_{\boldsymbol{z}(t)} = \sum_{\boldsymbol{a}^{-k}} z^k_t(1, \boldsymbol{a}^{-k})$, where $z^k_t(1, \boldsymbol{a}^{-k})$ denotes the empirical frequency of node k choosing to activate and the rest $\boldsymbol{a}^{-k}$ at time t. Thus, (58) is a consequence of (56).

In view of (57), $\widetilde{\boldsymbol{\theta}}(\cdot)$ is the slow component and $\boldsymbol{z}(\cdot)$ is the fast transient. Therefore, $\widetilde{\boldsymbol{\theta}}(\cdot)$ is quasi-static, i.e., remains almost constant, while analyzing the behavior of $\boldsymbol{z}(\cdot)$. The weak convergence argument shows that $(\boldsymbol{z}^\rho(\cdot), \widetilde{\boldsymbol{\theta}}^\rho(\cdot))$ converge weakly to $(\boldsymbol{z}(\cdot), \widetilde{\boldsymbol{\theta}}(\cdot))$ as $\rho \longrightarrow 0$, such that the limit is a solution to

$$\begin{cases} \frac{d\widetilde{\boldsymbol{\theta}}(t)}{dt} = \mathbf{0}, \\ \frac{d\boldsymbol{z}(t)}{dt} \in \left\{\boldsymbol{\sigma}(\boldsymbol{z}(t), \boldsymbol{\theta}(t)) \times \nu^{-k}; \nu^{-k} \in \Delta\mathcal{A}^{-k}\right\} - \boldsymbol{z}(t). \end{cases} \tag{59}$$

Technical details are tedious and omitted for brevity; see [6, Chapter 8]. The second step in the convergence analysis is then to prove asymptotic stability and characterize the set of global attractors of:

$$\frac{d\boldsymbol{z}(t)}{dt} \in \left\{\boldsymbol{\sigma}(\boldsymbol{z}(t), \boldsymbol{\theta}) \times \nu^{-k}; \nu^{-k} \in \Delta\mathcal{A}^{-k}\right\} - \boldsymbol{z}(t). \tag{60}$$

Step 2: There exists a close connection between $\boldsymbol{z}_n$ and the pair process $(\boldsymbol{\alpha}^k_n, \boldsymbol{\beta}^k_n)$ for each node k following Algorithm 1: $(\boldsymbol{\alpha}^k_n, \boldsymbol{\beta}^k_n)$ record the global activation behavior $\boldsymbol{z}_n$ of the network translated based on each node's utility function into regrets. In what follows, since it is more convenient to work with $(\boldsymbol{\alpha}^k_n, \boldsymbol{\beta}^k_n)$, we study the limiting process for $(\boldsymbol{\alpha}^k_n, \boldsymbol{\beta}^k_n)$ and analyze its asymptotic stability. We then formally translate the results to prove asymptotic stability of (60).

Before proceeding, a few definitions are in order. Define:

$$\boldsymbol{\sigma}^k(\boldsymbol{z}, \boldsymbol{\theta}) = (1-\delta)\,\boldsymbol{\chi}(\boldsymbol{\alpha}^k, \boldsymbol{\beta}^k) + \frac{\delta}{2} \cdot \begin{bmatrix} 1 \\ 1 \end{bmatrix} \tag{61}$$

where $\boldsymbol{\chi}^k(\boldsymbol{\theta}) = \boldsymbol{\chi}(\boldsymbol{\alpha}^k, \boldsymbol{\beta}^k)$ is an invariant measure of $\boldsymbol{\psi}^k$ when $\delta = 0$ (see (39)), satisfying

$$\sum_{j \in \mathcal{A}\setminus\{i\}} \chi^k_j \cdot \left|\alpha^k(j,i) + \beta^k(j,i)\right|^+ = \chi^k_i \cdot \sum_{j \in \mathcal{A}\setminus\{i\}} \left|\alpha^k(i,j) + \beta^k(i,j)\right|^+. \tag{62}$$

Let $\nu^k \in \Delta\mathcal{A}^{\mathcal{N}_k}$ denote a probability measure over the joint activation decisions of node k's neighbors, namely, $\boldsymbol{a}^{\mathcal{N}_k}$, and define the expected payoff:

$$U^k_{\text{loc}}(a^k, \nu^k; \theta) = \sum_{\boldsymbol{a}^{\mathcal{N}_k} \in \mathcal{A}^{\mathcal{N}_k}} U^k_{\text{loc}}(a^k, \boldsymbol{a}^{\mathcal{N}_k}; \theta)\nu^k(\boldsymbol{a}^{\mathcal{N}_k}). \tag{63}$$

Here, with a slight abuse of notation, we denote by $U^k(\cdot;\boldsymbol{\theta})$ the multi-linear extension of the local utility function to the set of mixed (probabilistic) strategies. Associated with the game-theoretic activation mechanism, define further the interpolated pair process:

$$\left(\boldsymbol{\alpha}^{k,\rho}(\cdot), \boldsymbol{\beta}^{k,\rho}(\cdot)\right) = \left(\boldsymbol{\alpha}^k_n, \boldsymbol{\beta}^k_n\right) \text{ for } \quad t \in [n\rho, (n+1)\,\rho)\,.$$

The following theorem shows that the limit system associated with the stochastic approximation iterates $(\boldsymbol{\alpha}^k_n, \boldsymbol{\beta}^k_n)$ is a system of coupled differential inclusions that is Lyapunov stable and characterizes its global attractors set.

