Abstract—Distributed node counting in wireless sensor networks can be important in various applications such as network maintenance and information aggregation. In this paper, a distributed consensus algorithm for estimating the number of nodes in a wireless sensor network in the presence of communication noise is introduced. In networks with a fusion center, counting the number of nodes can be easily done by letting each node transmit a fixed constant value to the fusion center. In a network without a fusion center, where nodes do not know the graph structure, estimating the number of nodes is not straightforward. The proposed algorithm is based on distributed average consensus, and norm estimation. Different sources of error are explicitly discussed, the Fisher information and the distribution of the final estimate are derived. Several design parameters and how they affect the performance of the algorithm are studied, which provide guidelines towards making the estimation error smaller. Simulation results corroborating the theory are also provided.


I. INTRODUCTION

There are many advantages of using a distributed network without a fusion center: a distributed system is more scalable than a centralized system with a fusion center, and it is more robust to link failures. Since the nodes in a decentralized network communicate only with their neighbors, the sensors require low power.

Counting the number of nodes in a decentralized network is essential in several applications. For example, some overlay maintenance protocols require the system size to incorporate a newly joined node in the system. In [3] the soft-max based max consensus method requires the system size (number of nodes). In [4], the sum of the initial values is calculated using a consensus approach by using the system size.

In a centralized network with a fusion center, counting the number of nodes in the network is straightforward: each node transmits a fixed constant value to the fusion center; the estimate of number of nodes can be obtained from the aggregate at the fusion center. However, node counting is more challenging in a decentralized system where sensors only have local information.

Zhang, Tepedelenlioğlu and Spanias are from the Sensip Center, School of ECCE, Arizona State University. Banavar is from the Department of ECE, Clarkson University. The authors from Arizona State University are funded in part by the NSF award ECCS - 1307982 and the Sensip Center, School of ECCE, Arizona State University. Banavar is funded in part by the NSF CRH award 1464222 (Email: {szhang98, cihan, spanias}@asu.edu, mbanavar@clarkson.edu). Parts of the material in this paper were presented at the 2015 Asilomar Conference on Signals, Systems and Computers.
nodes and nodes exchange information based on maximum or average consensus strategies. By generating random variables with a specific distribution, the maximum likelihood estimator for $N^{-1}$ can be expressed in terms of the initially generated random variables. It is also shown in [20] that if max consensus is used, the distribution of the random variables locally generated by the nodes is not a design variable and will not affect the estimation performance as long as the CDFs of random variables are strictly monotonic and continuous. However, the proposed uniform + maximum + ML algorithm is sensitive to noise since the max operator is used for max consensus, and max consensus diverges in the presence of communication noise. The Gaussian + average + ML algorithm of [20] was proposed in the absence of noise, and is a special case of our proposed algorithm, which works in the presence of communication noise.

A. Statement of Contributions

In this paper, we design a fully distributed node counting algorithm for any connected distributed network based on $L_2$ norm estimation and average consensus in the presence of communication noise. A linear iterative average consensus algorithm similar to [21] is used, with pre-processed initial values. Initial values are generated at each sensor node, denoted by $x = [x_1 \cdots x_N]^T$, where $N$ is the number of nodes in the network. Then by applying pre-processing, average consensus, and post processing, a scaled estimate of the $L_2$ norm of the initial values, $|x|^2/N^2$ is obtained at each node. By running another consensus algorithm on the square of the initial values, the nodes obtain an estimate of $|x|^2/N$. Finally, by taking the ratio, each node has an estimate of number of nodes. It is shown that our algorithm is a generalization of the ML approach in [20] where $x$ is chosen as a constant vector. The performance analysis in the presence of noise is provided and shows that the choice of $x$ affects performance. The sources of error between the states of nodes and the desired convergence result is quantified. The Fisher information and distribution of the estimate of $N$ at each node is also derived. The analysis not only shows how the performance of the algorithm is affected by the number of iterations, noise variance and structure of the graph, but also provides guidelines towards choosing the design parameters.

Compared to the literature, this is the first paper that considers node counting in the presence of communication noise. Nodes do not have to be labeled and they do not need to know the structure of the graph. We consider the consensus process explicitly, and the inevitable error caused by lack of convergence in finite number of consensus iterations and error caused by noise is also quantified.

B. Organization

The rest of this paper is organized as follows. In Sections II and III the system model and a brief review on average consensus are given. The proposed node counting algorithm is described in Section IV. In Section V different sources of error are explicitly discussed, and convergence rate and estimation error are quantified. The overall performance analysis is given in Section VI. The Fisher information and the distribution of the estimate of the number of nodes are calculated, and how the design parameters affect the performance is analyzed. In Section VII simulation results for the proposed node counting algorithms are presented. Finally, conclusions are given in Section VIII.

