LINK PROBABILITY CONTROL FOR PROBABILISTIC DIFFUSION LEAST-MEAN SQUARES OVER RESOURCE-CONSTRAINED NETWORKS

Noriyuki Takahashi
Tokyo Institute of Technology
Global Edge Institute
2-12-1-S3-60, Tokyo 152-8550, Japan
E-mail: takahashi@comm.ss.titech.ac.jp

Isao Yamada
Tokyo Institute of Technology
Dept. of Communications and Integrated Systems
2-12-1-S3-60, Tokyo 152-8550, Japan
E-mail: isao@comm.ss.titech.ac.jp

ABSTRACT

This paper presents an efficient link probability control strategy for distributed estimation problems over probabilistic diffusion networks where the mean usage of communication resources is restricted. The proposed algorithm controls link probabilities so that the estimation error is minimized under given resource constraints. Simulation results show that the probabilistic diffusion least-mean squares (LMS) algorithm with the proposed probability control not only outperforms those with static probabilities but also reduces the amount of communications among nodes.

Index Terms—Adaptive filter, distributed estimation, adaptive networks, probabilistic diffusion, probability control

1. INTRODUCTION

We consider the problem of distributed estimation over adaptive networks [1–7] where, to estimate some parameter of interest, communications among nodes are allowed but the amount of network traffic is restricted. Among several cooperation strategies [1–3], the probabilistic diffusion strategy [3] will be suitable for such resource-constrained networks because it is capable of reducing communications among the nodes without presenting a significant decrease in performance, by randomly selecting a subset of neighbors to cooperate. Therefore, the link probability, which determines how frequently communications occur, plays a key role in the probabilistic diffusion in terms of both performance and communications. The goal of this paper is to establish an efficient link probability control strategy in order to improve the system performance to a maximum extent while keeping the communication cost as low as possible.

In this paper, we formulate the problem of controlling link probabilities in a systematic way. Although a straightforward formulation leads to a non-convex optimization problem, we find an asymptotically optimal convex relaxation of the original problem. We then use the problem to propose a link probability control algorithm. Here, the hybrid steepest-descent method [8] is employed to solve the problem by a combination of simple mathematical tools: gradients and projections [9]. This leads to an adaptive implementation of the proposed algorithm. Simulation results show that the probabilistic diffusion least-mean squares (LMS) algorithm with the proposed probability control successfully improves the system performance while reducing communications as much as possible.

2. PROBABILISTIC DIFFUSION LMS

To begin with, let us introduce our notation. We use boldface letters for random variables and normal fonts for deterministic quantities. Capital letters are used for matrices and small letters for vectors. The notation $(\cdot)^T$ and $(\cdot)^*$ stand for transposition and conjugate transposition for vectors and matrices, respectively. Expectation is denoted by $E[\cdot]$.

Consider $N$ nodes in a predefined network topology; see Fig. 1. We denote by $N_k$ the neighborhood of node $k$ including $k$ itself and by $n_k$ the degree of node $k$, i.e., the cardinality of $N_k$. At each time $t \geq 0$, each node $k$ has access to a scalar measurement $d_k(\ell) \in \mathbb{C}$ and a regression row vector $u_{k,i} \in \mathbb{C}^{1 \times M}$ of length $M$ that are related via

$$d_k(\ell) = u_{k,i}w^\alpha + v_k(\ell),$$

where $w^\alpha \in \mathbb{C}^M$ is an unknown column vector we wish to estimate and $v_k(\ell) \in \mathbb{C}$ accounts for noise and modeling errors. The objective of the distributed estimation problem is to generate an estimate $\hat{v}_{k,i}$ of $w^\alpha$ at each node $k$ and time $t$ in a distributed manner [1–7].