Theorem 6.1: For a fixed $\boldsymbol{\theta}$,

1) As $\rho \to 0$, the interpolated pair process $\left(\boldsymbol{\alpha}^{k,\rho}(\cdot), \boldsymbol{\beta}^{k,\rho}(\cdot)\right)$ converges weakly to $\left(\boldsymbol{\alpha}^k(\cdot), \boldsymbol{\beta}^k(\cdot)\right)$ that is a solution of the differential inclusion:
$$\begin{cases} \frac{d\boldsymbol{\alpha}^k(t)}{dt} \in L^k\left(\boldsymbol{\alpha}^k(t), \boldsymbol{\beta}^k(t), \boldsymbol{\theta}\right) - \boldsymbol{\alpha}^k(t), \\ \frac{d\boldsymbol{\beta}^k}{dt} = \boldsymbol{G}^k\left(\boldsymbol{\alpha}^k(t), \boldsymbol{\beta}^k(t)\right) - \boldsymbol{\beta}^k(t), \end{cases} \tag{64}$$
where elements of $L^k(\cdot)$ and $\mathbf{G}^k(\cdot)$ are given by: (65) and (66) at the bottom of the page. In (66), $U^k_{\text{glob},t}(\cdot)$ is the interpolated process of the global payoffs accrued from the game.
2) Rearrange $\boldsymbol{\alpha}^k(\cdot)$ and $\boldsymbol{\beta}^k(\cdot)$ as 4×1 vectors. For each $\epsilon > 0$, there exists $\widehat{\delta}(\epsilon) > 0$ such that if $0 < \delta < \widehat{\delta}(\epsilon)$, the limit system (64) is globally asymptotically stable, and
$$\lim_{t \to \infty} d\left(\left(\boldsymbol{\alpha}^k(t), \boldsymbol{\beta}^k(t)\right), \mathcal{R}\right) = 0, \tag{67}$$
where
$$\mathcal{R} = \{(\boldsymbol{\alpha}^k, \boldsymbol{\beta}^k) \in \mathbb{R}^4 \times \mathbb{R}^4; |\boldsymbol{\alpha}^k + \boldsymbol{\beta}^k|^+ \le \epsilon \cdot \mathbb{1}_r\}, \tag{68}$$
and $d(\cdot,\cdot)$ denotes the usual distance function.
3) As $\rho \to 0$ followed by $t \to \infty$, the limit points of $\left(\boldsymbol{\alpha}^{k,\rho}(\cdot), \boldsymbol{\beta}^{k,\rho}(\cdot)\right)$ and $\left(\boldsymbol{\alpha}^k(\cdot), \boldsymbol{\beta}^k(\cdot)\right)$ coincide, i.e.,
$$\left(\boldsymbol{\alpha}^{k,\rho}(t), \boldsymbol{\beta}^{k,\rho}(t)\right) \Rightarrow \mathcal{R}, \quad \text{as}\rho \to 0, \text{t} \to \infty \tag{69}$$
4) $\boldsymbol{z}^\rho(t) \to \mathcal{C}_\epsilon(\boldsymbol{\theta})$ if and only if
$$\left(\boldsymbol{\alpha}^{k,\mu}(t), \boldsymbol{\beta}^{k,\mu}(t)\right) \to \mathcal{R}, \qquad \text{forall} \quad k \in \mathcal{K}. \tag{70}$$

Proof: The proof for part 1) uses martingale averaging techniques and is based on [6, Chapter 8]; see Appendix A in [15] for detailed proof.

The proof for part 2) is based on [22]. Define the Lyapunov function:

$$V\left(\boldsymbol{\alpha}^k, \boldsymbol{\beta}^k\right) = \left\|\boldsymbol{\alpha}^k + \boldsymbol{\beta}^k\right\|^2_F \tag{71}$$

where $\|\cdot\|_F$ denotes the Frobenius norm. Taking the time-derivative of (71), we obtain (with some details omitted):

$$\frac{d}{dt}V\left(\boldsymbol{\alpha}^k, \boldsymbol{\beta}^k\right) \le A(\boldsymbol{\theta})\delta \sum_{i,j} \left|\alpha^k(i,j) + \beta^k(i,j)\right|^+ - 2V\left(\boldsymbol{\alpha}^k, \boldsymbol{\beta}^k\right), \tag{72}$$

$$\mathcal{L}^k_{ij}\left(\boldsymbol{\alpha}^k, \boldsymbol{\beta}^k, \boldsymbol{\theta}\right) = \left\{ \left[U^k_l(j, \nu^k; \boldsymbol{\theta}) - U^k_l(i, \nu^k; \boldsymbol{\theta})\right] \sigma^k_i; \nu^k \in \Delta\mathcal{A}^{\mathcal{N}_k} \right\} \tag{65}$$

$$G^k_{ij}\left(\boldsymbol{\alpha}^k, \boldsymbol{\beta}^k\right) = \left[U^k_{\text{glob},t}(j) - U^k_{\text{glob},t}(i)\right] \sigma^k_i \tag{66}$$

where the constant $A(\boldsymbol{\theta})$ depends on the range of utility function $U^k(\cdot;\boldsymbol{\theta})$; see Appendix B in [15] for omitted details. Then, $|\boldsymbol{\alpha}^k+\boldsymbol{\beta}^k|^+ > \epsilon\cdot\mathbb{1}_{A^2}$ implies δ can be chosen small enough such that:

$$\frac{d}{dt}V\big(\boldsymbol{\alpha}^k,\boldsymbol{\beta}^k\big) \le -V\big(\boldsymbol{\alpha}^k,\boldsymbol{\beta}^k\big). \tag{73}$$