Compared to the conference version in [II], performance analysis of the algorithm is given, Fisher information is derived, and the steady state distribution of the estimator is provided. More simulations corroborating the results are also given.

II. SYSTEM MODEL

A. Graph Representation

The distributed wireless sensor network is modeled as an undirected connected graph $G = (N, E)$ containing a set of nodes $N = \{1, \ldots, N\}$ and a set of edges $E$. The set of neighbours of node $i$ is denoted by $N_i$, i.e., $N_i = \{j | \{i, j\} \in E\}$. Two nodes can communicate with each other if and only if they are neighbours. The number of neighbours of node $i$ is $d_i$. We define a diagonal degree matrix, $D = \text{diag}[d_1, d_2, \ldots, d_N]$. The connectivity structure of the graph is characterized by the adjacency matrix $A = \{a_{ij}\}$ such that $a_{ij} = 1$ if $\{i, j\} \in E$ and $a_{ij} = 0$ otherwise. The Laplacian matrix $L$ is defined as $L = D - A$. For a connected graph, the smallest eigenvalue of the graph Laplacian is always zero, i.e., $\lambda_1(L) = 0$ and $\lambda_2(L) > 0$, $i = 2, \ldots, N$. The performance of consensus algorithms often depends on $\lambda_2(L)$, which is also known as the algebraic connectivity of the graph [22].

B. Communications Model

We consider a connected network with $N$ nodes. Nodes do not have knowledge about the structure of the graph. We assume that the sensors maintain a state vector and each node broadcasts its state to its neighbors at each iteration. Nodes update the states based on local received states from their neighbors, which is described in Section III. We also assume analog transmissions between nodes [21], [23], [24] and the communication between nodes is imperfect with communication noise, which is i.i.d with 0 mean and variance $\sigma_n^2$.

III. REVIEW OF LINEAR AVERAGE CONSENSUS IN THE PRESENCE OF COMMUNICATION NOISE

Average consensus is well studied in literature [21], [25]–[28]. In this section, we briefly review the noisy linear average consensus algorithm. In [21], a linear distributed average consensus with communication noise is introduced. A weighted step $\alpha(t)$ is used to bound the variance of communication noise. To compute the average of initial state $y(0) = [y_1(0) \cdots y_N(0)]^T$, the average consensus algorithm can be expressed as,

$$y_i(t + 1) = [1 - \alpha(t)d_i]y_i(t) + \alpha(t)\sum_{j \in N_i} y_{ij}(t) + n_{ij}(t),$$

(1)
where \( i = 1, 2, \ldots, N \), and \( t = 0, 1, 2, \ldots \), is the time index. The value \( y_i(t+1) \) is the state update of node \( i \) at time \( t+1 \), \( y_j(t) \) is the state value of the \( j^{th} \) neighbour of node \( i \) at time \( t \), and \( n_{ij}(t) \) is the noise associated with the reception of \( y_j(t) \), which is assumed to be independent across time and space with zero mean and variance \( \sigma^2_n \). \( \alpha(t) \) is a positive step and is a decreasing function of \( t \).

For the convergence proof, we make the following assumptions on the system model:

**Assumptions:**

**A1) Connected Graph:** The graph is connect, i.e. \( \lambda_1(L) = 0 \) and \( \lambda_1(L) > 0, i = 2, \ldots, N \).

**A2) Independent Noise Sequence:** The reception noise is an independent sequence with zero mean and bounded variance, i.e.

\[
E[n_{ij}(t)] = 0, \forall N \geq i, j \geq 1, t \geq 0.
\]

\[
E[n^2_{ij}(t)] = \sigma^2_n \leq \infty.
\]

**A3) Persistence Condition:** The positive weight step \( \alpha(t) \) is a decreasing function of \( t \), but decreases not too fast, i.e.

\[
\alpha(t) > 0, \sum_{t=0}^{\infty} \alpha(t) = \infty, \quad \beta := \sum_{t=0}^{\infty} \alpha^2(t) < \infty.
\]

For completeness, we include the following theorem which characterizes the convergence result of the average consensus algorithm.

