Distributed Consensus Algorithms in Sensor Networks With Imperfect Communication: Link Failures and Channel Noise

Soummya Kar and José M. F. Moura

Abstract

The paper studies average consensus with random topologies (intermittent links) and noisy channels. Consensus with noise in the network links leads to the bias-variance dilemma—running consensus for long reduces the bias of the final average estimate but increases its variance. We present two different compromises to this tradeoff: the \( \mathcal{A} - \mathcal{N}D \) algorithm modifies conventional consensus by forcing the weights to satisfy a persistence condition (slowly decaying to zero;) and the \( \mathcal{A} - \mathcal{N}C \) algorithm where the weights are constant but consensus is run for a fixed number of iterations \( \hat{\tau} \), then it is restarted and rerun for a total of \( \hat{p} \) runs, and at the end averages the final states of the \( \hat{p} \) runs (Monte Carlo averaging). We use controlled Markov processes and stochastic approximation arguments to prove almost sure convergence of \( \mathcal{A} - \mathcal{N}D \) to the desired average (asymptotic unbiasedness) and compute explicitly the m.s.e. (variance) of the consensus limit. We show that \( \mathcal{A} - \mathcal{N}D \) represents the best of both worlds—low bias and low variance—at the cost of a slow convergence rate; rescaling the weights balances the variance versus the rate of bias reduction (convergence rate). In contrast, \( \mathcal{A} - \mathcal{N}C \), because of its constant weights, converges fast but presents a different bias-variance tradeoff. For the same number of iterations \( \hat{\tau}, \hat{p} \), shorter runs (smaller \( \hat{\tau} \)) lead to high bias but smaller variance (larger number \( \hat{p} \) of runs to average over.) For a static non-random network with Gaussian noise, we compute the optimal gain for \( \mathcal{A} - \mathcal{N}C \) to reach in the shortest run length \( \hat{\tau} \), with high probability \( (1 - \delta), (\epsilon, \delta) \)-consensus (\( \epsilon \) residual bias). Our results hold under fairly general assumptions on the random link failures and communication noise.

Keywords: Consensus, random topology, additive noise, sensor networks, stochastic approximation, convergence

The authors are with the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA, USA 15213 (e-mail: soummyak@andrew.cmu.edu, moura@ece.cmu.edu, ph: (412)268-6341, fax: (412)268-3890.)

Work supported by the DARPA DSO Advanced Computing and Mathematics Program Integrated Sensing and Processing (ISP) Initiative under ARO grant # DAAD19-02-1-0180, by NSF under grants # ECS-0225449 and # CNS-0428404, by an IBM Faculty Award, and by the Office of Naval Research under MURI N000140710747.
I. INTRODUCTION

Distributed computation in sensor networks is a well-studied field with an extensive body of literature (see, for example, [1] for early work.) Average consensus computes iteratively the global average of distributed data using local communications, see [2], [3], [4], [5] that consider versions and extensions of basic consensus. A review of the consensus literature is in [6]. Reference [7] designs the optimal link weights that optimize the convergence rate of the consensus algorithm when the connectivity graph of the network is fixed (not random). Our previous work, [8], [9], [10], [11], extends [7] by designing the topology, i.e., both the weights and the connectivity graph, under a variety of conditions, including random links and link communication costs, under a network communication budget constraint.

We consider distributed average consensus when simultaneously the network topology is random (link failures, like when packets are lost in data networks) and the communications among sensors is commonly noisy. A typical example is time division multiplexing, where, in a particular user’s time slot the channel may not be available, and, if available, we assume the communication is analog and noisy. Our approach can handle spatially correlated link failures through Markovian sequences of Laplacians and certain types of Markovian noise, which go beyond independently, identically distributed (i.i.d.) Laplacian matrices and i.i.d. communication noise sequences. Noisy consensus leads to a tradeoff between bias and variance. Running consensus longer reduces bias, i.e., the error between the desired average and the consensus reached. But, due to noise, the variance of the limiting consensus grows with longer runs. To address this dilemma, we consider two versions of consensus with link failures and noise that represent two different bias-variance tradeoffs: the $A - ND$ and the $A - NC$ algorithms.

$A - ND$ updates each sensor state with a weighted fusion of its current neighbors’ states (received distorted by noise). The fusion weights $\alpha(i)$ satisfy a persistence condition, decreasing to zero, but not too fast. $A - ND$ falls under the purview of controlled Markov processes and we use stochastic approximation techniques to prove its almost sure (a.s.) consensus when the network is connected on the average: the sensor state vector sequence converges a.s. to the consensus subspace. A simple condition on the mean Laplacian, $L = \mathbb{E}\{L\}$, for connectedness is on its second eigenvalue, $\lambda_2(L) > 0$. We establish that the sensor states converge asymptotically a.s. to a finite random variable $\theta$ and, in particular, the expected sensor states converge to the desired average $r$ (asymptotic unbiasedness.) We determine the variance of $\theta$, which is the mean square error (m.s.e.) between $\theta$ and the desired average. By properly tuning the weights sequence $\{\alpha(i)\}$, the variance of $\theta$ can be made arbitrarily small, though at a cost of slowing $A - ND$’s convergence rate, i.e., the rate at which the bias goes to zero.

$A - NC$ is a repeated averaging algorithm that performs in-network Monte-Carlo simulations: it runs consensus $\hat{p}$ times with constant weight $\alpha$, for a fixed number of iterations $\hat{r}$ each time and then each
sensor averages its $\hat{p}$ values of the state at the final iteration $\hat{\tau}$ of each run. $A - NC$’s constant weight $\alpha$ speeds its convergence rate relative to $A - ND$’s, whose weights $\alpha(i)$ decrease to zero. We determine the number of iterations $\hat{\tau} p$ required to reach $(\epsilon, \delta)$-consensus, i.e., for the bias of the consensus limit at each sensor to be smaller than $K\epsilon$, with high probability $(1 - \delta)$. For non-random networks, we establish a tight upper bound on the minimizing $\hat{\tau} p$ and compute the corresponding optimal constant weight $\alpha$. We quantify the tradeoff between the number of iterations $\hat{\tau}$ per Monte-Carlo run and the number of runs $\hat{p}$.

Finally, we compare the bias-variance tradeoffs between the two algorithms and the network parameters that determine their convergence rate and noise resilience. The fixed weight $A - NC$ algorithm can converge faster but requires greater inter-sensor coordination than the $A - ND$ algorithm.

**Comparison with existing literature.** Random link failures and additive channel noise have been considered separately. Random link failures, but noiseless consensus, is in [11], [12], [13], [14], [15], [16]. References [11], [12], [13] assume an erasure model: the network links fail independently in space (independently of each other) and in time (link failure events are temporally independent.) Papers [14], [16] study directed topologies with only time i.i.d. link failures, but impose distributional assumptions on the link formation process. In [15] the link failures are i.i.d. Laplacian matrices, the graph is directed, and no distributional assumptions are made on the Laplacian matrices. The paper presents necessary and sufficient conditions for consensus using the ergodicity of products of stochastic matrices.

Similarly, [17], [18], [19] consider consensus with additive noise, but fixed or static, non random topologies (no link failures.) They use a decreasing weight sequence to guarantee consensus. These references do not characterize the m.s.e. For example, [18], [19] rely on the existence of a unique solution to an algebraic Lyapunov equation. The more general problem of distributed estimation (of which average consensus is a special case) in the presence of additive noise is in [20], again with a fixed topology. Both [17], [20] assume a temporally white noise sequence, while our approach can accommodate a more general Markovian noise sequence, in addition to white noise processes.

In summary, with respect to [11]–[20], our approach considers: i) random topologies and noisy communication links simultaneously; ii) spatially correlated (Markovian) dependent random link failures; iii) time Markovian noise sequences; iv) undirected topologies; v) no distributional assumptions; vi) consensus (estimation being considered elsewhere;) and vii) two versions of consensus representing different compromises of bias versus variance.

Briefly, the paper is as follows. Sections II and III summarize relevant spectral graph and average consensus results. Sections IV and V treat the additive noise with random link failure communication analyzing the $A - ND$ and $A - NC$ algorithms, respectively. Finally, Section VI concludes the paper.
II. ELEMENTARY SPECTRAL GRAPH THEORY

We summarize briefly facts from spectral graph theory. For an undirected graph $G = (V,E)$, $V = [1 \cdots N]$ is the set of nodes or vertices, $|V| = N$, and $E$ is the set of edges, $|E| = M$. The unordered pair $(n,l) \in E$ if there exists an edge between nodes $n$ and $l$. We only consider simple graphs, i.e., graphs devoid of self-loops and multiple edges. A path between nodes $n$ and $l$ of length $m$ is a sequence $(n = i_0, i_1, \cdots, i_m = l)$ of vertices, such that, $(i_k, i_{k+1}) \in E$, $0 \leq k \leq m - 1$. A graph is connected if there exists a path, between each pair of nodes. The neighborhood of node $n$ is

$$\Omega_n = \{l \in V \mid (n,l) \in E\}$$  \hspace{1cm} (1)

Node $n$ has degree $d_n = |\Omega_n|$ (number of edges with $n$ as one end point.) The structure of the graph can be described by the symmetric $N \times N$ adjacency matrix, $A = [A_{nl}]$, $A_{nl} = 1$, if $(n,l) \in E$, 0 otherwise. Let the degree matrix be the diagonal matrix $D = \text{diag}(d_1 \cdots d_N)$. The graph Laplacian matrix, $L$, is

$$L = D - A$$  \hspace{1cm} (2)

The Laplacian is a positive semidefinite matrix; hence, its eigenvalues can be ordered as

$$0 = \lambda_1(L) \leq \lambda_2(L) \leq \cdots \leq \lambda_N(L)$$  \hspace{1cm} (3)

The multiplicity of the zero eigenvalue equals the number of connected components of the network; for a connected graph, $\lambda_2(L) > 0$. This second eigenvalue is the algebraic connectivity or the Fiedler value of the network; see [21], [22], [23] for detailed treatment of graphs and their spectral theory.

III. DISTRIBUTED AVERAGE CONSENSUS WITH IMPERFECT COMMUNICATION

In a simple form, distributed average consensus computes the average $r$ of the initial node data

$$r = \frac{1}{N} \sum_{n=1}^{N} x_n(0)$$  \hspace{1cm} (4)

by local data exchanges among neighbors. For noiseless and unquantized data exchanges across the network links, the state of each node is updated iteratively by

$$x_n(i+1) = w_{nn}(i)x_n(i) + \sum_{l \in \Omega_n(i)} w_{nl}(i)x_l(i), \hspace{0.5cm} 1 \leq n \leq N$$  \hspace{1cm} (5)

where the link weights, $w_{nl}$'s, may be constant or time varying. Similarly, the topology of a time-varying network is captured by making the neighborhoods, $\Omega_n$'s, to be a function of time. Because noise causes consensus to diverge, [24], [10], we let the link weights to be the same across different network links,
but vary with time. Eq. (5) becomes

\[ x_n(i + 1) = [1 - \alpha(i)d_n(i)] x_n(i) + \alpha(i) \sum_{l \in \Omega_n(i)} x_l(i), \quad 1 \leq n \leq N \]  

(6)

We address consensus with imperfect inter-sensor communication, where each sensor receives noise corrupted versions of its neighbors’ states. Eq. (6) is now This leads to the state update given by

\[ x_n(i + 1) = [1 - \alpha(i)d_n(i)] x_n(i) + \alpha(i) \sum_{l \in \Omega_n(i)} f_{nl,i}(x_l(i)), \quad 1 \leq n \leq N \]  

(7)

where \( \{f_{nl,i}\}_{1 \leq n,l \leq N, i \geq 0} \) is a sequence of functions (possibly random) modeling the channel imperfections. In the following sections, we analyze the consensus problem given by eqn. (7), when the channel communication is corrupted by additive noise. In [25], we consider the effects of quantization (see also [26] for a treatment of consensus algorithms with quantized communication.) Here, we study two different algorithms. The first, \( A - \mathcal{ND} \), considers a decreasing weight sequence (\( \alpha(i) \to 0 \)) and is analyzed in Section IV. The second, \( A - \mathcal{NC} \), uses repeated averaging with a constant link weight and is detailed in Section V.