This implies that (64) is globally asymptotically stable for each $\boldsymbol{\theta}$, and

$$\lim_{t\to\infty} d\big(\big(\boldsymbol{\alpha}^k(t),\boldsymbol{\beta}^k(t)\big),\mathcal{R}\big)=0. \tag{74}$$

Part 3) is similar to part 3) in Theorem 2.1. In part 1), we consider ρ small and n large, however ρn remains bounded, whereas in part 3), we simply look at the asymptotics of the limiting process (64) as $t\longrightarrow\infty$; see the proof of Theorem 2.1.

The proof for part 4) relies on the definition of local- and global-regrets. In light of (48)–(49),

$$\lim_{t\to\infty}\alpha^{k,\rho}(i,j)(t)+\beta^{k,\rho}(i,j)(t)=\lim_{t\to\infty} \sum_{\boldsymbol{a}^{-k}} z^{k,\rho}\big(i,\boldsymbol{a}^{-k}\big)(t)\left[U^k\big(j,\boldsymbol{a}^{-k};\boldsymbol{\theta}\big)-U^k\big(i,\boldsymbol{a}^{-k};\boldsymbol{\theta}\big)\right]. \tag{75}$$

(In (75), we use the fact that $\boldsymbol{\beta}_n^k$ is an unbiased estimator of the true global-regret matrix as if node k observed the activation decisions of non-neighbors; see [15, Theorem4.2] for details.) On any convergent subsequence $\{\boldsymbol{z}_{n'}\}_{n'\ge 0}\longrightarrow\boldsymbol{\pi}_{\boldsymbol{\theta}}$, with slight abuse of notation, let $\boldsymbol{z}^\rho(t)=\boldsymbol{z}_{n'}$ and $(\boldsymbol{\alpha}^{k,\rho}(t),\boldsymbol{\beta}^{k,\rho}(t))=(\boldsymbol{\alpha}_{n'}^k,\boldsymbol{\beta}_{n'}^k)$ for $t\in[n'\rho,(n'+1)\rho)$. Then,

$$\lim_{t\to\infty}\alpha^{k,\rho}(i,j)(t)+\beta^{k,\rho}(i,j)(t) =\sum_{\boldsymbol{a}^{-k}}\pi_{\boldsymbol{\theta}}^k\big(i,\boldsymbol{a}^{-k}\big)\left[U^k\big(j,\boldsymbol{a}^{-k};\theta\big)-U^k\big(i,\boldsymbol{a}^{-k};\theta\big)\right]. \tag{76}$$

Finally, comparing (76) with Definition 5.1, we conclude that $\boldsymbol{z}^\rho(t)$ converges to the ϵ-CE set $\mathcal{C}_\epsilon(\boldsymbol{\theta})$ if and only if, for all $k\in\mathcal{K}$,

$$\lim_{t\to\infty}\big|\boldsymbol{\alpha}^{k,\rho}(t)+\boldsymbol{\beta}^{k,\rho}(t)\big|^+\le\epsilon\cdot\mathbb{1}. \tag{77}$$

This completes the proof. ■

The above theorem asserts that the fast dynamics (60) is globally asymptotically stable for each fixed $\boldsymbol{\theta}$ and the set of correlated ϵ-equilibria $\mathcal{C}_\epsilon(\boldsymbol{\theta})$ constitutes the global attractors set. Then, for sufficiently small values of ρ, we expect $\boldsymbol{z}(t)$ to closely track $\mathcal{C}_\epsilon(\boldsymbol{\theta}(t))$ for $t>0$.

Step 3: This in turn suggests looking at the non-autonomous differential inclusion:

$$\frac{d\widetilde{\boldsymbol{\theta}}(t)}{dt}\in\Big\{M_{\boldsymbol{\pi}^c_{\boldsymbol{\theta}(t)}}\big(\widehat{B}-R_h^d\big)\widetilde{\boldsymbol{\theta}}(t);\boldsymbol{\pi}^c_{\boldsymbol{\theta}(t)}\in\mathcal{C}_\epsilon(\boldsymbol{\theta}(t))\Big\}, \tag{78}$$

which captures the dynamics of the slow parameter $\widetilde{\boldsymbol{\theta}}(t)$ when the fast parameter $\boldsymbol{z}(t)$ is equilibrated at $\mathcal{C}_\epsilon(\boldsymbol{\theta}(t))$. The following lemma proves asymptotic stability of (78) and characterizes its global attractor.