**Theorem 1.** Let \( y(t) = [y_1(t) \cdots y_N(t)]^T \) be the vector containing the states of nodes at time \( t \). Then under **A1), A2) and A3)**, by running the iterative algorithm as in equation (1), there exists finite real random variable \( \theta \) such that,

\[
Pr \left( \lim_{t \to \infty} y(t) = \theta \right) = 1.
\]

Let \( \bar{y} = \frac{1}{N} \sum_{i=1}^{N} y_i(0) \) be the average of the initial values. Define \( \xi = E[(\theta - \bar{y})^2] \) be the mean square error, is given by

\[
\xi = \left( \frac{\sum_{i=1}^{N} d_i}{N^2} \right) \sigma^2_n \beta
\]

\[
\leq \frac{(N-1)\sigma^2_n}{N} \beta.
\]

**Proof:** The proof is similar to the proof of Theorem 4 and Lemma 5 in [21]. Equation (4) can be obtained by assuming the initial values are zero and the network is converging to the average of scaled noise samples received at nodes.

**IV. Node Counting using Average Consensus**

**A. Problem Statement**

Consider a connected network with \( N \) nodes. We assume that the sensors always keep a state vector and they update it based on local received states from their neighbors. We also assume that the communication between nodes is imperfect with communication noise. It is desired that the nodes reach consensus on the number of nodes in the network.

Average consensus is well studied in literature, in which the states of the nodes converge to the sample mean of the initial states. The key to relating the network size to the average consensus is to observe that

\[
N = \frac{||x||^2_2}{||x||_2^2/N^2}.
\]

As will be seen in the paper, the value of the denominator and numerator in (3) can be estimated using the average consensus algorithm. Therefore, an estimate of the number of nodes in the network can be obtained. In the following subsections, the proposed node counting algorithm is introduced.

**B. Node Counting Algorithm**

By running the average consensus algorithm as mentioned in Section III, nodes in the sensor network converge to the sample mean of the initial values. As a result, the \( L_2 \) estimation method can be used to relate the average of the initial states and the number of nodes in the network. The node counting algorithm can be described in three phases: an estimate of the value of the denominator and numerator in (3) can be obtained using the average consensus algorithm in phase I and phase II respectively, and \( \hat{N} \) is calculated in phase III by using the consensus results of phases I and II to compute the ratio in (3).

In the following, details of the three phases of the algorithm are provided.

1) Phase I - \( L_2 \) Norm Estimation Consensus: In Phase I of the node counting algorithm, an estimate of the denominator in equation (3) is calculated based on \( L_2 \) norm estimation and average consensus algorithm. Assume the initial values are \( x = [x_1 \cdots x_i \cdots x_N] \), where \( x_i \) is the initial value at node \( i \). Each node in the network generates \( K \) initial state values. The initial state values at node \( i \) is denoted as \( y_i(0) = [y_i^{(1)}(0) \cdots y_i^{(K)}(0)] \), where \( y_i^{(k)}(0) = r_i^{(k)} x_i, 1 \leq k \leq K \), and \( r_i^{(k)} \) are i.i.d. random variables with zero mean and variance one.

By running average consensus algorithm, each node updates the \( k^{th} \) element in the state vector of node \( i \) at time \( t+1 \) with

\[
y_i^{(k)}(t+1) = [1 - \alpha(t) d_i] y_i^{(k)}(t) + \alpha(t) \sum_{j \in N_i} y_j^{(k)}(t) + n_{ij}^{(k)}(t),
\]

where \( n_{ij}^{(k)}(t) \) is the noise associated with the reception of \( y_j^{(k)}(t) \) and \( \alpha(t) \) satisfies equation (4). When \( t \) is large, the \( k^{th} \) element of node \( i \) is converging to a noisy version of the average \( \frac{1}{K} \sum_{i=1}^{N} r_i^{(k)} x_i \).

A post processing function \( f(\cdot) \) is applied at each node by squaring each element in the state vector and take the average of the result. For node \( i \), the post processed result can be expressed as,

\[
f(y_i(t)) = \frac{1}{K} ||y_i(t)||^2 = \frac{1}{K} \sum_{k=1}^{K} \left( y_i^{(k)}(t) \right)^2,
\]

where \( y_i(t) = [y_i^{(1)}(t) \cdots y_i^{(K)}(t)] \) is the state vector of node \( i \) at time \( t \).
Assume the consensus stops at iteration time \( t^* \). To relate the post processed result \( f(\mathbf{y}_i(t^*)) \) at time \( t^* \) to the \( L_2 \) norm of the initial values \( \mathbf{x} \), we have,

\[
f(\mathbf{y}_i(t^*)) = \frac{1}{K} \sum_{k=1}^{K} (\mathbf{y}_i^{(k)}(t^*))^2
\]

where \((\mathbf{y}_i^{(k)}(t^*)) \) is the average value of the initial state \( x_i \) over \( K \) consensus runs and \( f(\cdot) \) is a function that imposes a desirable property on the post processed result. The presence of noise affects the estimation of \( \mathbf{x} \) in the network, and the estimate of number of nodes \( N \) is affected by the presence of noise and initial values. The equation (12) holds since \( x_i \) is fixed and the expectation taken with respect to random variables \( r_i^{(k)} \), and (16) holds since \( r_i^{(k)} \) are i.i.d. random variables.