IV. \( A - \mathcal{ND} \): Consensus in Additive Noise and Random Link Failures

We consider distributed consensus when the network links fail or become alive at random times, and data exchanges are corrupted by additive noise. The network topology varies randomly across iterations. We analyze the convergence properties of the \( A - \mathcal{ND} \) algorithm under this generic scenario. We start by formalizing the assumptions underlying \( A - \mathcal{ND} \) in the next Subsection.

A. Problem Formulation and Assumptions

We compute the average of the initial state \( x(0) = [x_1(0) \cdots x_N(0)]^T \in \mathbb{R}^{N \times 1} \) with the distributed consensus algorithm with communication channel imperfections given in eqn. (7). Let \( \{v_{nl}(i)\}_{1 \leq n,l \leq N, i \geq 0} \) be a sequence of independent zero mean random variables. For additive noise,

\[ f_{nl,i}(y) = y + v_{nl}(i) \]  

(8)

Recall the Laplacian \( L \) defined in (2). Collecting the states \( x_n(i) \) in the vector \( x(i) \), eqn. (7) is

\[
\begin{align*}
  x(i + 1) & = x(i) - \alpha(i) [L(i)x(i) + n(i)] \\
  [n(i)]_l & = n_l(i) = - \sum_{k \in \Omega_l(i)} v_{lk}(i), \quad 1 \leq l \leq N, \ i \geq 0
\end{align*}
\]  

(9) (10)
We now state the assumptions of the $A-ND$ algorithm.\footnote{See also [27], where parts of the results are presented.}

1) Random Network Failure: We propose two models; the second is more general than the first.

1.1) Temporally i.i.d. Laplacian Matrices: The graph Laplacians are

$$L(i) = \mathcal{L} + \tilde{L}(i), \ \forall i \geq 0$$

(11)

where $\{L(i)\}_{i \geq 0}$ is a sequence of i.i.d. Laplacian matrices with mean $\mathcal{L} = \mathbb{E}[L(i)]$, such that $\lambda_2(\mathcal{L}) > 0$. We do not make any distributional assumptions on the link failure model, and, in fact, as long as the sequence $\{L(i)\}_{i \geq 0}$ is independent with constant mean $\mathcal{L}$, satisfying $\lambda_2(\mathcal{L}) > 0$, the i.i.d. assumption can be dropped. During the same iteration, the link failures can be spatially dependent, i.e., correlated across different edges of the network. This model subsumes the erasure network model, where the link failures are independent both over space and time. Wireless sensor networks motivate this model since interference among the sensors communication correlates the link failures over space, while over time, it is still reasonable to assume that the channels are memoryless or independent.

Connectedness of the graph is an important issue. We do not require that the random instantiations $G(i)$ of the graph be connected; in fact, it is possible to have all these instantiations to be disconnected. We only require that the graph stays connected on average. This is captured by requiring that $\lambda_2(\mathcal{L}) > 0$, enabling us to capture a broad class of asynchronous communication models; for example, the random asynchronous gossip protocol analyzed in [28] satisfies $\lambda_2(\mathcal{L}) > 0$ and hence falls under this framework.

1.2) Temporally Markovian Laplacian Matrices: Our results hold when the Laplacian matrix sequence $\{L(i, x(i))\}_{i \geq 0}$ is state-dependent. More precisely, we assume that there exists a two-parameter random field, $\{L(i, x)\}_{i \geq 0, x \in \mathbb{R}^N}$ of Laplacian matrices such that

$$\mathbb{E}[L(i, x)] = \mathcal{L}, \ \forall i, x$$

(12)

and $\lambda_2(\mathcal{L}) > 0$. We also require that, for a fixed $i$, the random matrices, $\{L(i, x)\}_{x \in \mathbb{R}^N}$, are independent of the sigma algebra, $\sigma (x(j), 0 \leq j \leq i)$.\footnote{This guarantees that the Laplacian $L(i, x(i))$ may depend on the past state history $\{x(j), j \leq i\}$, only through the present state $x(i)$.} It is clear then that the Laplacian matrix sequence, $\{L(i, x(i))\}_{i \geq 0}$, is Markov. We will show that our convergence analysis holds also for this general link failure model. Such a model may be appropriate in stochastic formation control scenarios, see [29], [30], [31], where the network topology is state-dependent.

2) Communication Noise Model: We propose two models; the second is more general than the first.
2.1) Independent Noise Sequence: The additive noise \( \{v_{nl}(i)\}_{1 \leq n, l \leq N, \ i \geq 0} \) is an independent sequence

\[
E[v_{nl}(i)] = 0, \ \forall 1 \leq n, l \leq N, \ i \geq 0, \ \sup_{n,l,i} E[v_{nl}^2(i)] = \mu < \infty \tag{13}
\]

The sequences, \( \{L(i)\}_{i \geq 0} \) and \( \{v_{nl}(i)\}_{1 \leq n, l \leq N, \ i \geq 0} \) are mutually independent. Hence, \( L(i), \{v_{nl}(i)\}, 1 \leq n, l \leq N, \ i \geq 0 \) are independent of \( \sigma(x(j), 0 \leq j \leq i), \forall i \). Then, from eqn. (10),

\[
E[n(i)] = 0, \ \forall i, \ \sup_i E[\|n(i)\|^2] = \eta \leq N(N-1)\mu < \infty \tag{14}
\]

No distributional assumptions are required on the noise sequence.

2.2) Markovian Noise Sequence: Our approach allows the noise sequence to be Markovian through state-dependence. Let the two-parameter random field, \( \{n(i, x)\}_{i \geq 0, x \in \mathbb{R}^{N \times 1}} \) of random vectors

\[
E[n(i)] = 0, \ \forall i, x \tag{15}
\]

For fixed \( i \), the random vectors, \( \{n(i, x)\}_{x \in \mathbb{R}^{N \times 1}} \), are independent of the \( \sigma \)-algebra, \( \sigma(x(j), 0 \leq j \leq i) \) and the random families \( \{L(i, x)\}_{x \in \mathbb{R}^{N \times 1}} \) and \( \{n(i, x)\}_{x \in \mathbb{R}^{N \times 1}} \) are independent. It is clear then that the noise vector sequence, \( \{n(i, x(i))\}_{i \geq 0} \), is Markov. Note, however, in this case the resulting Laplacian and noise sequences, \( \{L(i, x(i))\}_{i \geq 0} \) and \( \{n(i, x(i))\}_{i \geq 0} \) are no longer independent; they are coupled through the state \( x(i) \). In addition to (15), we require the variance of the noise component orthogonal to the consensus subspace (see eqn. (31)) to satisfy, for constants, \( c_1, c_2 \geq 0 \),

\[
E[\|n_{C}^\perp(i, x)\|^2] \leq c_1 + c_2 \|x_{C}^\perp\|^2 \tag{16}
\]

We do not restrict the variance growth rate of the noise component in the consensus subspace. This clearly subsumes the bounded noise variance model. An example of such noise is

\[
n(i, x(i)) = \vartheta(i) (x(i) + w(i)) \tag{17}
\]

where \( \{\vartheta(i)\}_{i \geq 0} \) and \( \{w(i)\}_{i \geq 0} \) are zero mean finite variance mutually i.i.d. sequences of scalars and vectors, respectively. It is then clear that the condition in eqn. (16) is satisfied, and the noise model 2.2) applies. The model in eqn. (17) arises, for example, in multipath effects in MIMO systems, when the channel adds multiplicative noise whose amplitude is proportional to the transmitted data.

3) Persistence Condition: The weights decay to zero, but not too fast

\[
\alpha(i) > 0, \ \sum_{i \geq 0} \alpha(i) = \infty, \ \sum_{i \geq 0} \alpha^2(i) < \infty \tag{18}
\]
This condition is commonly assumed in adaptive control and signal processing. Examples include

\[ \alpha(i) = \frac{1}{i^\beta}, \quad 0.5 < \beta \leq 1 \quad (19) \]

For clarity, in the main body of the paper, we prove the results for the \( A - ND \) algorithm under Assumptions 1.1), 2.1), and 3). In the Appendix, we point out how to modify the proofs when the more general assumptions 1.2) and 2.2) hold.

We now prove the almost sure (a.s.) convergence of the \( A - ND \) algorithm in eqn.(9) by using results from the theory of stochastic approximation algorithms, [32].

### B. A Result on Convergence of Markov Processes

A systematic and thorough treatment of stochastic approximation procedures has been given in [32]. In this section, we modify slightly a result from [32] and restate it as a theorem in a form relevant to our application. We follow the notation of [32], which we now introduce.

Let \( X = \{x(i)\}_{i \geq 0} \) be a Markov process on \( \mathbb{R}^{N \times 1} \). The generating operator \( L \) of \( X \) is

\[ L V(i, x) = \mathbb{E}[V(i + 1, x(i + 1)) | x(i) = x] - V(i, x) \quad (20) \]

for functions \( V(i, x), \quad i \geq 0, \quad x \in \mathbb{R}^{N \times 1} \), provided the conditional expectation exists. We say that \( V(i, x) \in L \) in the domain \( A \), if \( V(i, x) \) is finite for all \( (i, x) \in A \).

Denote the Euclidean metric by \( \rho(\cdot) \). For \( B \subset \mathbb{R}^{N \times 1} \), the \( \epsilon \)-neighborhood of \( B \) and its complement is

\[ U_\epsilon(B) = \left\{ x \mid \inf_{y \in B} \rho(x, y) < \epsilon \right\} \quad (21) \]
\[ V_\epsilon(B) = \mathbb{R}^{N \times 1} \setminus U_\epsilon(B) \quad (22) \]

We now state the desired theorem, whose proof we sketch in the Appendix.

**Theorem 1** Let \( X \) be a Markov process with generating operator \( L \). Let there exist a non-negative function \( V(i, x) \in L \) in the domain \( i \geq 0, \quad x \in \mathbb{R}^{N \times 1} \), and \( B \subset \mathbb{R}^{N \times 1} \) with the following properties:

1) \[ \inf_{i \geq 0, x \in V_\epsilon(B)} V(i, x) > 0, \quad \forall \epsilon > 0 \quad (23) \]
\[ V(i, x) \equiv 0, \quad x \in B \quad (24) \]
\[ \lim_{x \to B} \sup_{i \geq 0} V(i, x) = 0 \quad (25) \]
2) \[ L V(i, x) \leq g(i)(1 + V(i, x)) - \alpha(i)\phi(i, x) \quad (26) \]
where $\varphi(i, x), i \geq 0, x \in \mathbb{R}^{N \times 1}$ is a non-negative function such that

$$\inf_{i, x \in V_\epsilon(B)} \varphi(i, x) > 0, \quad \forall \epsilon > 0 \quad (27)$$

$$\alpha(i) > 0, \quad \sum_{i \geq 0} \alpha(i) = \infty \quad (28)$$

$$g(i) > 0, \quad \sum_{i \geq 0} g(i) < \infty \quad (29)$$

Then, the Markov process $X = \{x(i)\}_{i \geq 0}$ with arbitrary initial distribution converges a.s. to $B$ as $i \to \infty$. In other words,

$$P\left(\lim_{i \to \infty} \rho(x(i), B) = 0\right) = 1 \quad (30)$$

\textbf{C. Proof of Convergence of the $A - ND$ Algorithm}

The $A - ND$ distributed consensus algorithm is given by eqn. (9) in Section IV-A. To establish its a.s. convergence using Theorem 1, define the \textit{consensus} subspace, $C$, aligned with $1$, the vectors of $1$'s,

$$C = \{x \in \mathbb{R}^{N \times 1} | x = a1, a \in \mathbb{R}\} \quad (31)$$

We recall a result on distance properties in $\mathbb{R}^{N \times 1}$ to be used in the sequel. We omit the proof.