Lemma 6.1: The non-autonomous differential inclusion (78) is globally asymptotically stable and

$$\lim_{t\longrightarrow\infty}\big\|\widetilde{\boldsymbol{\theta}}(t)\big\|=\mathbf{0}. \tag{79}$$

Proof: The proof follows from that of Theorem 2.1. Since $M_{\boldsymbol{\pi}^c_{\boldsymbol{\theta}}}$ is diagonal with all positive elements, $M_{\boldsymbol{\pi}^c_{\boldsymbol{\theta}}}$ is positive-definite. Since $M_{\boldsymbol{\pi}^c_{\boldsymbol{\theta}}}$ is symmetric and positive definite and $\widehat{B}-R_h^d$ is symmetric and negative definite (following the same argument as in proof of Theorem 2.1), $M_{\boldsymbol{\pi}^c_{\boldsymbol{\theta}}}(\widehat{B}-R_h^d)<\mathbf{0}$ [23]. Therefore, (78) is globally asymptotically stable. Further, $\det\big(M_{\boldsymbol{\pi}^c_{\widetilde{\theta}}}(\widehat{B}-R_h^d)\big)<0$, hence, $M_{\boldsymbol{\pi}^c_{\widetilde{\theta}}}(\widehat{B}-R_h^d)\widetilde{\boldsymbol{\theta}}=0$ is non-singular and $\mathbf{0}$ is the unique global attractor of (78) for any $\boldsymbol{\pi}^c_{\boldsymbol{\theta}}\in\mathcal{C}_\epsilon(\boldsymbol{\theta}(t))$. ■

We then expect $(\boldsymbol{\theta}(t),\boldsymbol{z}(t))$ in (57) to approximately converge to (i.e., converge to a small neighborhood of) the point $(\overline{\boldsymbol{\theta}},\mathcal{C}_\epsilon(\overline{\boldsymbol{\theta}}))$. This intuition indeed carries over to the energy-aware diffusion LMS in Algorithm 1: the activation mechanism views the diffusion LMS as quasi-static while the diffusion LMS views the game-theoretic activation mechanism as almost equilibrated.

VII. Numerical Study

This section provides a numerical example to demonstrate the performance of the proposed energy-aware diffusion LMS in a small network of nodes. In Section VII-A. we answer the question "why is it impossible to obtain an analytical rate of convergence for the proposed two-timescale stochastic approximation algorithm?". We then elaborate on our semi-analytic approach to illustrate global network performance in Section VII-B. Finally, the simulation setup and the numerical results are presented in Section VII-C.

A. Why is Derivation of Analytical Convergence Rate Impossible?

It is well known that the asymptotic covariance of the limit process provides information about the rate of convergence of stochastic recursive algorithms; see [6, Chapter 10]. (The rate of convergence refers to the asymptotic properties of the normalized errors about the limit point $\overline{\theta}$.) For instance, in the standard LMS with constant step-size μ, the error $\theta_n-\overline{\theta}$ is asymptotically normally distributed with mean 0 and covariance $\mu\Sigma$, where

$$\Sigma=\mathbb{E}(\zeta_0{\zeta_0}')+\sum_{\tau=1}^{\infty}\mathbb{E}\zeta_0{\zeta_\tau}'+\sum_{\tau=1}^{\infty}\mathbb{E}\zeta_\tau{\zeta_0}'. \tag{80}$$

In (80), $\zeta_\tau={h_\tau}'(y_\tau-h_n\overline{\theta})$. If the noise is i.i.d., the covariance can be computed as:

$$\Sigma=\mathbb{E}[{h_n}'h_n(y_n-h_n\overline{\theta})^2]=R_h\sigma_v^2, \tag{81}$$

where $R_h=\mathbb{E}{h_n}'h_n$ and $\sigma_v^2=\mathbb{E}v_n^2$. The figures of merit such as the asymptotic *excess mean square error* (EMSE) can then be obtained as:

$$\mathrm{EMSE}=\frac{1}{2}\mathrm{Tr}(\Sigma), \tag{82}$$

where $\mathrm{Tr}(\cdot)$ denotes the trace operator.

This approach, however, does not apply to our analysis of the energy-aware diffusion LMS since the limiting process for the diffusion LMS is a differential inclusion rather than an ordinary

differential equation—see (78). That is, for each initial condition, the sample path of errors belongs to a set and independent runs of the algorithm leads to different sample paths. This prohibits deriving an analytical rate of convergence (accordingly analytical figure of merits) for Algorithm 1. It is, however, obvious that the performance of Algorithm 1 is upper bounded by the diffusion LMS without activation mechanism; see [1] for related results. In what follows, we resort to Monte Carlo simulations in order to illustrate and compare the rate of convergence and performance of Algorithm 1.