Note that in Phase I, \( K \) consensus runs are required in [9] before the computation of (10).

2) Phase II - \( L_2 \) Norm Consensus: In Phase II of the node counting algorithm, the numerator in equation (8) is calculated based on the definition of \( L_2 \) norm and average consensus algorithm. Each node \( i \) sets its initial state value to \( z_i(0) = x_i^2 \).

Nodes in the network run:

\[
z_i(t+1) = [1 - \alpha(t)d_i] z_i(t) + \alpha(t) \sum_{j \in \mathcal{N}_i} [z_{ij}(t) + n_{ij}(t)].
\]

After \( t^* \) iterations we have,

\[
z_i(t^*) \approx \frac{1}{N} ||\mathbf{x}||_2^2,
\]

where the error in (18) is discussed in Section V similar to (12). Note that Phase II requires a single consensus run.

3) Phase III - Node Counting: By comparing the results from equations (16) and (13), the estimate of number of nodes in the network \( \hat{N}_i(t^*) \) at node \( i \) at time \( t^* \) can be obtained as

\[
\hat{N}_i(t^*) = \frac{z_i(t^*)}{f(\mathbf{y}_i(t^*))}.
\]

Note that Phase I and Phase II can be done at the same time by using a \( K+1 \) state vector containing both the \( K \times 1 \) process \( \mathbf{y}_i(t) \) and the scalar \( z_i(t) \).

From (19), the algorithm works for any \( \mathbf{x} \neq 0 \). In the algorithm, \( x_i \) can be sensor measurements or values designed for improved performance (see Section V). Random variables \( r_i^{(k)} \) are i.i.d. and generated at nodes with mean 0 and variance 1. Two simple ways to choose \( r_i^{(k)} \) can be: i) Normally distributed, \( r_i^{(k)} \sim N(0,1) \); ii) Bernoulli distributed with \( \pm 1 \), i.e. \( \Pr[r_i^{(k)} = 1] = \Pr[r_i^{(k)} = -1] = 1/2 \). How the value of \( x_i \) and \( r_i^{(k)} \) affect the performance is detailed analyzed in Section V and VI. In the following subsection, a special case where initial values are chosen to be all equal, \( x_i = a \) is considered.

C. Special Case: Equal \( x_i \)

Recall that the node counting algorithm is based on the estimation of \( ||\mathbf{x}||_2^2 \) using average consensus algorithm. The algorithm works for any \( \mathbf{x} \neq 0 \) value. Therefore, a special case of the node counting algorithm is to let all the initial values to be a fixed constant value \( a \) and the initial state vector of the node \( i \) is \( \mathbf{y}_i(0) = [y_i^{(1)}(0) \cdots y_i^{(K)}(0)] \) and \( y_i^{(k)}(0) = ar_i^{(k)} \).

Then by running the average consensus algorithm as in equation (9) and applying the same post processing function after consensus is reached, the final result of Phase I at node \( i \) at time \( t^* \) can be expressed as,

\[
f(\mathbf{y}_i(t^*)) = \frac{1}{K} \sum_{k=1}^{K} (y_i^{(k)}(t^*))^2
\]

\[
\approx \frac{1}{K} \sum_{k=1}^{K} \left( \frac{1}{N} \sum_{i=1}^{N} ar_i^{(k)} \right)^2 = \frac{a^2}{N}.
\]

Since \( a \) is a constant known to all nodes, Phase II is not needed and the estimate of \( N \) can be obtained as,

\[
\hat{N}_i(t^*) = \frac{\alpha^2}{f(\mathbf{y}_i(t^*))}.
\]

Note that the value of \( a \) controls the power of transmitted signal from each node, which makes no difference in the absence of communication noise and is chosen in (20) as \( a = 1 \) along with the choice \( r_i^{(k)} \sim N(0,1) \). However, in the presence of noise considered herein, \( a \) offers a trade off between transmit power and SNR.

V. SOURCES OF ERROR

In this subsection, different sources of error are explicitly discussed. The transient analysis of the consensus result is given in Section V-A. We show that the bias of the convergence result is going to zero with number of iterations, and the graph structure and initial states affect the convergence speed of the bias.