\textbf{Lemma 2} Let $S$ be a subspace of $\mathbb{R}^{N \times 1}$. For $x \in \mathbb{R}^{N \times 1}$, consider the orthogonal decomposition $x = x_S + x_{S^\perp}$. Then $\rho(x, S) = \|x_{S^\perp}\|$. 

\textbf{Theorem 3 (A – ND a.s. convergence)} Let assumptions 1.1), 2.1), and 3) hold. Consider the $A - ND$ consensus algorithm in eqn. (9) in Section IV-A with initial state $x(0) \in \mathbb{R}^{N \times 1}$. Then,

$$P\left[\lim_{i \to \infty} \rho(x(i), C) = 0\right] = 1 \quad (32)$$

\textbf{Proof:} Under the assumptions, the process $X = \{x(i)\}_{i \geq 0}$ is Markov. Define

$$V(i, x) = x^T L x \quad (33)$$

The potential function $V(i, x)$ is non-negative. Since $x \in C$ is an eigenvector of $L$ with zero eigenvalue,

$$V(i, x) \equiv 0, \quad x \in C, \quad \lim_{x \to C} \sup_{i \geq 0} V(i, x) = 0 \quad (34)$$
The second condition follows from the continuity of $V(i, x)$. By Lemma 2 and the definition in eqn. (22) of the complement of the $\epsilon$-neighborhood of a set

$$x \in V_\epsilon(C) \implies \|x_{\not C}^+\| \geq \epsilon$$

(35)

Hence, for $x \in V_\epsilon(C)$,

$$V(i, x) = x^T L x$$

$$\geq \lambda_2(L) \|x_{\not C}^+\|^2$$

$$\geq \lambda_2(L) \epsilon^2$$

(36)

Then, since by assumption 1.1) $\lambda_2(L) > 0$ (note that the assumption $\lambda_2(L) > 0$ comes into play here), we get

$$\inf_{i \geq 0, x \in V_\epsilon(C)} V(i, x) \geq \lambda_2(L) \epsilon^2 > 0$$

(37)

Now consider $LV(i, x)$ in eqn. (20). Using eqn. (9) in (20), we obtain

$$LV(i, x) = E \left[ x(i+1)^T L x(i+1) \mid x(i) = x \right] - x^T L x$$

$$= E \left[ x - \alpha(i) L x - \alpha(i) \bar{L}(i) x - \alpha(i) n(i) \right]^T L \left[ x - \alpha(i) L x - \alpha(i) \bar{L}(i) x - \alpha(i) n(i) \right]$$

$$- x^T L x$$

(38)

Using the independence of $\bar{L}(i)$ and $n(i)$ with respect to $x(i)$, that $\bar{L}(i)$ and $n(i)$ are zero-mean, and that the subspace $C$ lies in the null space of $L^3$ and $\bar{L}(i)$ (the latter because this is true for both $L(i)$ and $L$), eqn. (38) leads successively to

$$LV(i, x) = -2\alpha(i)x^T L^2 x + \alpha^2(i)x^T L^3 x + E \left[ \alpha^2(i) (\bar{L}(i)x)^T L (\bar{L}(i)x) \right] + E \left[ \alpha^2(i)n(i)^T L n(i) \right]$$

$$\leq -2\alpha(i)x^T L^2 x + \alpha^2(i)\lambda_N(L)^3\|x_{\not C}^+\|^2 + \alpha^2(i)\lambda_N(L)E \left[ \|\bar{L}(i)x\|^2 \right]$$

$$+ \alpha^2(i)\lambda_N(L)E \left[ \|n(i)\|^2 \right]$$

$$\leq -2\alpha(i)x^T L^2 x + \alpha^2(i)\lambda_N(L)^3\|x_{\not C}^+\|^2 + \alpha^2(i)\lambda_N(L)E \left[ \lambda_{\max}^2(\bar{L}(i)) \right] \|x_{\not C}^+\|^2$$

$$+ \alpha^2(i)\lambda_N(L)\eta$$

$$\leq -2\alpha(i)x^T L^2 x + \alpha^2(i)\lambda_N(L)^3\|x_{\not C}^+\|^2 + 4\alpha^2(i)N^2\lambda_N(L)\|x_{\not C}^+\|^2 + \alpha^2(i)\lambda_N(L)\eta$$

(39)
The last step follows because all the eigenvalues of $\tilde{L}(i)$ are less than $2N$ in absolute value, by the Gershgorin circle theorem. Now, by the fact $x^T L x \geq \lambda_2 (L) \| x \|_2^2$ and $\lambda_2 (L) > 0$, we have

$$LV(i, x) \leq -2\alpha(i)x^T L x + \alpha^2(i) \left[ \lambda_N(L) \eta + \frac{\lambda_3(L)}{\lambda_2(L)} x^T L x + \frac{4N^2 \lambda_N(L)}{\lambda_2(L)} x^T L x \right]$$

where

$$\varphi(i, x) = 2x^T L^2 x, \quad g(i) = \alpha^2(i) \max \left( \lambda_N(L) \eta, \frac{\lambda_3(L)}{\lambda_2(L)} \right)$$

It is easy to see that $\varphi(i, x)$ and $g(i)$ defined above satisfy the conditions for Theorem 1. Hence,

$$\mathbb{P} \left[ \lim_{i \to \infty} \rho(x(i), C) = 0 \right] = 1 \quad (42)$$

Theorem 3 assumes 1.1), 2.1), 3). For an equivalent statement under 1.2), 2.2), see Theorem 11 in the Appendix.

Theorem 3 shows that the sample paths approach the consensus subspace, $C$, with probability 1 as $i \to \infty$. We now show that the sample paths, in fact, converge a.s. to a finite point in $C$.

**Theorem 4 (a.s. consensus: Limiting random variable)** Let assumptions 1.1), 2.1), and 3) hold. Consider the $A - ND$ consensus algorithm in eqn. (9) in Section IV-A with initial state $x(0) \in \mathbb{R}^{N \times 1}$. Then, there exists an almost sure finite real random variable $\theta$ such that

$$\mathbb{P} \left[ \lim_{i \to \infty} x(i) = \theta 1 \right] = 1 \quad (43)$$

**Proof:** Denote the average of $x(i)$ by

$$x_{avg}(i) = \frac{1}{N} 1^T x(i) \quad (44)$$

The conclusions of Theorem 3 imply that

$$\mathbb{P} \left[ \lim_{i \to \infty} \| x(i) - x_{avg}(i) 1 \| = 0 \right] = 1 \quad (45)$$

Recall the distributed average consensus algorithm in eqn. (9). Premultiplying both sides of eqn. (9) by $\frac{1}{N} 1^T$, we get the stochastic difference equation for the averages as

$$x_{avg}(i + 1) = x_{avg}(i) - \xi(i)$$

$$= x_{avg}(0) - \sum_{0 \leq j \leq i} \xi(j) \quad (46)$$
where
\[ \xi(i) = \frac{\alpha(i)}{N} \mathbf{1}^T \mathbf{n}(i) \]

Given eqn. (14), in particular, the sequence \( \{\mathbf{n}(i)\} \) is time independent, it follows that
\[
\begin{align*}
\mathbb{E} [\xi(i)] &= 0, \quad \forall i, \\
\sum_{i \geq 0} \mathbb{E}[\xi(i)]^2 &= \sum_{i \geq 0} \frac{\alpha^2(i)}{N^2} \mathbb{E} [||\mathbf{n}(i)||^2] \\
&\leq \frac{\eta}{N^2} \sum_{i \geq 0} \alpha^2(i) \\
&< \infty
\end{align*}
\]

which implies
\[
\mathbb{E} \left[ (x_{\text{avg}}(i))^2 \right] \leq x_{\text{avg}}^2(0) + \frac{\eta}{N^2} \sum_{j \geq 0} \alpha^2(j), \quad \forall i
\]

Thus, the sequence \( \{x_{\text{avg}}(i)\}_{i \geq 0} \) is an \( L_2 \) bounded martingale and, hence, converges a.s. and in \( L_2 \) to a finite random variable \( \theta \) (see, [33].) The theorem then follows from eqn. (45).

Again, we note that we obtain an equivalent statement under Assumptions 1.2), 2.1) in Theorem 12 in the Appendix. Proving under Assumption 2.2) requires more specific information about the mixing properties of the Markovian noise sequence, which due to space limitations is not addressed here.

D. Mean Square Error

By Theorem 4, the sensors reach consensus asymptotically and converge a.s. to a finite random variable \( \theta \). Viewed as an estimate of the initial average \( r \) (see eqn. (4)), \( \theta \) should possess desirable properties like unbiasedness and small m.s.e. The next Lemma characterizes the desirable statistical properties of \( \theta \).

**Lemma 5**  Let \( \theta \) be as in Theorem 4 and \( r \) as in eqn. (4). Define the m.s.e. \( \zeta \) as
\[
\zeta = \mathbb{E} \left[ (\theta - r)^2 \right]
\]

Then, the consensus limit \( \theta \) is unbiased and its m.s.e. bounded as below.
\[
\begin{align*}
\mathbb{E} [\theta] &= r \quad (50) \\
\zeta &\leq \frac{\eta}{N^2} \sum_{i \geq 0} \alpha^2(i) \quad (51)
\end{align*}
\]

**Proof:** From eqn. (46), we have
\[
\mathbb{E} [x_{\text{avg}}(i)] = r, \quad \forall i
\]
Since \( \{x_{avg}(i)\}_{i \geq 0} \) converges in \( L_2 \) to \( \theta \) (Theorem 4), it also converges in \( L_1 \), and we have
\[
E[\theta] = \lim_{i \to \infty} E[x_{avg}(i)] = r
\]
which proves (50). For (51), by Theorem 4, the sequence \( \{(x_{avg}(i))^2\}_{i \geq 0} \) converges in \( L_2 \) to \( (\theta - r)^2 \).
\[
\zeta = E[(\theta - r)^2] = \lim_{i \to \infty} E[(x_{avg}(i) - r)^2] = \sum_{i \geq 0} \frac{\alpha^2(i)}{N^2} E[\|n(i)\|^2]
\]
Eqn. (51) follows from (54) and (48), with \( \eta \) the bound on the noise variance.

Lemma 13 in the Appendix shows equivalent results under Assumptions 1.2), 2.1). Proving under Assumption 2.2) requires more specific information about the mixing properties of the Markovian noise sequence, which we do not pursue here.

Eqn. 54 gives the exact representation of the m.s.e. As a byproduct, we obtain the following corollary for an erasure network with identical link failure probabilities and i.i.d. channel noise.

**Corollary 6** Consider an erasure network with \( M \) realizable links, identical link failure probability, \( p \), for each link, and the noise sequence, \( \{v_{nl}(i)\}_{1 \leq n, l \leq N, i \geq 0} \), be of identical variance \( \sigma^2 \). Then the m.s.e. is
\[
\zeta = \frac{2M\sigma^2(1 - p)}{N^2} \sum_{j \geq 0} \alpha^2(j)
\]
**Proof:** Using the fact that the link failures and the channel noise are independent, we have
\[
E[\|n(i)\|^2] = 2\sigma^2 M(1 - p)
\]
The result then follows from eqn. (54).