B. Semi-Analytic Approach to Numerical Study

To demonstrate the performance of Algorithm 1, we conduct a semi-analytic numerical study. In light of the proof of Theorem 5.1, the limiting process for the diffusion LMS is a differential inclusion (78). This suggests that, at each time, the elements of $M_{\boldsymbol{z}_n}$ depend on $\boldsymbol{z}_n$ drawn from the set of correlated ϵ-equilibria $\mathcal{C}_\epsilon(\boldsymbol{\theta}_n)$. On the other hand, weak convergence of $\boldsymbol{z}_n$ to $\mathcal{C}_\epsilon(\boldsymbol{\theta}_n)$ is convergence to a set rather than a particular point in that set. In fact, even if $\boldsymbol{\theta}_n = \widehat{\boldsymbol{\theta}}$ is held fixed, $\boldsymbol{z}_n$ can generally move around in the polytope $\mathcal{C}_\epsilon(\widehat{\boldsymbol{\theta}})$. Since the game-theoretic activation mechanism is run on the faster timescale, its behavior in the slower timescale (diffusion LMS) can be modeled by arbitrarily picking points from $\mathcal{C}_\epsilon(\boldsymbol{\theta}_n)$ and constructing $M_{\boldsymbol{z}_n}$ accordingly.

In our simulations, we use Matlab's `cprnd` function, which draws samples from the uniform distribution over the interior of a polytope defined by a system of linear inequalities $Ax < b$. (Theoretically, finding a correlated ϵ-equilibrium in a game is equivalent to solving a linear feasibility problem of the form $A\boldsymbol{\pi} < \epsilon I$; see (53).) We then take the sample $\boldsymbol{\pi}^c_{\boldsymbol{\theta}_n}$ and marginalize it on each node to obtain activation strategies. Uniform distribution for sampling from $\mathcal{C}_\epsilon(\boldsymbol{\theta}_n)$ is no loss of generality since it assumes no prior on the interior points of $\mathcal{C}_\epsilon(\boldsymbol{\theta}_n)$ (which matches the essence of convergence to a set). This reduces the complexity of our numerical study to a one-level (rather than a two-level) stochastic simulation.

C. Numerical Example

Fig. 2 depicts the network topology we study in this example. We define $Q = [q_{kl}]$ in (21) as

$$q_{kl} = \begin{cases} \frac{1}{d_{\mathcal{G}}(k)}, & l \in \mathcal{N}_k^c, \\ 0, & \text{otherwise} \end{cases}. \tag{83}$$

This is the most natural choice that first comes into mind and is enough for our demonstration purposes. One can however employ a smart adaptive weighting scheme to achieve better performance [1]. Fig. 2 further illustrates the statistics of each node's regressors, generated by a Gaussian Markov source with local correlation function of the form $r(\tau) = \sigma^2_{h,k}\alpha^{|\tau|}_{h,k}$, where $\alpha_{h,k}$ is the correlation index. In our simulations, we assume nodes are equipped with Chipcon CC2420 transceiver chipset which implements CSMA/CA protocol for exchanging estimates with neighbors. The reader is referred to [12, Appendix] for detailed model description and expressions for $E_{\mathrm{Tx}}(\cdot)$ ($k_c = k_m = 1$)

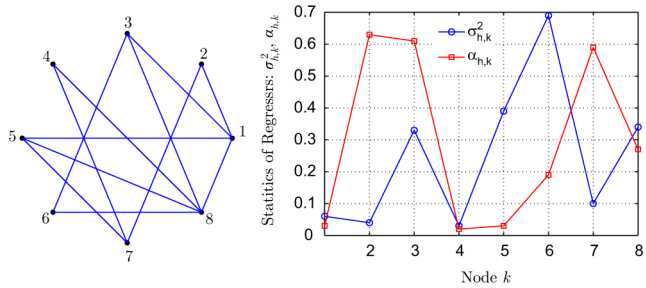

Fig. 2. Network topology and nodes' regressors statistics.

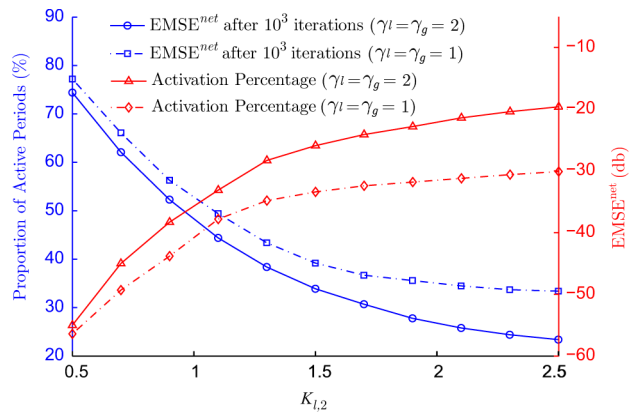

Fig. 3. Trade-off between energy expenditure and estimation accuracy after 1000 iterations.

and E_{Act} in (36). We further assume the noise v^k_n is i.i.d. with $\sigma^2_{v,k} = 10^{-3}$, and $\varepsilon = 0.01$.