In the probabilistic diffusion mode [3], at each time $t \geq 0$, every node $k$ selects its instantaneous neighborhood $N_{k,i} \subseteq N_k$ at random. To represent this dynamics, we introduce a Bernoulli processes $\chi_{\ell k} = \{\chi_{\ell k}(i), i \geq 0\}$ for each pair $(\ell, k) \in \{1, \ldots, N\}^2$:

$$\chi_{\ell k}(i) \overset{\Delta}{=} \begin{cases} 1 & \text{with probability } p_{\ell k}, \\ 0 & \text{with probability } 1 - p_{\ell k}. \end{cases}$$

When $\chi_{\ell k}(i) = 1$, the link from node $\ell$ to node $k$ is activated and node $k$ receives the estimate $\hat{v}_{\ell k}(t)$ from node $\ell$ (i.e., the random matrix $[\chi_{\ell k}(i)]_{\ell \in N_k}$ represents the adjacency matrix of a topology at time $t$). Here, for each $k = 1, \ldots, N$, we impose $p_{\ell k} = 1$ and $p_{\ell k} = 0$ for $\ell \notin N_k$ for convenience. Note that, in this model, we assume that instantaneous topologies can be directed, i.e., one-way communications are allowed. Using $\{\chi_{\ell k}(i)\}_{i=1}^{N}$, the neighborhood $N_{k,i}$ can be expressed as $N_{k,i} = \{\ell \in N_k : \chi_{\ell k}(i) = 1\}$.

In this paper, we focus on the Adapt-then-Combine (ATC) diffusion strategy [5–7]. For example, using the LMS algorithm as the core adaptive filter leads to the ATC probabilistic diffusion LMS al-

Fig. 1: A distributed network with $N$ nodes.
3. LINK PROBABILITY CONTROL

We first introduce a model for resource constrains and then formulate a problem that determines an optimal probability assignment \( \{ p_{\ell k} \}_{\ell,k=1}^N \) for algorithm (2). Then, the hybrid steepest-descent method [8] is used to approximate the optimal probabilities, leading to a control method for the link probabilities. In what follows, we only highlight main results due to lack of space.

3.1. Resource Constraints

Let us take communications cost into account by assuming that some cost \( a_{\ell k} \in \mathbb{R} \) is required at node \( \ell \) for communications with node \( k \). We assume that \( a_{\ell k} = 0 \) for each \( k \) (values of \( a_{\ell k} \) for \( \ell \notin \mathcal{N}_k \) may be arbitrary because these values have no effect on our formulation). Then, the average cost necessary at node \( k \) is given by

\[
E \left[ \sum_{\ell \in \mathcal{N}_k \setminus \{k\}} a_{\ell k} x_{\ell k}(i) \right] = \sum_{\ell \in \mathcal{N}_k \setminus \{k\}} a_{\ell k} p_{\ell k} = a_k^T L_k L_k^T \mathbf{p}_k, \tag{3}
\]

where \( a_k = (a_{1k}, \ldots, a_{Nk})^T \in \mathbb{R}^N \), \( p_k = (p_{1k}, \ldots, p_{Nk})^T \in [0,1]^N \), and \( L_k \) is an \( N \times (n_k - 1) \) matrix defined as

\[
L_k \triangleq [e_\ell]_{\ell \in \mathcal{N}_k \setminus \{k\}}, \quad e_\ell \triangleq \ell\text{-th column of } I_N.
\]

Here, \( L_k \) is introduced just for notational convenience: \( L_k^T \mathbf{p}_k \in \mathbb{R}^{n_k-1} \) is the subvector of \( \mathbf{p}_k \) consisting of those components that correspond to indices \( \ell \in \mathcal{N}_k \setminus \{k\} \). We control \( p_k \) so that the average cost (3) does not exceed a given amount \( \alpha_k \in \mathbb{R} \). Namely, we impose \( p_k \in [0,1]^N \) and

\[
L_k^T \mathbf{p}_k \in S_k \triangleq \{ x \in \mathbb{R}^{n_k-1} : a_k^T L_k x \leq \alpha_k \} \quad \text{for all } k. \tag{4}
\]

In this paper, we call \( \alpha_k \) target cost for node \( k \).