While interpreting the dependence of eqn. (55) on the number of nodes, \( N \), it should be noted that the number of realizable links \( M \) must be \( \Omega(N) \) for the connectivity assumptions to hold.

Lemma 5 shows that, for a given bound on the noise variance, \( \eta \), the m.s.e. \( \zeta \) can be made arbitrarily small by properly scaling the weight sequence, \( \{\alpha(j)\}_{j \geq 0} \). As an example, consider the weight sequence,
\[
\alpha(j) = \frac{1}{j + 1}, \ \forall j
\]
Clearly, this choice of \( \alpha(i) \) satisfies the persistence conditions of eqn. (28) and, in fact,
\[
\sum_{j \geq 0} \alpha^2(j) = \sum_{j \geq 1} \frac{1}{j^2} = \frac{\pi^2}{6}
\]
Then, for any $\epsilon > 0$, the scaled weight sequence, $\{\tilde{\alpha}(j)\}_{j \geq 0}$,

$$\tilde{\alpha}(j) = \frac{\sqrt{6\epsilon} N}{\sqrt{\eta\pi(j + 1)}}$$

will guarantee that $\zeta \leq \epsilon$. However, reducing the m.s.e. by scaling the weights in this way will reduce the convergence rate of the algorithm; this trade-off is considered in the next Subsection.

### E. Convergence Rate

The $A - ND$ algorithm falls under the framework of stochastic approximation and, hence, a detailed convergence rate analysis can be done through the ODE method (see, for example, [34].) For clarity of presentation, we skip this detailed analysis here; rather, we present a simpler convergence rate analysis, involving the mean state vector sequence only under assumptions 1.1), 2.1), and 3). From the asymptotic unbiasedness of $\theta$, it follows

$$\lim_{i \to \infty} \mathbb{E}[x(i)] = r1$$

Our goal is to determine the rate at which the sequence $\{\mathbb{E}[x(i)]\}_{i \geq 0}$ converges to $r1$. Since $L(i)$ and $x(i)$ are independent, and $n(i)$ is zero mean, we have

$$\mathbb{E}[x(i + 1)] = (I - \alpha(i)\overline{L}) \mathbb{E}[x(i)], \forall i$$

By the persistence condition (18) the sequence $\alpha(i) \to 0$. Without loss of generality, we can assume that

$$\alpha(i) \leq \frac{2}{\lambda_2(\overline{L}) + \lambda_N(\overline{L})}, \forall i$$

Then, it can be shown that (see [11])

$$\|\mathbb{E}[x(i)] - r1\| \leq \left(\prod_{0 \leq j \leq i-1} \left(1 - \alpha(j)\lambda_2(\overline{L})\right)\right) \|\mathbb{E}[x(0)] - r1\|$$

Now, since $1 - a \leq e^{-a}, 0 \leq a \leq 1$, we have

$$\|\mathbb{E}[x(i)] - r1\| \leq \left(e^{-\lambda_2(\overline{L})(\sum_{0 \leq j \leq i-1} \alpha(j))}\right) \|\mathbb{E}[x(0)] - r1\|$$

Eqn. (61) shows that the rate of convergence of the mean consensus depends on the topology through the algebraic connectivity $\lambda_2(\overline{L})$ of the mean graph and through the weights $\alpha(i)$. It is interesting to note that the way the random link failures affect the convergence rate (at least for the mean state sequence) is through $\lambda_2(\overline{L})$, the algebraic connectivity of the mean Laplacian, in the exponent, whereas, for a static network, this reduces to the algebraic connectivity of the static Laplacian $L$, recovering the results in [17].

Eqns. (61) and (51) show a tradeoff between the m.s.e. and the rate of convergence at which the
sequence \( \{ \mathbb{E} [\mathbf{x}(i)] \}_{i \geq 0} \) converges to \( r \mathbf{1} \). Eqn. (61) shows that this rate of convergence is closely related to the rate at which the weight sequence, \( \alpha(i) \), sums to infinity. For a faster rate, we want the weights to sum up fast to infinity, i.e., the weights to be large. In contrast, eqns. (51) shows that, to achieve a small \( \zeta \), the weights should be small.

We studied the trade-off between convergence rate and m.s.e. of the mean state vectors only. In general, more effective measures of convergence rate are appropriate; intuitively, the same trade-offs will be exhibited, in the sense that the rate of convergence will be closely related to the rate at which the weight sequence, \( \alpha(i) \), sums to infinity, as verified by the numerical studies presented next.

**F. Numerical Studies - \( \mathcal{A} - \mathcal{N} \mathcal{D} \)**

We present numerical studies on the \( \mathcal{A} - \mathcal{N} \mathcal{D} \) algorithm that verify the analytical results. The first set of simulations confirms the a.s. consensus in Theorems 3 and 4. Consider an erasure network on \( N = 100 \) nodes and \( M = 5N \) realizable links, with identical probability of link failure \( p = .4 \) and identical channel noise variance \( \sigma^2 = 15 \). We take \( \alpha(i) = 1/4i \) and plot on Fig. 1 on the left, the sample paths \( \{ x_n(i) \}_{1 \leq n \leq N} \), of the sensors over an instantiation of the \( \mathcal{A} - \mathcal{N} \mathcal{D} \) algorithm. We note that the sensor states converge to consensus, thus verifying our analytical results.

The second set of simulations confirms the m.s.e. in Corollary 6. We consider the same erasure network, but take \( \sigma^2 = 30 \) and \( \alpha(i) = 1/5i \). We simulate 50 runs of the \( \mathcal{A} - \mathcal{N} \mathcal{D} \) algorithm from the initial state. Fig. 1 on the center plots the propagation of the squared error \( (x_n(i) - r)^2 \) for a randomly chosen sensor \( n \) for each of the 50 runs. The cloud of (blue) lines denotes the 50 runs, whereas the extended dashed (red) line denotes the exact m.s.e. computed in Corollary 6. The paths are clustered around the exact m.s.e., thus verifying our results.

The third set of simulations studies the trade-off between m.s.e. and convergence rate. We consider the same erasure network, but take \( \sigma^2 = 50 \), and run the \( \mathcal{A} - \mathcal{N} \mathcal{D} \) algorithm from the same initial conditions, but for the weight sequences \( \{ \alpha_s(i) = s/i \}_{i \geq 0} \), \( s = .33, .1 \). Fig. 1 on the right depicts the propagation of the squared error averaged over all the sensors, \( (1/N) \sum_{n=1}^{N} (x_n(i) - r)^2 \), for each case. We see that the solid (blue) line (\( s = .33 \)) decays much faster initially than the dotted (red) line (\( s = .1 \)) and reaches a steady state. The dotted line ultimately crosses the solid line and continues to decay at a very slow rate, thus verifying the m.s.e. versus convergence rate trade-off, which we established rigorously by restricting attention to the mean state only.

**V. \( \mathcal{A} - \mathcal{N} \mathcal{C} \): Consensus with Repeated Averaging**

The stochastic approximation approach to the average consensus problem, discussed in Section IV, achieves arbitrarily small m.s.e., see eqn. (51), possibly at the cost of a lower convergence rate, especially,
when the desired m.s.e. is small. This is mainly because the weights $\alpha(i)$’s decrease to zero, slowing the convergence as time progresses, as discussed in Subsection IV-E. In this Section, we consider an alternative approach based on repeated averaging, which removes this difficulty. We use a constant link weight (or step size) and run the consensus iterations for a fixed number, $\hat{\tau}$, of iterations. This procedure is repeated $\hat{\mu}$ times, each time starting from the same initial state $x(0)$. Since the $\hat{\mu}$ final states obtained at iteration $\hat{\tau}$ of each of the $\hat{\mu}$ runs are independent, we average them and get the law of large numbers to work for us. There is an interesting tradeoff between $\hat{\tau}$ and $\hat{\mu}$ for a constant total number of iterations $\hat{\tau}\hat{\mu}$. We describe and analyze this algorithm and consider its tradeoffs next. The Section is organized as follows. Subsection V-A sets up the problem, states the assumptions, and gives the algorithm $\mathcal{A} - \mathcal{NC}$ for distributed average consensus with noisy communication links. We analyze the performance of the $\mathcal{A} - \mathcal{NC}$ algorithm in Subsection V-B. In Subsection V-C, we present numerical studies and suggest generalizations in Subsection V-D.

A. $\mathcal{A} - \mathcal{NC}$: Problem Formulation and Assumptions

Again, we consider distributed consensus with communication channel imperfections in eqn. (7) to average the initial state, $x(0) \in \mathbb{R}^{N \times 1}$. The setup is the same as in eqns. (8) to (10).

$\mathcal{A} - \mathcal{NC}$ Algorithm: The $\mathcal{A} - \mathcal{NC}$ algorithm is the following Monte Carlo (MC) averaging procedure:

$$x^p(i + 1) = x^p(i) - \alpha (L x^p(i) + n^p(i)) , \ 0 \leq i \leq \hat{\tau} - 1, \ 1 \leq p \leq \hat{\mu}, \ x^p(0) = x(0)$$ (62)

where $x^p_n(i)$ is the state at sensor $n$ at the $i$-th iteration of the $p$-th MC run. In particular, $x^p_n(\hat{\tau})$ is the state at sensor $n$ at the end of the $p$-th MC run. Each run of the $\mathcal{A} - \mathcal{NC}$ algorithm proceeds for $\hat{\tau}$ iterations.

Fig. 1. Left: Sensors sample paths, $\{x_n(i)\}_{1 \leq n \leq N}$, of the $\mathcal{A} - \mathcal{ND}$ algorithm, verifying a.s. consensus. Center: Sample paths of the $\mathcal{A} - \mathcal{ND}$ algorithm, verifying the m.s.e. bound. Right: m.s.e versus convergence rate tradeoff.
and there are \( p \) MC runs. Finally, the estimate, \( \tilde{x}_n(\bar{\tau}) \), of the average, \( x_{\text{avg}}(0) \), at sensor \( n \) is

\[
\tilde{x}_n(\bar{\tau}) = \frac{1}{p} \sum_{p=1}^{p} x_n^p(\bar{\tau})
\]

We analyze the \( A - NC \) algorithm under the following assumptions. These make the analysis tractable, but, are not necessary. They can be substantially relaxed, as shown in Subsection V-D.

1) **Static Network**: The Laplacian \( L \) is fixed (deterministic,) and the network is connected, \( \lambda_2(L) > 0 \). (In Subsection V-D we will allow random link failures.)

2) **Independent Gaussian Noise Sequence**: The additive noise \( \{v_{nl}^p(i)\}_{1 \leq n,l \leq N, i,p \geq 0} \) is an independent Gaussian sequence with

\[
E[v_{nl}^p(i)] = 0, \quad \forall 1 \leq n, l \leq N, \quad i, p \geq 0, \quad \sup_{n,l,i,p} E[(v_{nl}^p(i))^2] = \mu < \infty
\]

From eqn. (10), it then follows that

\[
E[n^p(i)] = 0, \quad \forall i,p \sup_{i,l,p} E||n^p(i)||^2 = \phi^2_{\text{max}} \leq (N - 1) \mu < \infty
\]

3) **Constant link weight**: The link weight \( \alpha \) is constant across iterations and satisfies

\[
0 < \alpha < \frac{2}{\lambda_N(L)}
\]

Let \( r = \frac{1}{N}1^T x(0) \) be the initial average. To define a uniform convergence metric, assume that the initial sensor observations \( x(0) \) belong to the following set (for some \( K \in [0, \infty) \)):

\[
\mathcal{K} = \{ x(0) \in \mathbb{R}^{N \times 1} | \|x(0) - r1\| \leq K \}
\]

As performance metric for the \( A - NC \) approach, we adopt the \( \epsilon - \delta \) averaging time, \( T^\alpha(\epsilon, \delta) \), given by

\[
T^\alpha(\epsilon, \delta) = i^* p^*
\]

where

\[
(i^*, p^*) = \arg \inf_{i,p} \left\{ (i,p) \left| \inf_{x(0)\in\mathcal{K}} \inf_{n} \Pr\left[ \frac{|\tilde{x}_n(\bar{\tau}) - r|}{K} \leq \epsilon \right] \geq 1 - \delta \right. \right\}
\]

and the superscript \( \alpha \) denotes explicitly the dependence on the link weight \( \alpha \).