The sources of steady state error are also analyzed. The error caused by communication noise is considered in Section V-B where the MSE is shown to depend on the noise variance and the step size. In Section V-C the error in the \( L_2 \) norm estimation is described where how the values \( r_i^{(k)} \), \( x_i \), and \( K \) affect the error is studied.

A. Transient of Bias

We now quantify the transient of the bias in the algorithm to see how it decays with the number of iterations. The states of nodes ideally converge to the average state vector \( \mathbf{y} = [\breve{y}^{(1)} \cdots \breve{y}^{(k)} \cdots \breve{y}^{(K)}]^T \), where \( \breve{y}^{(k)} = \frac{1}{N} \sum_{i=1}^{N} r_i^{(k)} x_i \).

Let \( \mathbf{y}^{(k)}(t) = [y_i^{(k)}(t) y_2^{(k)}(t) \cdots y_N^{(k)}(t)]^T \) contain the \( k \)th element in the state vector from all \( N \) nodes at time \( t \). The
convergence rate of the mean of \( y^{(k)}(t) \) is quantified by [21 eqn (61)] as
\[
\left\| \mathbb{E} \left[ y^{(k)}(t) \right] - \bar{y}^{(k)} \right\|_2 \leq \left( e^{-\lambda_2(L)} \sum_{t=0}^{t} \alpha(t) \right) \left\| y^{(k)}(0) \right\|_2.
\] (22)

It is clear from equation (22) that the convergence is fast if the algebraic connectivity \( \lambda_2(L) \) is large, which implies a faster convergence of bias in a more connected graph. To see this more clearly, we further simplify the derivation of (22) on the iteration index \( t \) by assuming \( \alpha(t) = \frac{1}{t+1} \) as in [21] and [24]. We have the following approximation,
\[
\sum_{\tau=0}^{t} \alpha(\tau) = \sum_{\tau=0}^{t} \frac{1}{\tau + 1} = \ln(t + 1) + \gamma + \varepsilon_{t+1}
\] (23)
where \( \gamma \) is the Euler constant and \( \varepsilon_{t+1} \sim \frac{1}{2(t+1)} \) which approaches 0 as \( t \) goes to infinity. Therefore, the convergence rate expression in (22) can be expressed for large \( t \) as,
\[
\left\| \mathbb{E} \left[ y^{(k)}(t) \right] - \bar{y}^{(k)} \right\|_2 \leq (t + 1)^{-\lambda_2(L)} \left\| y^{(k)}(0) \right\|_2.
\] (24)
Equation (24) shows that the error \( \left\| \mathbb{E} \left[ y^{(k)}(t) \right] - \bar{y}^{(k)} \right\|_2 \) is polynomial decreasing with \( t \) with an exponent given by the algebraic connectivity of the graph.

**B. Mean Square Error**

Even though \( \mathbb{E}[y^{(k)}(t)] \to \bar{y}^{(k)} \) as \( t \to \infty \), the elements in the state vectors do not converge to the true average of the initial states due to the fact that communications between nodes is noisy. Instead, the \( k \)th elements of the state vectors for all nodes converge a.s. to a finite random variable \( \bar{y}^{(k)} \) as in Theorem [1] which is an unbiased estimator of the average and satisfies the following properties,
\[
\mathbb{E}[^{k}y] = \frac{1}{N} \sum_{i=1}^{N} x_i = \bar{y}^{(k)},
\] (25)
\[
\zeta^{(k)} = \mathbb{E} \left[ (\theta^{(k)} - \bar{y}^{(k)})^2 \right] \leq \frac{(N-1)\sigma_n^2 \beta}{N}.
\] (26)

where \( \sigma_n^2 \) is the noise variance.

It is seen in (26) that the error is proportional to \( \sigma_n^2 \) and is bounded if \( \alpha(t) \) satisfies equation (4).

**C. L2 Norm Estimation Error**

It is seen from [13] and [14] that the \( L_2 \) estimation result in [16] can be obtained due to the law of large numbers (large \( K \)). We now study the effect of \( K \) on variance of the \( L_2 \) estimation result. The analysis on how the design parameters \( r_i^{(k)} \) and \( x_i \) will affect the variance is also given. Let
\[
Y = \frac{1}{K} \sum_{k=1}^{K} \left( \frac{1}{N} \sum_{i=1}^{N} r_i^{(k)} x_i \right)^2.
\] (27)
Then it can be shown that \( \mathbb{E}[Y] = ||x||_2^2/N^2 \) since \( r_i^{(k)} \) are i.i.d. random variables. In the following, the relationship between the value of \( K \) and the variance of \( Y \), denoted as \( \sigma_Y^2 \) will be shown.