Define the network excess mean square error:

$$\mathrm{EMSE}^{\mathrm{net}} \triangleq \frac{1}{K}\sum_{k=1}^{K}\mathrm{EMSE}_k, \tag{84}$$

where EMSE_k denotes the EMSE for node k. $\mathrm{EMSE}^{\mathrm{net}}$ is simply obtained by averaging $\mathbb{E}|h^k_n(\overline{\theta} - \phi^{k-1}_n)|^2$ across all nodes in the network. Fig. 3 demonstrates the trade-off between energy expenditure in the network and the rate of convergence of the diffusion LMS algorithm in terms of $\mathrm{EMSE}^{\mathrm{net}}$ after 1000 iterations. Nodes become more conservative by increasing the energy consumption parameter $K_{l,2}$ and, accordingly, activate less frequently due to receiving lower utilities; see (36). This reduces the average proportion of active nodes and increases $\mathrm{EMSE}^{\mathrm{net}}$ due to recording fewer measurements and less frequent fusion of neighboring estimates. Increasing the pricing parameters γ_l, in (36), and γ_g, in (37), has the same effect as can be observed in Fig. 3. The global performance of Algorithm 1 is further compared with the standard diffusion LMS [1] in Fig. 4. As the pricing parameters $K_{l,1}$ in (36) (corresponding to the contribution of nodes in local parameter estimation) and K_g in (37) (corresponding to the contribution of nodes in local diffusion) increase, nodes activate more frequently. As shown in Fig. 4, this improves the rate of convergence of the energy-aware diffusion LMS algorithm across the network.

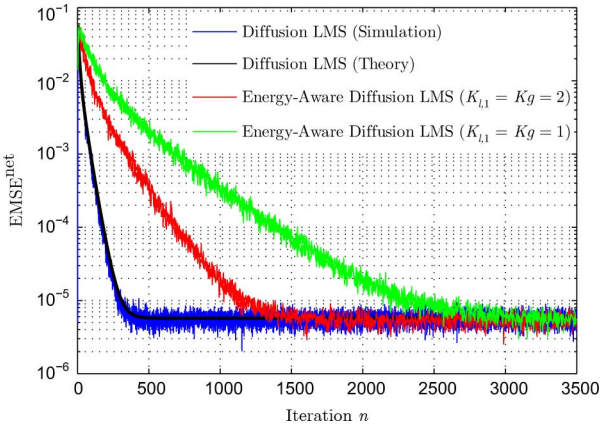

Fig. 4. Network excess mean square error EMSE^{net}.

VIII. Conclusion

We considered energy-aware parameter estimation via diffusion LMS by equipping nodes with a game-theoretic activation mechanism. The small step-size diffusion LMS was revised such that data fusion takes place on the same timescale as the assimilation of measurements. The revised diffusion LMS further allowed us to employ the powerful ODE method [5], [6] to analyze performance, which led to simpler derivation of the known stability results for diffusion LMS. The energy-aware activation control was formulated as a noncooperative repeated game, where nodes repeatedly decide whether to activate based on a utility function that captures the trade-off between energy consumption and the value of the data provided by the node. Finally, the diffusion LMS was combined with a game-theoretic adaptive filter that updates the activation strategy of nodes and adapts it to the changes in the network. The convergence analysis revealed that the proposed energy-aware diffusion LMS is asymptotically consistent, yet the global activation behavior at each time tracks the set of approximate correlated equilibria of the underlying activation control game. Simulation results were further provided to confirm theoretical findings.

Appendix
Proof of Theorem 2.1

Here, we present the results of Theorem 2.1 with more technical details followed by their proofs.

Theorem A.1: Consider the interpolated process $\widetilde{\boldsymbol{\theta}}^{\varepsilon}(\cdot)$, defined in (32). The following assertions hold:

1) Under (C2), $\widetilde{\boldsymbol{\theta}}^{\varepsilon}(\cdot)$ is tight in $D([0,\infty):\mathbb{R}^{KM})$.
2) Replace (25)–(26) in (C2) by:
   (i) $1/m\sum_{\tau=n}^{n+m}\mathbb{E}_n^k\, h_\tau^{k\prime} v_\tau^k \to 0$,
   (ii) $1/m\sum_{\tau=n}^{n+m}\mathbb{E}_n^k\, h_\tau^{k\prime} h_\tau^k \to R_h^k$, in probability as $m\to\infty$.
   Assume further $\widetilde{\boldsymbol{\theta}}_0$ is independent of ε. Then, $\widetilde{\boldsymbol{\theta}}^{\varepsilon}(\cdot)$ converges weakly to $\widetilde{\boldsymbol{\theta}}(\cdot)$ as $\varepsilon\to 0$ such that the limit is the solution to the ODE (33).
3) The linear system (33) is globally asymptotically stable and $\lim_{t\to\infty}\|\widetilde{\boldsymbol{\theta}}(t)\| = \mathbf{0}$.
4) Denote by $\{q(\varepsilon)\}$ any sequence of real numbers satisfying $q(\varepsilon)\to\infty$ as $\varepsilon\to 0$. Then, $\widetilde{\boldsymbol{\theta}}^{\varepsilon}(\cdot + q(\varepsilon))$ converges weakly to $\mathbf{0}$ as $\varepsilon\to 0$.