We say that the \( A - NC \) algorithm achieves \( (\epsilon, \delta) \)-consensus if \( T^\alpha(\epsilon, \delta) \) is finite. A similar notion of averaging time has been used by others, see for example, [35], [36]. The next Subsection upper bounds the averaging time and analyzes the performance of \( A - NC \).
B. Performance Analysis of $A - NC$

The $A - NC$ iterations can be rewritten as

$$x^p(i + 1) = Wx^p(i) + \chi^p(i), \ 0 \leq i \leq \hat{i} - 1, \ 1 \leq p \leq \hat{p}, \ x^p(0) = x(0) \tag{70}$$

where

$$W = I - \alpha L, \ \chi^p(i) = -\alpha n^p(i), \ 0 \leq i \leq \hat{i} - 1, \ 1 \leq p \leq \hat{p} \tag{71}$$

Also, for the choice of $\alpha$ given in eqn. (66), the following can be shown (see [7]) for the spectral norm

$$\gamma_2 = \rho(W - \frac{1}{N}J) = \rho(I - \alpha L - \frac{1}{N}J) < 1, \ J = 11^T \tag{72}$$

where $\rho(\cdot)$ is the spectral radius, which is equal to the induced matrix 2 norm for symmetric matrices.

We next develop an upper bound on the averaging time $T^\alpha(\epsilon, \delta)$ given in (68). Actually, we derive a general bound that holds for generic weight matrices $W$. This we do next, in Theorem 7. We come back in Theorem 10 to bounding the averaging time (68) for the model (70) when the weight matrix $W$ is as in (71) and the spectral norm is given by (72).

**Theorem 7 (Averaging Time)** Consider the distributed iterative procedure (70). The weight matrix $W$ is generic, i.e., is not necessarily of the form (71). It does satisfy the following assumptions:

1) **Symmetric Weights:**

$$W = W^T, \ W1 = 1, \ \gamma_2 = \rho(W - \frac{1}{N}J) < 1 \tag{73}$$

2) **Noise Assumptions:** The sequence $\{\chi^p(i)\}_{i \geq 0, p \geq 1}$ is a sequence of independent Gaussian noise vectors with uncorrelated components, such that

$$E[\chi^p(i)] = 0, \ \sup_{i \geq 0, p \geq 1} \sup_{1 \leq l \leq N} E[(\chi^p_l(i))^2] \leq \phi_{\text{max}}^2 < \infty \tag{74}$$

Then, we have the following upper bound on the averaging time $T^\gamma_2(\epsilon, \delta)$ given by eqn. (68):

$$T^\gamma_2(\epsilon, \delta) \leq \tilde{T}^\gamma_2(\epsilon, \delta) \tag{75}$$

$$\tilde{T}^\gamma_2(\epsilon, \delta) = \left(\frac{\ln \epsilon/2}{\ln \gamma_2} + 1\right) \left[\left(\frac{4\phi_{\text{max}}^2 \ln(2/\delta)}{K\epsilon}\right) \left(\frac{\ln \epsilon/2}{N \ln \gamma_2} + 1 + \frac{1 - \gamma_2^2 \epsilon^2/4}{1 - \gamma_2^2} \left(1 - \frac{1}{N}\right)\right)\right] + 1 \tag{76}$$

(Note we replace the superscript $\alpha$ by $\gamma_2$ because we prove results here for arbitrary $W$ satisfying eqn. (73), not necessarily of the form $I - \alpha L$.)

For the proof we need a result from [10], which we state as a Lemma.
Lemma 8  Let the assumptions in Theorem 7 hold true. Define

\[ m_l(\hat{i}) = \mathbb{E} \left[ x^p_l(\hat{i}) \right], \quad v_l(\hat{i}) = \mathbb{E} \left[ (x^p_l(\hat{i}) - m_l(\hat{i}))^2 \right], \quad 1 \leq p \leq \hat{p}, \quad 1 \leq l \leq N \] (77)

(Note that these quantities do not depend on \( p \).) Then we have the following:

1) Bound on error mean:

\[ |m_l(\hat{i}) - r| \leq \gamma_2^2 \| \mathbf{x}(0) - r \mathbf{1} \| \leq \gamma_2^2 K, \quad \forall \mathbf{x}(0) \in \mathcal{K}, \quad 1 \leq l \leq N \] (78)

2) Bound on error variance:

\[ v_l(\hat{i}) \leq \phi^2_{\text{max}} \left[ \frac{\hat{\tau}}{N} + \frac{1 - \gamma_2^2}{1 - \gamma_2^2} \left( 1 - \frac{1}{N} \right) \right], \quad \forall \mathbf{x}(0) \in \mathbb{R}^{N \times 1}, \quad 1 \leq l \leq N \] (79)

3) \( \{x^p_l(\hat{i})\}^{\hat{p}}_{p=1} \) is an i.i.d. sequence, with

\[ x^p_l(\hat{i}) \overset{\text{d}}{=} \mathcal{N}(m_l(\hat{i}), v_l(\hat{i})), \quad 1 \leq p \leq \hat{p} \] (80)

Proof: For proof, see [10].

We now return to the proof of Theorem 7.

Proof: [Theorem 7] The estimate, \( \hat{x}^p_l(\hat{i}) \), of the average at each sensor after \( \hat{p} \) runs is in eqn. (63). From Lemma 8 and a standard Chernoff type bound for Gaussian random variables (see [37]), then

\[ \mathbb{P} \left( \left| \frac{\hat{x}^p_l(\hat{i}) - m_l(\hat{i})}{K} \right| \leq \epsilon \right) \geq 1 - 2 \exp \left\{ -\frac{\hat{p}K\epsilon}{2v_l(\hat{i})} \right\} \] (81)

(For the present derivation we assume Gaussian noise; however, the analysis for other noise models may be done by using the corresponding large deviation rate functions.)

For arbitrary \( \epsilon > 0 \), define

\[ \hat{\gamma}(\epsilon) = \left\lceil \frac{\ln \epsilon}{\ln \gamma_2} \right\rceil \] (82)

Also, for arbitrary \( \epsilon, \delta > 0 \), define

\[ \hat{\rho}(\epsilon, \delta) = \left\lceil \frac{2v_l(\hat{i})}{K \epsilon \ln \frac{2}{\delta}} \right\rceil \] (83)

Then, we have from eqn. (78) in Lemma 8,

\[ \left| \frac{m_l(\hat{i}) - r}{K} \right| \leq \epsilon/2, \quad \forall \hat{\gamma} \geq \hat{\gamma}(\epsilon/2), \quad 1 \leq l \leq N \] (84)

Also, we have from eqn. (81),

\[ \mathbb{P} \left( \left| \frac{\hat{x}^p_l(\hat{i}) - m_l(\hat{i})}{K} \right| \leq \epsilon/2 \right) \geq 1 - \delta, \quad \forall \hat{\rho} \geq \hat{\rho}(\epsilon/2, \delta), \quad 1 \leq l \leq N \] (85)
From the triangle inequality, we get
\[ |\mathbf{x}_l^p(\hat{\mathbf{i}}) - r| \leq |\mathbf{x}_l^p(\hat{\mathbf{i}}) - m_l(\hat{\mathbf{i}})| + |m_l(\hat{\mathbf{i}}) - r|, \quad \forall 1 \leq l \leq N \tag{86} \]

It then follows from eqn. (84) that
\[ |\mathbf{x}_l^p(\hat{\mathbf{i}}) - m_l(\hat{\mathbf{i}})| \leq K\epsilon/2 \Rightarrow |\mathbf{x}_l^p(\hat{\mathbf{i}}) - r| \leq K\epsilon, \quad \forall \hat{\mathbf{i}} \geq \hat{\mathbf{i}}(\epsilon/2), \quad 1 \leq l \leq N \tag{87} \]

We thus have for \( \hat{\mathbf{i}} \geq \hat{\mathbf{i}}(\epsilon/2) \) and \( \hat{\mathbf{p}} \geq \hat{\mathbf{p}}(\epsilon/2, \delta) \)
\[ P \left( \frac{|\mathbf{x}_l^p(\hat{\mathbf{i}}) - r|}{K} \leq \epsilon \right) \geq P \left( \frac{|\mathbf{x}_l^p(\hat{\mathbf{i}}) - m_l(\hat{\mathbf{i}})|}{K} \leq \epsilon/2 \right) \geq 1 - \delta, \quad 1 \leq l \leq N \tag{88} \]

where the last inequality follows from eqn. (85).

From the definition of \( T^{\gamma_2}(\epsilon, \delta) \), see eqn. (68), we then have
\[ T^{\gamma_2}(\epsilon, \delta) \leq \hat{\mathbf{p}}(\epsilon/2, \delta) \]}

\[ \leq \left( \frac{\ln \epsilon/2}{\ln \gamma_2} + 1 \right) \left( \frac{4\mathbf{v}_0(\hat{\mathbf{i}})/\epsilon}{\ln \gamma_2} \right) \left( \frac{2}{\ln \gamma_2} + 1 \right) \]}
\[ \leq \left( \frac{\ln \epsilon/2}{\ln \gamma_2} + 1 \right) \left( \frac{4\delta_{\max}(2/\delta)}{\ln \gamma_2} \right) \left( \frac{\ln \epsilon/2}{\ln \gamma_2} + 1 \right) \left( \frac{1 - \gamma_2^2/2}{1 - \gamma_2^2} \right) \left( 1 - \frac{1}{N} \right) \]}
\[ = \hat{T}^{\gamma_2}(\epsilon, \delta) \tag{89} \]

where the third inequality follows from eqn. (79).

We call the upper bound on \( T^{\gamma_2}(\epsilon, \delta) \) in (76) given in Theorem 7 the approximate averaging time. We use \( \hat{T}^{\gamma_2}(\epsilon, \delta) \) to characterize the convergence rate of \( A - NC \). We state a property of \( \hat{T}^{\gamma_2}(\epsilon, \delta) \).

**Lemma 9** Recall the spectral radius, \( \gamma_2 \), defined in eqn. (73). Then, for \( 0 < \epsilon, \delta < 1 \), \( \hat{T}^{\gamma_2}(\epsilon, \delta) \) is an increasing function of \( \gamma_2 \) in the interval \( 0 < \gamma_2 < 1 \).

**Proof:** The lemma follows by differentiating \( \hat{T}^{\gamma_2}(\epsilon, \delta) \) with respect to \( \gamma_2 \).

We now study the convergence properties of the \( A - NC \) algorithm, eqn. (62), i.e., when the weight matrix \( W \) is of the form (71) and the spectral norm satisfies (72). Then, the averaging time becomes a function of the weight \( \alpha \). We make this explicit by denoting as \( T^{\alpha}(\epsilon, \delta) \) and \( \hat{T}^{\alpha}(\epsilon, \delta) \) the averaging time and the approximate averaging time, respectively.