Let \( Z^2 = \left( \frac{1}{K} \sum_{i=1}^{N} r_i^{(k)} x_i \right)^2 \) be an unbiased estimator of \( ||x||_2^2 \) because of (16). The variance of \( Z^2 \) can be calculated as,
\[
\text{Var}[Z^2] = \mathbb{E}[Z^4] - (\mathbb{E}[Z^2])^2 = \left( \left( \mathbb{E} \left[ r_i^{(k)} \right]^4 \right) / N^4 \right) - 1 \sum_{i=1}^{N} x_i^4 + \frac{4}{N^4} \sum_{i<j} x_i^2 x_j^2.
\] (29)

Then, the variance of \( Y \) can be expressed as,
\[
\sigma_Y^2 = \text{Var}[Y] = \frac{1}{K} \mathbb{E}[Z^2] (30)
\]

Equation (29) and (30) shows that the variance will be small when \( K \) chosen to be large, and it is also related to \( r_i^{(k)} \) and \( x_i(0) \). Therefore there is a trade-off between the accuracy of the algorithm and the storage at sensor nodes: a more accurate estimate of number of nodes can be obtained if \( K \) large, but the nodes need to keep a larger state vector and therefore increase the required storage at nodes.

For deterministic \( x_i \) and \( K \) values, the distribution of \( r_i^{(k)} \) also affects variance of the \( L_2 \) norm estimation result. \( r_i^{(k)} \) needs to be chosen to be \( 0 \) mean and variance 1 as mentioned. In the following, we calculate the \( L_2 \) norm estimation variance for two common \( r_i^{(k)} \) distributions: (i) For \( r_i^{(k)} \sim \mathcal{N}(0,1) \), we have,
\[
\text{Var}[Y] = \frac{1}{K N^4} \left( 2 \sum_{i=1}^{N} x_i^4 + 4 \sum_{i<j} x_i^2 x_j^2 \right).
\] (31)
If initial values are chosen to be equal, \( x_i = a \), we have,
\[
\text{Var}[Y] = \frac{2a^4 + 2(N-1)a^4}{K N^3} = \frac{2a^4}{K N^2}.
\] (32)
(ii) If \( r_i^{(k)} \) is Bernoulli distributed with \( \text{Pr}[r_i = 1] = \text{Pr}[r_i = -1] = \frac{1}{2} \), we have,
\[
\text{Var}[Y] = \frac{4}{K N^4} \sum_{i<j} x_i^2 x_j^2.
\] (33)
If initial values are chosen to be equal, \( x_i = a \), we have,
\[
\text{Var}[Y] = \frac{2(N-1)a^4}{K N^3}.
\] (34)

Note that (31) is always larger than (33) by \( 2(\sum_i x_i^4)/KN^4 \).

The following theorem characterizes the minimum \( L_2 \) norm estimation variance.

**Theorem 2.** The distribution of \( r_i^{(k)} \) that minimizes the \( L_2 \) norm estimation error in equation (29) and (30) is Bernoulli distribution with \( \pm 1 \) with probability 0.5.

**Proof:** For fixed initial values \( x_i \) and system size \( N \), the \( L_2 \) estimation variance is related to \( \mathbb{E} \left[ \left( \mathbb{E} \left[ r_i^{(k)} \right]^4 - 1 \right) \right] \) as shown in equation (29). For any \( r_i^{(k)} \) distribution with mean 0 and variance 1, the fourth moment satisfies,
\[
\mathbb{E} \left[ r_i^{(k)} \right]^4 \geq \left( \mathbb{E} \left[ \left( r_i^{(k)} \right) ^3 \right] \right)^2 + 1 \geq 1.
\] (35)

Equation (35) can be obtained from a lower bound on Kurtosis in [29] by setting the variance to be 1. Bernoulli distribution
with $\pm 1$ achieves the lower bound 1 and Theorem 2 is proved.

Based on the above analysis, we have the following observations and conclusions on the $L_2$ norm estimation error. The $L_2$ norm estimation variance is inversely proportional to $K$. By comparing Equation (31) and (33), it is seen that for fixed $x_i$ values, the $L_2$ norm estimation variance will be smaller if $r_i^{(k)}$ are chosen to be Bernoulli distributed. The difference between Equation (31) and (33) will be small when $N$ is large; When $r_i^{(k)}$ is Bernoulli distributed, the $L_2$ norm estimation variance achieves 0 if the initial values are chosen as follows:

$$x_1 \neq 0; \ x_i = 0, i \neq 1.$$ (36)

However, this choice if not fully distributed since setting $x_1 \neq 0$ requires labeling at least one node.