3.2. Problem Formulation

Suppose that for each \( k \in \{1, \ldots, N\} \) the intermediate estimates \( \{ \phi_{k,i} : i \geq 0 \} \) are realizations of some random vectors \( \{ \hat{\phi}_k \}_{k=1}^N \). We further define, for each \( k \in \{1, \ldots, N\} \),

\[
H_k \triangleq \left[ c_{\ell k}(\phi_{\ell} - \hat{\phi}_{k}) \right]_{\ell=1}^N, \quad X_k \triangleq \left( x_{1k}, x_{2k}, \ldots, x_{Nk} \right)^T,
\]

where \( H_k \) is an \( M \times N \) complex random matrix and \( X_k \) is an \( N \times 1 \) random vector whose \( \ell\)-th component follows a Bernoulli distribution. Here, the minimum probability \( p_{\ell k} \) is assumed to be independent of each other. Then, the estimates \( \{ \psi_{k,i} : i \geq 0 \} \) generated by the convex combination (2) are realizations of the random variable

\[
\psi_k \triangleq \phi_k + H_k L_k L_k^T X_k.
\]

Our approach is to minimize the mean square deviation (MSD) of \( \phi_k \) at node \( k \), i.e., we consider the following problem for each \( k \):

\[
\begin{align*}
\text{minimize} & \quad J_k(p_k) \triangleq E[\|\psi_k - w^o\|^2] \\
\text{subject to} & \quad L_k^T p_k \in B_k \cap S_k, \\
& \quad p_{\ell k} = 1 \quad \text{and} \quad p_{\ell k} = 0 \quad \text{for all } \ell \notin \mathcal{N}_k,
\end{align*}
\]

where \( S_k \) is defined in (4) and \( B_k \) is defined as follows:

\[
B_k \triangleq [p_{\min}, p_{\max}]^{n_k-1} \subset [0, 1]^{n_k-1}.
\]

Here, the minimum probability \( p_{\min} \) is important: once \( p_{\ell k} \) is set to zero, the link \( (\ell, k) \) will never be activated. To maintain the network connection, \( p_{\min} \) should be a nonzero value in practice.

Now, assuming that \( X_k \) is independent of both \( \psi_k \) and \( H_k \) and that \( E[H_k] = 0 \) for all \( k \in \{1, \ldots, N\} \), we obtain

\[
J_k(p_k) = f_k(L_k^T p_k) + E[\|\phi_k - w^o\|^2],
\]
where the function $f_k : \mathbb{R}^{nk-1} \to \mathbb{R}$ is defined as follows:

$$f_k(x) = x^T Q_k x + 2 r_k^T x + \operatorname{tr}(Q_k D(x)),$$

$$D(x) = \operatorname{diag}(x(1 - x), \ldots, x_{nk-1}(1 - x_{nk-1})).$$

$$Q_k \triangleq L_k^T \operatorname{Re}(\mathbb{E}[H_k^* H_k]) L_k,$$

$$r_k \triangleq L_k^T \operatorname{Re}(\mathbb{E}[H_k^* ( \phi_k - E\phi_k )]).$$

Hence, by introducing an auxiliary variable $q_k \triangleq L_k^T p_k \in \mathbb{R}^{nk-1}$, problem (5) can be reformulated as follows:

$$\begin{align*}
\text{minimize} & \quad q_k \in \mathbb{R}^{nk-1} \quad f_k(q_k) = \frac{1}{2} q_k^T Q_k q_k + r_k^T q_k + \frac{1}{2} \operatorname{tr}(Q_k D(x)) \\
\text{subject to} & \quad q_k \in B_k \cap S_k.
\end{align*}
$$

(6)

(Note: $p_k$ is recovered via $p_k = L_k q_k + e_k$. Although problem (6) is non-convex, we can find a quadratic function that approximates $f_k$ on $B_k$:

**Theorem 1.** The function $\tilde{f}_k(x) \triangleq \frac{1}{2} x^T Q_k x + r_k^T x$ satisfies

$$\tilde{f}_k(x) \leq f_k(x) \leq \tilde{f}_k(x) + \frac{(n_k - 1) \max_x E\|\phi_k - \phi_k\|^2}{8(1 + (n_k - 2) \min \max_{x_k, \epsilon_k})}$$

for all $x \in B_k$, where the min and max are taken over $N_k \setminus \{k\}$.

This theorem states that the difference between $f_k$ and $\tilde{f}_k$ becomes negligible on $B_k$ as $n_k$ increases, provided that there exists a positive constant $\epsilon$ that is independent of $n_k$ and such that $\min \max_{x_k, \epsilon_k} \epsilon_k \geq \epsilon$. In the light of Theorem 1, we consider the following problem:

$$\begin{align*}
\text{minimize} & \quad \tilde{f}_k(q_k) = \frac{1}{2} q_k^T Q_k q_k + r_k^T q_k \\
\text{subject to} & \quad q_k \in B_k \cap S_k.
\end{align*}
$$

(7)

This is a quadratic program (QP) and many algorithms are available to solve the problem [8, 9]. In the paper, we propose a gradient-type solution for the purpose of adaptive implementation.