Proof: To prove part 1), we apply the tightness criterion [24, p.47]. It suffices that for any $\delta > 0$ and $t, s > 0$ such that $s < \delta$, $\lim_{\delta\to 0}\limsup_{\mu\to 0}\sup_{0\le s\le\delta}\mathbb{E}[\mathbb{E}_t^{\mu}|\boldsymbol{\theta}^{\mu}(t+s) - \boldsymbol{\theta}^{\mu}(t)|^2] = 0$, where $\mathbb{E}_t^{\mu}$ denotes the σ-algebra generated by the μ-dependent past data up to time t. Using (C1), direct calculation leads to

$$\mathbb{E}_t^{\mu}\left|\widetilde{\boldsymbol{\theta}}^{\varepsilon}(t+s) - \widetilde{\boldsymbol{\theta}}^{\varepsilon}(t)\right|^2 \le K\varepsilon^2\left(\frac{t+s}{\varepsilon} - \frac{t}{\varepsilon}\right)^2 = \mathcal{O}(\delta^2). \quad (85)$$

The desired tightness follows by first taking $\limsup_{\mu\to 0}$ and then $\lim_{\delta\to 0}$ in (84). Tightness is equivalent to sequential compactness on any complete separable metric space. Thus, by virtue of the Prohorov's theorem, we may extract a weakly convergent subsequence. For notational simplicity, still denote the subsequence by $\widetilde{\boldsymbol{\theta}}^{\varepsilon}(\cdot)$ with limit $\widetilde{\boldsymbol{\theta}}(\cdot)$.

The proof of part 2) uses stochastic averaging theory and follows from the standard argument in [6, Chapter 8] to characterize the limit $\widetilde{\boldsymbol{\theta}}(\cdot)$ of the the convergent subsequence $\widetilde{\boldsymbol{\theta}}^{\varepsilon}(\cdot)$. Details are tedious and omitted due to limited space. We merely note that, in view of (21) and (24), $B = I_{KM} + \varepsilon\widehat{B}$. Looking at the last term in the r.h.s. of (31), only I_{KM} survives in the limit and the term $\varepsilon\widehat{B}$ is averaged out to $\mathbf{0}$.

To prove part 3), we use standard results on Lyapunov stability of linear systems. In view of (21) and (24), since $|\widehat{b}_{kk}| = \sum_{l\ne k}\widehat{b}_{kl} = \sum_{l\ne k}|\widehat{b}_{kl}|$ for all k, the symmetric matrix $\widehat{B}$ is weakly diagonally dominant, hence, is negative semi-definite $\widehat{B} \le \mathbf{0}$. On the other hand, $R_h^d = (R_h^d)' > \mathbf{0}$. Therefore, $\widehat{B} - R_h^d < \mathbf{0}$ (negative-definite). The set of global attractors of (33) is defined by $\mathcal{A}_g = \{\widetilde{\boldsymbol{\theta}} | (\widehat{B} - R_h^d)\widetilde{\boldsymbol{\theta}} = \mathbf{0}\}$. Since $\det(\widehat{B} - R_h^d) = \prod_i \lambda(\widehat{B} - R_h^d) < 0$, $\widehat{B} - R_h^d$ is nonsingular and $(\widehat{B} - R_h^d)\widetilde{\boldsymbol{\theta}} = \mathbf{0}$ has the unique solution $\widetilde{\boldsymbol{\theta}} = \mathbf{0}$. Therefore, $\lim_{t\to\infty}\|\widetilde{\boldsymbol{\theta}}(t)\| = 0$.

To prove Part 4), we need to establish the limit as first $\varepsilon\to 0$ and then $t\to\infty$. The consequence is that the limit points of the ODE and the small step-size diffusion LMS recursions coincide as $t\to\infty$. However, in lieu of considering a two stage limit by first letting $\varepsilon\to 0$ and then $t\to\infty$, we look at $\widetilde{\boldsymbol{\theta}}^{\varepsilon}(t + q(\varepsilon))$ and require $q(\varepsilon)\to\infty$. The interested reader is referred to [6, Chapter 8] for an extensive treatment. ■

References

[1] C. G. Lopes and A. H. Sayed, "Diffusion least-mean squares over adaptive networks: Formulation and performance analysis," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 3122–3136, Jul. 2008.
[2] F. Cattivelli and A. Sayed, "Diffusion LMS strategies for distributed estimation," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1035–1048, Mar. 2010.
[3] S.-Y. Tu and A. H. Sayed, "Diffusion strategies outperform consensus strategies for distributed estimation over adaptive networks," *IEEE Trans. Signal Process.*, vol. 60, no. 12, pp. 6217–6234, Dec. 2012.
[4] C. G. Lopes and A. H. Sayed, "Incremental adaptive strategies over distributed networks," *IEEE Trans. Signal Process.*, vol. 55, no. 8, pp. 4064–4077, Aug. 2007.
[5] A. Benveniste, M. Metivier, and P. Prioret, *Adaptive Algorithms and Stochastic Approximations*. New York, NY, USA: Springer-Verlag, 1990.
[6] H. J. Kushner and G. Yin, *Stochastic Approximation and Recursive Algorithms and Applications*, 2nd ed. New York, NY, USA: Springer-Verlag, 2003.