Also note that if we assume that consensus is reached ($t \to \infty$), $\sigma_n^2 = 0$ and $N \to \infty$, we see from equation (6), (26) and (29) that the value of $r_i^{(k)}$ and $x_i$ will not affect the performance of the algorithm. This matches the result in Section III in (20) that for average consensus when there is no noise and $N$ is large, one can use central limit theorem to assume Gaussianity. As a result the distribution of the initial random variables locally generated by the nodes will not affect performance.

To sum up, we quantified three sources of errors: lack of convergence in finite iterations, error caused by communication noise and $L_2$ norm estimation error. In the following subsections, the distribution of $N_i$ and Fisher information are calculated, followed by the analysis on how the design parameters will affect the overall performance.

VI. OVERALL PERFORMANCE ANALYSIS

In this section, the overall performance analysis is given: the distribution of $N_i$ is derived in Section VI-A and how the design parameters affect the performance is given. We also analyze the Fisher information in Section VI-B.

A. Distribution of $N_i$

In this subsection, the distribution of the estimator $N_i$ is calculated under a framework where each consensus run is assumed to converge to the sample mean of the initial states potentially with some additive Gaussian noise. We also show how the design parameters affect the bias and the variance of the estimator.

1) Steady state distribution of $N_i$: According to Theorem 1 when consensus is reached, nodes in the network converge to a noisy version of the sample mean of the initial states. If this noise is approximated as Gaussian, the estimate of number of nodes at node $i$ can be calculated from the convergence result. The estimate at node $i$ can be expressed as

$$N_i = \frac{n_i' + \frac{1}{K} \sum_{l=1}^{K} n_l^{(k)} x_i^2}{\frac{1}{K} \sum_{l=1}^{K} n_l^{(k)} + \frac{1}{N} \sum_{l=1}^{K} r_l^{(k)} x_i},$$ (37)

where $n_i^{(k)}$ is the $k$th accumulated noise at node $i$ during phase I of the node counting algorithm mentioned in Section IV-B1 and $n'$ is the accumulated noise during phase II of the node counting algorithm as mentioned in Section V-B2. When $t \to \infty$, we have $n_i^{(k)} \sim N(0, \frac{1}{K N} \sum_{i=1}^{N} d_i \sigma_n^2)$ and $n' \sim N(0, \frac{1}{N} \sum_{i=1}^{N} d_i \sigma_n^2 \beta)$ from Theorem 1.

The following theorem characterizes the distribution of $N_i$ when $N$ and $K$ are large.

**Theorem 3.** For large $N$ and $K$, the probability density function of $N_i$, denoted as $p_{N_i}(z)$ is,

$$p_{N_i}(z) = \frac{b(z)d(z)}{\sqrt{2\pi \alpha^2(z)\sigma_1}\sigma_2} \left[ \Phi \left( \frac{a(z)}{\sigma_1} - \Phi \left( \frac{b(z)}{\sigma_1} \right) \right) - \Phi \left( \frac{-b(z)}{\sigma_2} \right) \right] + \frac{1}{\pi \alpha^2(z)\sigma_1\sigma_2} e^{-\frac{z^2}{2}},$$ (38)

where

$$a(z) = \sqrt{\frac{1}{\sigma_1^2} \mu_1^2 + \frac{1}{\sigma_2^2} \mu_2^2},$$

$$b(z) = \frac{\mu_1}{\sigma_1^2} \mu_1 + \frac{\mu_2}{\sigma_2^2} \mu_2,$$

$$c = \mu_1^2 \sigma_1^2 + \mu_2^2 \sigma_2^2,$$

$$d(z) = e^{-\frac{z^2}{2\sqrt{2\pi\alpha(z)\sigma_1}b(z)}},$$

$$\Phi(t) = \int_{-\infty}^{t} \frac{1}{\sqrt{2\pi}} e^{-u^2} du$$

and

$$\mu_1 = \frac{1}{N} \sum_{i=1}^{N} x_i^2,$$

$$\sigma_1^2 = \left(\frac{\sum_{i=1}^{N} d_i}{N^2}\right) \sigma_n^2 \beta,$$

$$\mu_2 = \frac{\sum_{i=1}^{N} x_i^2}{N^2} + \frac{\sum_{i=1}^{N} d_i}{N^2} \sigma_n^2 \beta,$$

$$\sigma_2^2 = \left(\frac{2}{K}\right) \left[ \frac{\sum_{i=1}^{N} x_i^2}{N^2} + \frac{\sum_{i=1}^{N} d_i}{N^2} \sigma_n^2 \beta \right]^2.$$

**Proof:** The proof relies on the ratio of two independent Gaussian random variables and details of the proof are given in Appendix.