**Theorem 10 (A - NC Averaging Time)** Consider the \( A - NC \) algorithm for distributed averaging, under
the assumptions given in Subsection V-A. Define

\[ T^*(\epsilon, \delta) = \inf_{0<\alpha<\lambda N} T^\alpha(\epsilon, \delta), \quad \bar{T}^*(\epsilon, \delta) = \inf_{0<\alpha<\lambda N} \bar{T}^\alpha(\epsilon, \delta) \]  

(91)

Then:

1) \( T^\alpha(\epsilon, \delta) < \infty, \quad \forall \ 0 < \epsilon, \delta < 1, \quad 0 < \alpha < \frac{2}{\lambda N(L)} \)  

(92)

This essentially means that the \( A - NC \) algorithm is realizable for \( \alpha \) in the interval \( (0, \frac{2}{\lambda N(L)}) \).

2) For \( 0 < \alpha < \frac{2}{\lambda N(L)} \) we have

\[
T^\alpha(\epsilon, \delta) \leq \bar{T}^\alpha(\epsilon, \delta)
\]

\[
\bar{T}^\alpha(\epsilon, \delta) = \left( \frac{\ln / 2}{\ln \gamma_2} + 1 \right) \left[ \left( \frac{4\alpha^2 \phi_{\text{max}}^2 \ln(2/\delta)}{K\epsilon} \right) \left( \frac{\ln / 2}{N \ln \gamma_2} + 1 \right) + \frac{1 - \gamma_2^2 \epsilon^2 / 4}{1 - \gamma_2^2} \left( 1 - \frac{1}{N} \right) \right]
\]

(93)

and \( \gamma_2 = \rho(I - \alpha L - \frac{1}{N} J) \).

3) For a given choice of the pair \( 0 < \epsilon, \delta < 1 \), the best achievable averaging time \( T^*(\epsilon, \delta) \) is bounded above by \( \bar{T}^*(\epsilon, \delta) \), given by

\[
\bar{T}^*(\epsilon, \delta) = \inf_{0<\alpha \leq \frac{2}{\lambda N(L)}} \left( \frac{\ln / 2}{\ln(1 - \alpha \lambda_2(L))} + 1 \right) \left[ \left( \frac{4\alpha^2 \phi_{\text{max}}^2 \ln(2/\delta)}{K\epsilon} \right) \left( \frac{\ln / 2}{N \ln(1 - \alpha \lambda_2(L))} \right) + \frac{1 - (1 - \alpha \lambda_2(L))^2 \epsilon^2 / 4}{1 - (1 - \alpha \lambda_2(L))^2} \left( 1 - \frac{1}{N} \right) \right] + 1
\]

(94)

(95)

Note that in (95) the optimization is over a smaller range on \( \alpha \) than in (91).

Proof: The iterations for the \( A - NC \) algorithm are given by (70), where the weight matrix \( W \) is given by (71) and the spectral norm \( \gamma_2 \) by (72). Also, since \( \chi^p(i) = -\alpha n^p(i) \), we get

\[
\sup_{i \geq 0, p \geq 1} \sup_{1 \leq t \leq N} \mathbb{E} \left[ (\chi^p(i))^2 \right] \leq \alpha^2 \phi_{\text{max}}^2 < \infty
\]

(96)

Then, the assumptions (73,74) in Theorem 7 are satisfied for \( \alpha \) in the range (66) and the two items (92) and (94) follow.
To prove item 3), we note that it follows from 2) that \( T^*(\varepsilon, \delta) \leq \hat{T}^*(\varepsilon, \delta) \), where

\[
\hat{T}^*(\varepsilon, \delta) = \inf_{0 < \alpha < \frac{2\lambda_2(L)}{\lambda_N(L)}} \hat{T}^\alpha(\varepsilon, \delta)
\]

\[
= \inf_{0 < \alpha < \frac{2\lambda_2(L)}{\lambda_N(L)}} \left( \frac{\ln \varepsilon / 2}{\ln \gamma_2} + 1 \right) \times \left[ \left( \frac{4\phi_{\text{max}}^2 \ln(2/\delta)}{K \epsilon} \right) \left( \frac{\ln \varepsilon / 2}{N \ln \gamma_2} + \frac{1}{N} + \frac{1 - \gamma_2^2 \varepsilon^2 / 4}{1 - \gamma_2^2} \left( 1 - \frac{1}{N} \right) \right) + 1 \right]
\]

and \( \gamma_2 = \rho \left( 1 - \alpha L - \frac{1}{N} J \right). \) Now, consider the functions

\[
g(\gamma_2) = \frac{4\phi_{\text{max}}^2 \ln(2/\delta)}{K \epsilon} \left( \frac{\ln \varepsilon / 2}{\ln \gamma_2} + 1 \right) \left[ \frac{\ln \varepsilon / 2}{N \ln \gamma_2} + \frac{1}{N} + \frac{1 - \gamma_2^2 \varepsilon^2 / 4}{1 - \gamma_2^2} \left( 1 - \frac{1}{N} \right) \right]
\]

and

\[
h(\gamma_2) = \left( \frac{\ln \varepsilon / 2}{\ln \gamma_2} + 1 \right)
\]

with \( \gamma_2 \) as before. Similar to Lemma 9, we can show that, \( g(\gamma_2) \) and \( h(\gamma_2) \) are non-decreasing functions of \( \gamma_2 \). It can be shown that \( \gamma_2 = \rho \left( 1 - \alpha L - \frac{1}{N} J \right) \) attains its minimum value at \( \alpha = \alpha^* \) (see [7]), where

\[
\alpha^* = \frac{2}{\lambda_2(L) + \lambda_N(L)}
\]

We thus have

\[
g \left( \rho \left( 1 - \alpha^* L - \frac{1}{N} J \right) \right) \leq g \left( \rho \left( I - \alpha L - \frac{1}{N} J \right) \right), \quad \alpha^* \leq \alpha < \frac{2}{\lambda_N(L)}
\]

and

\[
h \left( \rho \left( 1 - \alpha^* L - \frac{1}{N} J \right) \right) \leq h \left( \rho \left( I - \alpha L - \frac{1}{N} J \right) \right), \quad \alpha^* \leq \alpha < \frac{2}{\lambda_N(L)}
\]

which implies, that, for \( \alpha^* \leq \alpha < \frac{2}{\lambda_N(L)}, \)

\[
\hat{T}^\alpha(\varepsilon, \delta) = \alpha^2 g \left( \rho \left( I - \alpha L - \frac{1}{N} J \right) \right) + h \left( \rho \left( I - \alpha L - \frac{1}{N} J \right) \right)
\]

\[
\geq (\alpha^*)^2 g \left( \rho \left( I - \alpha^* L - \frac{1}{N} J \right) \right) + h \left( \rho \left( I - \alpha^* L - \frac{1}{N} J \right) \right)
\]

\[
= \hat{T}^{\alpha^*}(\varepsilon, \delta)
\]

So, there is no need to consider \( \alpha > \alpha^* \). This leads to

\[
\hat{T}^*(\varepsilon, \delta) = \inf_{0 < \alpha < \frac{2\lambda_2(L)}{\lambda_N(L)}} \hat{T}^\alpha(\varepsilon, \delta)
\]

\[
= \inf_{0 < \alpha \leq \frac{2}{\lambda_2(L) + \lambda_N(L)}} \hat{T}^\alpha(\varepsilon, \delta)
\]
Also, it can be shown that (see [7])
\[ \gamma_2 = \rho \left( I - \alpha L - \frac{1}{N} J \right) = 1 - \alpha \lambda_2(L), \quad 0 < \alpha \leq \frac{2}{\lambda_2(L) + \lambda_N(L)} \] (104)

This, together with eqn. (103) proves item 3).

We comment on Theorem 10. For a given connected network, it gives explicitly the weight \( \alpha \) for which the \( \mathcal{A} - \mathcal{NC} \) algorithm is realizable (i.e., the averaging time is finite for any \((\epsilon, \delta)\) pair.) For these choices of \( \alpha \), it provides an upper bound on the averaging time. The Theorem also addresses the problem of choosing the \( \alpha \) that minimizes this upper bound.\(^3\) There is, in general, no closed form solution for this problem and, as demonstrated by eqn. (95), even if a minimizer exists, its value depends on the actual choice of the pair \((\epsilon, \delta)\). In other words, in general, there is no uniform minimizer of the averaging time.

C. \( \mathcal{A} - \mathcal{NC} \): Numerical Studies

We present numerical studies on the \( \mathcal{A} - \mathcal{NC} \) algorithm. We consider a sensor network of \( N = 230 \) nodes, with communication topology given by an LPS-II Ramanujan graph (see [10]), of degree 6.\(^4\) For the first set of simulations, we take \( \phi_{\text{max}}^2 = 100 \), \( K = 50 \), and fix \( \delta \) at 0.05 (this guarantees that the estimate belongs to the \( \epsilon \)-ball with probability at least 0.95.) We vary \( \epsilon \) in steps, keeping the other parameters fixed, and compute the optimal \((\epsilon, \delta)\) averaging time, \( \tilde{T}^*(\epsilon, \delta) \), given by eqn. (95) and the corresponding optimum \( \alpha^* \). Fig. 2 on the left plots \( \tilde{T}^*(\epsilon, \delta) \) as a function of \( \epsilon \), while Fig. 2 on the right plots \( \alpha^* \) vs. \( \epsilon \). As expected, \( \tilde{T}^*(\epsilon, \delta) \) decreases with increasing \( \epsilon \). The behavior of \( \alpha^* \) is interesting. It shows that, to improve accuracy (small \( \epsilon \)), the link weight \( \alpha \) should be chosen appropriately small, while, for lower accuracy, \( \alpha \) can be increased, which speeds the algorithm. Also, as \( \epsilon \) becomes larger, \( \alpha^* \) increases to make the averaging time smaller, ultimately saturating at \( \alpha^* \), given by eqn. (99). This behavior is similar to the \( \mathcal{A} - \mathcal{ND} \) algorithm, where slower decreasing (smaller) weight sequences correspond to smaller asymptotic m.s.e. at a cost of lower convergence rate (increased accuracy.)

In the second set of simulations, we study the tradeoff between the number of iterations per Monte-Carlo pass, \( \hat{i} \), and the total number of passes, \( \hat{p} \). Define the quantities, as suggested by eqn. (89):
\[ \hat{i}^* = \frac{\ln \epsilon / 2}{\ln (1 - \alpha^* \lambda_2(L))} + 1 \] (105)

\(^3\) Note, that the minimizer, \( \alpha^* \) of the upper bound \( \tilde{T}^\alpha(\epsilon, \delta) \) does not necessarily minimize the actual averaging time \( T^\alpha(\epsilon, \delta) \). However, as will be demonstrated by simulation studies, the upper bound is tight and hence \( \alpha^* \) obtained by minimizing \( \tilde{T}^\alpha(\epsilon, \delta) \) is a good enough design criterion.