[7] R. J. Aumann, "Correlated equilibrium as an expression of bayesian rationality," *Econometrica*, vol. 55, no. 1, pp. 1–18, Jan. 1987.
[8] S. Hart and A. Mas-Colell, "A simple adaptive procedure leading to correlated equilibrium," *Econometrica*, vol. 68, no. 5, pp. 1127–1150, Sep. 2000.
[9] S. Hart and A. Mas-Colell, "A reinforcement procedure leading to correlated equilibrium," *Economic Essays: A Festschrift for Werner Hildenbrand*, pp. 181–200, 2001.
[10] A. Zoubir, V. Krishnamurthy, and A. H. Sayed, "Signal processing theory and methods (in the spotlight)," *IEEE Signal Process. Mag.*, vol. 28, no. 5, pp. 152–156, Sep. 2011.
[11] L. Wang and Y. Xiao, "A survey of energy-efficient scheduling mechanisms in sensor networks," *Mobile Netw. Appl.*, vol. 11, no. 5, pp. 723–740, Oct. 2006.
[12] V. Krishnamurthy, M. Maskery, and G. Yin, "Decentralized adaptive filtering algorithms for sensor activation in an unattended ground sensor network," *IEEE Trans. Signal Process.*, vol. 56, no. 12, pp. 6086–6101, Dec. 2008.
[13] O. Gharehshiran and V. Krishnamurthy, "Coalition formation for bearings-only localization in sensor networks-a cooperative game approach," *IEEE Trans. Signal Process.*, vol. 58, no. 8, pp. 4322–4338, Aug. 2010.
[14] A. H. Sayed, *Diffusion Adaptation Over Networks E-Reference Signal-Processing*, S. Theodoridis and R. Chellapa, Eds. Amsterdam, The Netherlands: Elsevier, 2013.
[15] O. N. Gharehshiran, V. Krishnamurthy, and G. Yin, "Distributed tracking of correlated equilibria in regime switching noncooperative games," *IEEE Trans. Autom. Control*, 2013.
[16] A. H. Sayed, *Fundamentals of Adaptive Filtering*. New York, NY, USA: Wiley, 2003.
[17] S. Kar, J. Moura, and K. Ramanan, "Distributed parameter estimation in sensor networks: Nonlinear observation models and imperfect communication," *IEEE Trans. Inf. Theory*, vol. 58, no. 6, pp. 3575–3605, Jun. 2012.
[18] S. Hart, "Adaptive heuristics," *Econometrica*, vol. 73, no. 5, pp. 1401–1430, Sep. 2005.
[19] R. Nau, S. G. Canovas, and P. Hansen, "On the geometry of Nash equilibria and correlated equilibria," *Intl. J. Game Theory*, vol. 32, no. 4, pp. 443–453, 2004.
[20] S. Hart and A. Mas-Colell, "Uncoupled dynamics do not lead to Nash equilibrium," *Amer. Econom. Rev.*, vol. 93, no. 5, pp. 1830–1836, 2003.
[21] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1989.
[22] M. Benaim, J. Hofbauer, and S. Sorin, "Stochastic approximations and differential inclusions; Part II: Applications," *Math. Oper. Res.*, vol. 31, no. 3, pp. 673–695, Nov. 2006.
[23] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 1990.
[24] H. J. Kushner, *Approximation and Weak Convergence Methods for Random Processes With Applications to Stochastic Systems Theory*. Cambridge, MA, USA: MIT Press, 1984, vol. 6.

Omid Namvar Gharehshiran (S'09) received his B.Sc. from Sharif University of Technology, Tehran, Iran, in 2007, and the M.A.Sc. from University of British Columbia, Vancouver, Canada, in 2010. He is currently working towards his Ph.D. and is a member of the Statistical Signal Processing Laboratory. He was the recipient of the UBC Four Year Doctoral Fellowship in 2010. His research interests span stochastic optimization, games and learning theory.

Vikram Krishnamurthy (F'05) was born in 1966. He received his bachelor's degree from the University of Auckland, New Zealand in 1988, and Ph.D. from the Australian National University in 1992. He currently is a professor and Canada Research Chair at the Department of Electrical Engineering, University of British Columbia, Vancouver, Canada.

Dr. Krishnamurthy's current research interests include game theory, stochastic control in sensor networks, and modeling of biological ion channels and biosensors. Dr. Krishnamurthy served as Editor in Chief of IEEE JOURNAL OF SELECTED TOPICS IN SIGNAL PROCESSING and as distinguished lecturer for the IEEE signal processing society. He has also served as associate editor for several journals including IEEE TRANSACTIONS ON AUTOMATIC CONTROL and IEEE TRANSACTIONS ON SIGNAL PROCESSING.

George Yin (F'02) received the B.S. degree in mathematics from the University of Delaware in 1983, M.S. degree in Electrical Engineering, and Ph.D. in Applied Mathematics from Brown University in 1987. He joined Wayne State University, Detroit, MI, in 1987, and became a professor in 1996. His research interests include stochastic systems theory, stochastic approximation, control, and applications.

He severed on many technical & conference committees; he co-chaired the 2011 SIAM Control Conference, the 1996 AMS-SIAM Summer Seminar, and the 2003 AMS-IMS-SIAM Summer Research Conference, and co-organized 2005 IMA Workshop on Wireless Communications. He has served as Vice Chair and Program Director of SIAM Activity Group on Control and Systems Theory; he also served as chair of the SIAM W.T. and Idalia Reid Prize Selection Committee, the SIAG/Control and Systems Theory Prize Selection Committee, and the SIAM SICON Best Paper Prize Committee. He is an associate editor of *SIAM Journal on Control and Optimization*, and is on the editorial board of a number of other journals. He was an Associate Editor of Automatica and IEEE TRANSACTIONS ON AUTOMATIC CONTROL. He is President of Wayne State University's Academy of Scholars.