### 3.3. Iterative and Adaptive Solutions

We utilize the gradient $\nabla \tilde{f}$ and projections\(^1\) by employing the hybrid steepest-descent method [8], which enables us to use the projections onto individual sets $B_k$ and $S_k$, say $P_{B_k}$ and $P_{S_k}$. Here, the closed form expressions for $P_{B_k}$ and $P_{S_k}$ are easily found to be

$$\begin{align*}
[P_{B_k}(x)]_j &= \max\{p_{\min}, \min\{p_{\max}, x_j\}\}, \\
P_{S_k}(x) = x - \frac{\max(0, a_k^T L_k x - x_k)}{L_k a_k^T L_k} L_k a_k,
\end{align*}$$

where $[P_{B_k}(x)]_j$ stands for the j-th component of $P_{B_k}(x)$. By using these projections and the gradient $\nabla \tilde{f}(x) = Q_k x + r_k$, we obtain one implementation of the hybrid steepest-descent method:

$$q_{k,i+1} = P_{B_k} P_{S_k} [q_k - \nu_{k,i} \nabla \tilde{f}(q_{k,i})],$$

(8)

where $\nu_{k,i} \geq 0$ is a stepsize. Note that algorithm (8) guarantees $(q_{k,i})_{i \geq 0} \subset B_k$. The sequence $(q_{k,i})_{i \geq 0}$ generated by (8) converges to the solution of (7), if the stepsize $\nu_{k,i}$ satisfies $\nu_{k,i} \to 0$

\(^1\)Given a convex set $C \subset \mathbb{R}^{nk-1}$, the projection of $x \in \mathbb{R}^{nk-1}$ onto $C$, denoted by $P_C(x)$, is defined as a unique point in $C$ that is closest to $x$; see, e.g., [9] for details.

and $\sum_{i=0}^{\infty} \nu_{k,i} = +\infty$; see for details [8]. Unfortunately, such a decreasing stepsize is not suitable for the adaptive estimation. However, any kind of stepsizes, such as constants and normalized ones, can be used to approximate the solution to problem (7).

We finally replace $Q_k$ and $r_k$ by their instantaneous approximations to derive an adaptive version of (8). Since node $k$ has no access to neighbors $\ell \notin N_{k,i}$, only a part of realizations of $H_k$ is available at each time. With this in mind, we use the following approximations:

$$Q_k \approx \tilde{Q}_{k,i} \triangleq L_k^T \Gamma_k L_k,$$

$$r_k \approx \tilde{r}_{k,i} \triangleq L_k^T \gamma_{k,i}.$$

(9)

where

$$[\Gamma_{k,i}]_{\ell m} = \begin{cases} c_{\ell k} c_{m k} (\phi_{\ell,k} - \phi_{m,k})^T (\phi_{m,k} - \phi_{k,i}) & \text{if } \ell, m \in N_{k,i}, \\
0 & \text{if } \ell, m \notin N_{k,i}. \end{cases}$$

$$[\gamma_{k,i}]_{\ell} = \begin{cases} c_{\ell k} (\phi_{\ell,k} - \phi_{k,i})^T (\phi_{k,i} - \phi_{k,i-1}) & \text{if } \ell \in N_{k,i}, \\
0 & \text{if } \ell \notin N_{k,i}. \end{cases}$$

Note that all the components of $\tilde{Q}_{k,i}$ and $\tilde{r}_{k,i}$ can be calculated using information available at node $k$. Any other approximations are available and will be reported elsewhere.