\(^4\) This is a 6-regular graph, i.e., all the nodes have degree 6.
Fig. 2. Left: Plot of $\hat{T}^*(\epsilon, \delta)$ with varying $\epsilon$, keeping $\delta$ fixed at .05. Right: Plot of $\alpha^*$ with varying $\epsilon$, keeping $\delta$ fixed at .05.

\[
\hat{p}^* = \left( \frac{4\alpha^*^2 \phi_{\text{max}}^2 \ln(2/\delta)}{K\epsilon} \right) \left( \frac{\hat{i}^*}{N} + \frac{1 - (1 - \alpha^* \lambda_2(L))^2 \epsilon^2 / 4}{1 - (1 - \alpha^* \lambda_2(L))^2} \left( 1 - \frac{1}{N} \right) \right) + 1
\]  

(106)

where $\alpha^*$ is the minimizer in eqn. (95). In the following, we vary $\epsilon$ and the channel noise variance $\phi_{\text{max}}^2$, taking $K = 50$, $\delta = .05$, and using the same communication network. In particular, in Fig. 3 (left) we plot $\hat{i}^*$ vs. $\epsilon$ for $\phi_{\text{max}}^2 = 10$, while in Figs. 3 (center) and 3 (right), we repeat the same for $\phi_{\text{max}}^2 = 30$ and $\phi_{\text{max}}^2 = 100$ respectively. The figures demonstrate an interesting trade-off between $\hat{i}^*$ and $\hat{p}^*$, and show that for smaller values of the channel noise variance, the number of Monte-Carlo passes, $\hat{p}^*$ are much smaller than to the number of iterations per pass, $\hat{i}^*$, as expected. As the channel noise variance grows, i.e., as the channel becomes more unreliable, we note, that $\hat{p}^*$ increases to combat the noise accumulated at each pass. Finally, we present a numerical study to verify the tightness of the bound $\hat{T}^*(\epsilon, \delta)$. For this

we consider a network of $N = 100$ nodes with $M = 5N$ edges (generated according to an Erdős-Rényi graph, see [37].) We consider $\phi_{\text{max}}^2 = 80$, $K = 50$, and $\delta = .05$. To obtain $T^*(\epsilon, \delta)$ for varying $\epsilon$, we fix $\alpha$, sample $x(0) \in \mathcal{K}$ and for each such $x(0)$, we generate 100 runs of the $\mathcal{A} - \mathcal{NC}$ algorithm. We
check the condition in eqn. (69) and compute $T^\alpha(\epsilon, \delta)$. We repeat this experiment for varying $\alpha$ to obtain

$$T^*(\epsilon, \delta) = \inf_{\alpha} T^\alpha(\epsilon, \delta).$$

We also obtain $\hat{T}^*(\epsilon, \delta)$ from eqn. (95). Fig. 4 on the left plots $T^*(\epsilon, \delta)$ (solid red line) and $\hat{T}^*(\epsilon, \delta)$ (dotted blue line) w.r.t $\epsilon$, while Fig. 4 on the right plots the ratio $\frac{\hat{T}^*(\epsilon, \delta)}{T^*(\epsilon, \delta)}$ w.r.t. $\epsilon$. The plots show that the bound $\hat{T}^*(\epsilon, \delta)$ is reasonably tight, especially at small and large values of $\epsilon$, with fluctuations in between. Note that because the numerical values obtained for $T^*(\epsilon, \delta)$ are a lower bound, the bound $\hat{T}^*(\epsilon, \delta)$ is actually tighter than appears in the plots.

The bound $\hat{T}^*(\epsilon, \delta)$ is significant since: i) it is easy to compute from eqn. (95); ii) it is a reasonable approximation to the exact $T^*(\epsilon, \delta)$; iii) it avoids the costly simulations involved in computing $T^*(\epsilon, \delta)$ by Monte Carlo; and iv) it gives the right tradeoff between $\hat{i}^*$ and $\hat{p}^*$ (see eqns. (105-106)), thus determining the stopping criterion of the $A - NC$ algorithm.

D. $A - NC$: Generalizations

In this Subsection, we suggest generalizations to the $A - NC$ algorithm. For convenience of analysis, we assumed before a static network and Gaussian noise. These assumptions can be considerably weakened. For instance, the static network assumption may be replaced by a random link failure model with $\lambda_2(\overline{L}) > 0$, where $\overline{L} = \mathbb{E}[L]$. Also, the independent noise sequence in eqn. (64) may be non-Gaussian. In this case, the $A - NC$ algorithm iterations will take the form

$$x^p(i + 1) = (x^p(i) - \alpha L^p(i)x^p(i)) - \alpha n^p(i), \quad 0 \leq i \leq \hat{i} - 1, \quad 1 \leq p \leq \hat{p}, \quad x^p(0) = x(0)$$

(107)

The definition of $T^*(\epsilon, \delta)$ (see eqn. (69)) requires the infimum for all $x(0) \in K$, which is uncountable, so we verify eqn. (69) by sampling points from $K$. The $T^*(\epsilon, \delta)$, thus obtained, is in fact, a lower bound for the actual $T^*(\epsilon, \delta)$. 

---

5The definition of $T^*(\epsilon, \delta)$ (see eqn. (69)) requires the infimum for all $x(0) \in K$, which is uncountable, so we verify eqn. (69) by sampling points from $K$. The $T^*(\epsilon, \delta)$, thus obtained, is in fact, a lower bound for the actual $T^*(\epsilon, \delta)$. 

---

Fig. 4. Left: Plot of $T^*(\epsilon, \delta), \hat{T}^*(\epsilon, \delta)$ with varying $\epsilon$. Right: Plot of $\frac{\hat{T}^*(\epsilon, \delta)}{T^*(\epsilon, \delta)}$ with varying $\epsilon$. 

---

The bound $\hat{T}^*(\epsilon, \delta)$ is significant since: i) it is easy to compute from eqn. (95); ii) it is a reasonable approximation to the exact $T^*(\epsilon, \delta)$; iii) it avoids the costly simulations involved in computing $T^*(\epsilon, \delta)$ by Monte Carlo; and iv) it gives the right tradeoff between $\hat{i}^*$ and $\hat{p}^*$ (see eqns. (105-106)), thus determining the stopping criterion of the $A - NC$ algorithm.
where, $L^p(i)$ is an i.i.d. sequence of Laplacian matrices with $\lambda_2(L) > 0$. We then have,

$$x^p(\hat{r}) = \left( \prod_{j=0}^{p-1} (I - \alpha L^p(j)) \right) x(0) + H^p(\hat{r}), \quad 1 \leq p \leq \hat{p}$$

(108)

It is clear that $\{H^p(\hat{r})\}_{1 \leq p \leq \hat{p}}$ is an i.i.d. sequence of zero mean random variables. Under the assumption $\lambda_2(L) > 0$, there exists $\alpha$ (see [11]) such that

$$\mathbb{P}\left[ \left( \prod_{j=0}^{p-1} (I - \alpha L^p(j)) \right) x(0) = r1 \right] = 1, \quad 1 \leq p \leq \hat{p}$$

(109)

We thus choose $\hat{r}$ so that $\left( \prod_{j=0}^{p-1} (I - \alpha L^p(j)) \right) x(0)$ is sufficiently close to $r1$. The final estimate is

$$x^\hat{p}(\hat{r}) = \frac{1}{\hat{p}} \sum_{p=1}^{\hat{p}} \left( \prod_{j=0}^{p-1} (I - \alpha L^p(j)) \right) x(0) + \frac{1}{\hat{p}} \sum_{p=1}^{\hat{p}} H^p(\hat{r})$$

(110)

The first sum is close to $r1$ by choice of $\hat{r}$. We now choose $\hat{p}$, large enough, so that the second term is close to zero. In this way, we can apply the $\mathcal{A} - \mathcal{NC}$ algorithm to more general scenarios.

The above argument guarantees that $(\epsilon, \delta)$-consensus is achievable under these generic conditions, in the sense that the corresponding averaging time, $T^\alpha(\epsilon, \delta)$, will be finite. A thorough analysis requires a reasonable computable upper bound like eqn. (94), followed by optimization over $\alpha$ to give the best achievable convergence rate. (Note a computable upper bound is required, because, as pointed earlier, it is very difficult to find stopping criterion using Monte-Carlo simulations, and the resulting $T^*(\epsilon, \delta)$ will be a lower bound since the set $K$ is uncountable.) One way to proceed is to identify the appropriate large deviation rate (as suggested in eqn. (81)). However, the results will depend on the specific nature of the link failures and noise, which we do not pursue in this paper due to lack of space.

VI. Conclusion

We consider distributed average consensus when the topology is random (links may fail at random times) and the communication in the channels is corrupted by additive noise. Noisy consensus leads to a bias-variance dilemma. We considered two versions of consensus that lead to two compromises to this problem: i) $\mathcal{A} - \mathcal{ND}$ fits the framework of stochastic approximation. It a.s. converges to the consensus subspace and to a consensus random variable $\theta$—an unbiased estimate of the desired average, whose variance we compute and bound; and ii) $\mathcal{A} - \mathcal{NC}$ uses repeated averaging by Monte Carlo, achieving $(\epsilon, \delta)$-consensus. In $\mathcal{A} - \mathcal{ND}$ the bias can be made arbitrarily small, but the rate at which it decreases can be traded for variance—trade-off between m.s.e. and convergence rate. $\mathcal{A} - \mathcal{NC}$ uses a constant weight $\alpha$ and hence outperforms $\mathcal{A} - \mathcal{ND}$ in terms of convergence rate. Computation-wise, $\mathcal{A} - \mathcal{ND}$ is
superior since $\cal A - \cal NC$ requires more inter-sensor coordination to execute the independent passes. The estimate obtained by $\cal A - \cal NC$ does not possess the nice statistical properties, including unbiasedness, as the computation is terminated after a finite time in each pass.

Finally, these algorithms may be applied to other problems in sensor networks with random links and noise, e.g., distributed load balancing in parallel processing or distributed network flow.

**Appendix**

**Proof of Theorem 1 and $\cal A - \cal ND$ Generalizations under Assumptions 1.2) and 2.2)**

*Proof:* [Theorem 1] The proof follows that of Theorem 2.7.1 in [32]. Suffices to prove it for $x(0) = x_0$ a.s., $x_0 \in \mathbb{R}^{N \times 1}$ is a deterministic starting state. Let the filtration $\{\cal F_i^x = \sigma \{x(j) : 0 \leq j \leq i\}\}$ w.r.t. which $\{x(i)\}_{i \geq 0}, x \in \mathbb{R}^{N \times 1}$ (and hence functionals of $\{x(i)\}_{i \geq 0}$) are adapted.

Define the function $W(i, x), i \geq 0, x \in \mathbb{R}^{N \times 1}$ as

$$W(i, x) = (1 + V(i, x)) \prod_{j \geq i} [1 + g(j)]$$ (111)

It can be shown that

$$\mathcal{L}W(i, x) \leq -\alpha(i) \phi(i, x), \ i \geq 0, \ x \in \mathbb{R}^{N \times 1}$$ (112)

and, hence, under the assumptions (Theorem 2.5.1 in [32])

$$\mathbb{P} \left( \liminf_{i \to \infty} \rho(x(i), B) = 0 \right) = 1$$ (113)

which, together with assumption (25), implies

$$\mathbb{P} \left( \liminf_{i \to \infty} V(i, x(i)) = 0 \right) = 1$$ (114)

Also, it can be shown that the process $(W(i, x(i)), \cal F_i^x)$ is a non-negative supermartingale (Theorem 2.2.2 in [32]) and, hence, converges a.s. to a finite value. It then follows from eqn. (111) that $V(i, x(i))$ also converges a.s. to a finite value. Together with eqn. (114), the a.s. convergence of $V(i, x(i))$ implies

$$\mathbb{P} \left( \lim_{i \to \infty} V(i, x(i)) = 0 \right) = 1$$ (115)

The theorem then follows from assumptions (23) and (24) (see also Theorem 2.7.1 in [32].) $\blacksquare$