The probabilistic diffusion LMS algorithm with the proposed probability control is summarized as follows:

**Probabilistic Diffusion LMS with Probability Control**

**Initialization:** At each node $k$, choose $q_{k,0} \in B_k \cap S_k$ and set $q_{k,-1} = \phi_{k,-1} = 0$, $\Gamma_{k,-1} = 0$, and $\gamma_{\ell,-1} = 0$, then, repeat the following for every $i \geq 0$:

1. **Adaptation:** $\psi_{k,i} = \psi_{k,i-1} + \nu_{k,i} (d_k(i) - u_k(i) \psi_{k,i-1}).$

2. **Exchange:** Randomly select a neighborhood $N_{k,i} \subset N_k$ with probability $p_{k,i} = L_k q_{k,i} + e_k$ and receive the intermediate estimates $\{\phi_{\ell,k}\}_{\ell \in N_{k,i}}$.

3. **Combination:** $\psi_{k,i} = \phi_{k,i} + \sum_{\ell \in N_{k,i}} c_{\ell k} (\phi_{\ell,k} - \phi_{k,i}).$

4. **Probability Update:** Update $\tilde{Q}_{k,i}$ and $\tilde{r}_{k,i}$, e.g., using (9). Then, update $q_{k,i}$ by $q_{k,i+1} = P_{B_k} P_{S_k} [q_k - \nu_{k,i} (\tilde{Q}_{k,i} q_{k,i} + \tilde{r}_{k,i})].$

### 4. SIMULATION RESULTS

We consider the network topology with $N = 15$ nodes of Fig. 3(a). The regressors, $u_k(i) = (u_k(i), \ldots, u_k(i - M + 1))$, are generated via a first order autoregressive model: $u_k(i) = \rho u_k(i - 1) + x_k(i)$, where $(x_k(i))$ is zero-mean Gaussian, independent in time and space, with variance $(1 - \rho^2) \sigma^2_{u,k}$. The signal profile is depicted in Fig. 3(b). The noise $v_k(i)$ is also zero-mean Gaussian with variance $\sigma^2_{v,k} = 10^{-3}$ for all $k$. The unknown vector is set to $w^\ast = 1_5 / \sqrt{5}$ ($M = 5$). For the resource constraint, we use $a_{\ell k} = 1$ for all links $(\ell, k)$ and $\alpha_k = 3$ for all $k$, which means that at most three neighbors are selected in the mean. We set $[p_{\min}, p_{\max}] = [0.2, 1]$. We compare the ATC probabilistic diffusion LMS algorithm with the proposed probability control and those with constant probabilities listed in Fig 3(c). Here, probabilities for the uniform rule was chosen so that the resource constraint is satisfied. For all algorithms, the stepsize of the LMS is set to $\mu_k = 0.03$ and the combination weights $\{c_{\ell k}\}$ are calculated according to the Metropolis rule (see, e.g., [10]). Namely, all algorithms use the same LMS algorithm and only the probabilities are different. For the proposed probability control, the initial probabilities are set to the minimum probabilities $q_{k,0} = p_{\min,1} \{u_k \in S_k \}$ for all $k$ or set to the uniform probabilities, and
we use a normalized stepsize \( \nu_k, i = 0.01/(\|g_k, i\| + 10^{-4}) \) for each \( k \), where \( g_k, i = Q_k, i q_k, i + \bar{r}_k, i \). Figure 4 shows the learning behavior of each algorithm in terms of the network mean-square deviation (MSD):

\[
\eta_{\text{network}}(i) \triangleq N^{-1} \sum_{k=1}^{N} \eta_k(i), \quad \eta_k(i) \triangleq E[\|w^o - \psi_{k, i}\|^2],
\]

where \( \eta_k(i) \) is the MSD at node \( k \) and the expectation is calculated by averaging 100 independent experiments. On the other hand, Figure 5 shows the steady-state MSD at each node, which is obtained by averaging the last 1000 samples after \( 2 \times 10^6 \) iterations. We observe that the proposed algorithm starting at the uniform probabilities outperforms the uniform rule. Figure 6 presents the instantaneous network traffic, where the capacity is defined as the maximum number of available links, which the standard diffusion wastes. An interesting observation is that the proposed algorithms improve the steady-state performance in spite of reducing the network traffic.

5. CONCLUDING REMARKS

An efficient probability control for the probabilistic diffusion LMS algorithm over adaptive networks has been proposed to improve the performance under given communications resource constraints. Although we focused on the LMS algorithm as the adaptive filter module, combinations with other adaptive filters are possible. Our probability controller will be used to detect and remove unnecessary links, leading to energy-efficient distributed networks.

6. REFERENCES