**Theorem 11 ($\cal A - \cal ND$: Convergence)** Consider the $\cal A - \cal ND$ algorithm given in Section IV-A with arbitrary initial state $x(0) \in \mathbb{R}^{N \times 1}$, under the Assumptions 1.2), 2.2), 3). Then,

$$\mathbb{P} \left[ \lim_{i \to \infty} \rho(x(i), C) = 0 \right] = 1$$ (116)
Proof: In the $\mathcal{A} - \mathcal{N} \mathcal{D}$ eqn. (9), the Laplacian $L(i, x(i))$ and the noise $n(i, x(i))$ are both state dependent. We follow Theorem 3 till eqn. (38) and modify eqn. (39) according to the new assumptions. The sequences $\{L(i, x)\}$ and $\{n(i, x)\}$ are independent. By the Gershgorin circle theorem, the eigenvalues of $(L(i, x) - \mathcal{T})$ are less than $2N$ in magnitude. From the noise variance growth condition we have,

$$
\mathbb{E} [\alpha^2(i)n^T(i, x)\mathcal{T}n(i, x)] = \alpha^2(i)\mathbb{E} [n^T_{c^+} (i, x) \mathcal{T}n_{c^+} (i, x)] \\
\leq \alpha^2(i)\lambda_N(L)\mathbb{E} [\|n_{c^+} (i, x)\|^2] \\
\leq \alpha^2(i)\lambda_N(L) [c_1 + c_2 \|x_{c^+}\|^2] \quad (117)
$$

Using eqn. (117) and a sequence of steps similar to eqn. (39) we have

$$
\mathcal{L}V(i, x) = -2\alpha(i)x^T\mathcal{T}^2 x + \alpha^2(i)x^T\mathcal{T}^3 x \\
+ \mathbb{E} [\alpha^2(i) ((L(i, x) - \mathcal{T}) x)^T \mathcal{T} ((L(i, x) - \mathcal{T}) x)] + \mathbb{E} [\alpha^2(i)n(i, x)^T \mathcal{T}n(i, x)] \\
\leq -2\alpha(i)x^T\mathcal{T}^2 x + \alpha^2(i)\lambda_N(\mathcal{T})^3 \|x_{c^+}\|^2 + 4\alpha^2(i)N^2\lambda_N(\mathcal{T}) \|x_{c^+}\|^2 \\
+ \alpha^2(i)\lambda_N(\mathcal{T}) [c_1 + c_2 \|x_{c^+}\|^2]
$$

Now, using the fact that $x^T\mathcal{T}x \geq \lambda_2(\mathcal{T}) \|x_{c^+}\|^2$, we have

$$
\mathcal{L}V(i, x) \leq -2\alpha(i)x^T\mathcal{T}^2 x + \alpha^2(i) \left[ c_1\lambda_N(\mathcal{T}) + \frac{\lambda_N^3(\mathcal{T})}{\lambda_2(\mathcal{T})} x^T\mathcal{T}x \\
+ \frac{4N^2\lambda_N(\mathcal{T})}{\lambda_2(\mathcal{T})} x^T\mathcal{T}x \right] \\
\leq -\alpha(i) \varphi(i, x) + g(i) [1 + V(i, x)]
$$

where $\varphi(i, x) = 2x^T\mathcal{T}^2 x$, $g(i) = \alpha^2(i) \max \left( c_1\lambda_N(\mathcal{T}), \frac{\lambda_N^3(\mathcal{T})}{\lambda_2(\mathcal{T})} + \frac{4N^2\lambda_N(\mathcal{T})}{\lambda_2(\mathcal{T})} \right)$. It can be verified that $\varphi(i, x)$ and $g(i)$ satisfy the conditions for Theorem 1. Hence (116).

Theorem 12 Consider the $\mathcal{A} - \mathcal{N} \mathcal{D}$ algorithm under the Assumptions 1.2, 2.1, and 3). Then, there exists a.s. a finite real random variable $\theta$ such that

$$
\mathbb{P} \left[ \lim_{i \to \infty} x(i) = \theta \mathbf{1} \right] = 1 \quad (118)
$$

Proof: Note that $1^TL(i, x(i)) = 0$, $\forall i$. The proof then follows from Theorem 4, since the noise assumptions are the same.

Lemma 13 Let $\theta$ be as given in Theorem 12 and $r$, the initial average, as given in eqn. (4), under the
Assumptions 1.2), 2.1), and 3). Let the m.s.e. $\zeta = E[\theta - r]^2$. Then we have:

1) **Unbiasedness:**

$$E[\theta] = r$$

(119)

2) **M.S.E. Bound:**

$$\zeta \leq \frac{\eta}{N^2} \sum_{i \geq 0} \alpha^2(i)$$

(120)

**Proof:** Follows from Theorem 12 and Lemma 5.

It is possible to have results similar to Theorem 12 and Lemma 13 under Assumption 2.2) on the noise. In that case, we need exact mixing conditions on the sequence. Also, Assumption 2.2) places no restriction on the growth rate of the variance of the noise component in the consensus subspace. By Theorem 11, we still get a.s. consensus, but the m.s.e. may become unbounded, if no growth restrictions are imposed.

**REFERENCES**

[1] J. N. Tsitsiklis, “Problems in decentralized decision making and computation,” Ph.D., MIT, Cambridge, MA, 1984.
[2] R. O. Saber and R. M. Murray, “Consensus protocols for networks of dynamic agents,” in *American Control Conference*, vol. 2, June 2003, pp. 951–956.
[3] A. Jadbabaie, J. Lin, and A. S. Morse, “Coordination of groups of mobile autonomous agents using nearest neighbor rules,” *IEEE Trans. Automat. Contr.*, vol. AC-48, no. 6, pp. 988–1001, June 2003.
[4] C. Reynolds, “Flocks, birds, and schools: A distributed behavioral model,” *Computer Graphics*, vol. 21, pp. 25–34, 1987.
[5] T. Vicsek, A. Czirok, E. B. Jacob, I. Cohen, and O. Schochet, “Novel type of phase transitions in a system of self-driven particles,” *Physical Review Letters*, vol. 75, pp. 1226–1229, 1995.
[6] R. Olfati-Saber, J. A. Fax, and R. M. Murray, “Consensus and cooperation in networked multi-agent systems,” *IEEE Proceedings*, vol. 95, no. 1, pp. 215–233, January 2007.
[7] L. Xiao and S. Boyd, “Fast linear iterations for distributed averaging,” *Syst. Contr. Lett.*, vol. 53, pp. 65–78, 2004.
[8] S. Kar and J. M. F. Moura, “Ramanujan topologies for decision making in sensor networks,” in *44th Allerton Conference on Communication, Control, and Computing*, Monticello, IL, Sept. 2006.
[9] ———, “Topology for global average consensus,” in *40th Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, Oct. 2006.
[10] S. Kar, S. A. Aldosari, and J. M. F. Moura, “Topology for distributed inference on graphs,” *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp. 2609–2613, June 2008. [Online]. Available: http://arxiv.org/abs/cs/0606052
[11] S. Kar and J. M. F. Moura, “Sensor networks with random links: Topology design for distributed consensus,” *IEEE Transactions on Signal Processing*, vol. 56, 2008. [Online]. Available: http://arxiv.org/abs/0704.0954
[12] Y. Hatano and M. Mesbahi, “Agreement over random networks,” in *43rd IEEE Conference on Decision and Control*, vol. 2, Dec. 2004, pp. 2010–2015.
[13] M. G. Rabbat, R. D. Nowak, and J. A. Bucklew, “Generalized consensus computation in networked systems with erasure links,” in *Proc. of the 6th Intl. Wkshp. on Sign. Proc. Adv. in Wireless Comm.*, New York, NY, 2005, pp. 1088–1092.
[14] C. Wu, “Synchronization and convergence of linear dynamics in random directed networks,” *IEEE Transactions on Automatic Control*, vol. 51, no. 7, pp. 1207–1210, July 2006.
[15] A. T. Salehi and A. Jadbabaie, “On consensus in random networks,” in *The Allerton Conference on Communication, Control, and Computing*, Allerton House, IL, September 2006.
[16] M. Porfiri and D. Stilwell, “Stochastic consensus over weighted directed networks,” in *Proceedings of the 2007 American Control Conference*, New York City, USA, July 11-13 2007.

[17] Y. Hatano, A. K. Das, and M. Mesbahi, “Agreement in presence of noise: pseudogradients on random geometric networks,” in *44th IEEE Conf. on Decision and Control, and European Control Conference. CDC-ECC '05*, Seville, Spain, Dec. 2005.

[18] M. Huang and J. Manton, “Stochastic Lyapunov analysis for consensus algorithms with noisy measurements,” in *2007 American Control Conference*, New York City, NY, USA, Jul. 11-13 2007.

[19] ——, “Stochastic approximation for consensus seeking: mean square and almost sure convergence,” in *IEEE 46th Conference on Decision and Control*, New Orleans, LA, USA, Dec. 12-14 2007.

[20] I. D. Schizas, A. Ribeiro, and G. B. Giannakis, “Consensus-based distributed parameter estimation in ad hoc wireless sensor networks with noisy links,” in *IEEE Int. Conf. on Ac., Speech and Sig. Proc.*, Honolulu, HI, 2007, pp. 849–852.

[21] F. R. K. Chung, *Spectral Graph Theory*. Providence, RI : American Mathematical Society, 1997.

[22] B. Mohar, “The Laplacian spectrum of graphs,” in *Graph Theory, Combinatorics, and Applications*, Y. Alavi, G. Chartrand, O. R. Oellermann, and A. J. Schwenk, Eds. New York: J. Wiley & Sons, 1991, vol. 2, pp. 871–898.

[23] B. Bollobás, *Modern Graph Theory*. New York, NY: Springer Verlag, 1998.

[24] L. Xiao, S. Boyd, and S.-J. Kim, “Distributed average consensus with least-mean-square deviation,” *Journal of Parallel and Distributed Computing*, vol. 67, pp. 33–46, 2007.

[25] S. Kar and J. Moura, “Distributed consensus algorithms in sensor networks: Quantized data,” November 2007, submitted for publication, 30 pages. [Online]. Available: http://arxiv.org/abs/0712.1609

[26] T. C. Aysal, M. Coates, and M. Rabbat, “Distributed average consensus using probabilistic quantization,” in *IEEE/SP 14th Workshop on Statistical Signal Processing Workshop*, Madison, Wisconsin, USA, August 2007, pp. 640–644.

[27] S. Kar and J. M. F. Moura, “Distributed consensus algorithms in sensor networks with communication channel noise and random link failures,” in *41st Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 2007.

[28] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, “Randomized gossip algorithms,” *IEEE/ACM Trans. Netw.*, vol. 14, no. 1, pp. 2508–2530, 2006.

[29] R. Olfati-Saber, “Flocking for multi-agent dynamic systems: Algorithms and theory,” *IEEE Transactions on Automatic Control*, vol. 51, no. 3, pp. 401–420, 2006.

[30] H. Tanner, A. Jadbabaie, and G. J. Pappas, “Flocking in fixed and switching networks,” *IEEE Transactions on Automatic Control*, vol. 52, no. 5, pp. 863–868, 2007.

[31] Y. Kim and M. Mesbahi, “On maximizing the second smallest eigenvalue of a state-dependent graph Laplacian,” *IEEE Transactions on Automatic Control*, vol. 51, no. 1, pp. 116 – 120, Jan. 2006.

[32] M. Nevel’son and R. Has’minskii, *Stochastic Approximation and Recursive Estimation*. Providence, Rhode Island: American Mathematical Society, 1973.

[33] D. Williams, *Probability with Martingales*. Cambridge University Press, 1991.

[34] H. Kushner and G. Yin, *Stochastic Approximation and Recursive Algorithms and Applications*. Springer, 2003.

[35] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, “Randomized gossip algorithms,” *IEEE Transactions on Information Theory*, vol. 52, pp. 2508 – 2530, June 2006.

[36] D. Mosk-Aoyama and D. Shah, “Computing separable functions via gossip,” in *PODC ’06: Proceedings of the Twenty-Fifth Annual ACM Symposium on Principles of Distributed Computing*. New York, NY, USA: ACM Press, 2006, pp. 113–122.

[37] F. R. K. Chung and L. Lu, *Complex Graphs and Networks*. Boston, MA, USA: American Mathematical Society, 2006.