The distribution of $N_i$ is given in Theorem 3. However, how the bias and variance of the estimator will be affected is not clear. In the following, we simplify the result of Theorem 3 by assuming equal initial values $x_i = a$ as in Section IV-C. The distribution of $N_i$ together with the bias and variance of the estimator is calculated.

2) Steady state distribution with $x_i = a$: When $x_i = a$, the estimate of number of nodes at node $i$ in (37) becomes:

$$N_i = \frac{\frac{1}{K} \sum_{l=1}^{K} n_l^{(k)} x_i^2}{\frac{1}{K} \sum_{l=1}^{K} n_l^{(k)} + \frac{1}{N} \sum_{l=1}^{K} r_l^{(k)} x_i},$$

$$= \frac{\frac{1}{K} \sum_{k=1}^{K} n_i^{(k)} + \frac{1}{N} \sum_{i=1}^{N} r_i^{(k)} a \sigma_n^2}{\frac{1}{K} \sum_{k=1}^{K} n_i^{(k)} + \frac{1}{N} \sum_{i=1}^{N} r_i^{(k)} a^2}.$$(39)
When \( N \) is large, from the non-identical central limit theorem (Lyapunov central limit theorem), we have,
\[
\frac{1}{N} \sum_{i=1}^{N} r_i^{(k)} a \sim \mathcal{N} \left( 0, \frac{\sigma^2}{N} \right),
\]
(40)
and \( \hat{N}_i \) is scaled inverse-chi-square distributed, and the probability density function can be expressed as,
\[
p_{\hat{N}_i}(z) = \left( \frac{(\nu^2\mu/2)^{\nu/2}}{\Gamma(\nu/2)} \right) \left( \frac{e^{-\mu z/2}}{2^{\nu/2}} \right) \left( 1 + \frac{\mu z}{2} \right)^{-\nu/2}.
\]
(41)

The mean of \( \hat{N}_i \) is,
\[
E[\hat{N}_i] = \frac{\nu \tau^2}{\nu - 2},
\]
(43)
\[
= N + \frac{2N}{K - 2} - \left( \frac{K}{K - 2} \right) \left( \frac{N \sigma_i^2 \beta \sum_{i=1}^{N} d_i}{N \sigma_i^2 + \sigma_i^2 \beta \sum_{i=1}^{N} d_i} \right),
\]
(44)
\[
= N + \frac{2N}{K - 2} - \left( \frac{K}{K - 2} \right) \left( \frac{N \beta \sum_{i=1}^{N} d_i}{N \text{SNR} + \beta \sum_{i=1}^{N} d_i} \right),
\]
(45)
where \( \text{SNR} := \frac{1}{N} \sum_{i=1}^{N} x_i^2 = \frac{a^2}{\sigma^2} \). From the steady state mean of \( \hat{N}_i \) as in equation (44) and (45), we have the following conclusions: i) The bias will be small for large \( K \) and \( \text{SNR} \); ii) \( E[\hat{N}_i] = N \) when we have,
\[
a^2 = \frac{(K - 2) \sigma_i^2 \beta \sum_{i=1}^{N} d_i}{2N}.
\]
(46)
However, equation (46) depends on \( N \) and \( \sum_{i} d_i \) which are usually unknown at nodes in practice; and iii) When \( K \) and \( N \) are large and \( K \gg N \), we have the approximation,
\[
E[\hat{N}_i] = N - \frac{\sigma_i^2 \beta (\sum_{i} d_i)}{a^2} = N - \frac{\beta \sum_{i} d_i}{\text{SNR}}.
\]
(47)
Equation (47) indicates that for large \( K \), the bias in \( \hat{N}_i \) given by \( E[\hat{N}_i - N] \) is always negative and can be made small at large SNR. Also note that equation (47) shows that smaller \( \sum_{i} d_i \) results in a smaller bias when \( t \rightarrow \infty \). However, this does not imply that a less connected graph would behave better in real applications where the algorithm needs to be stopped in finite iteration time. This is because the convergence speed of the algorithm is also related to the connectivity of the graph as shown in Section VII(A). Therefore, the required stopping time will be less for a more connected graph and the accumulated noise at nodes may also be smaller.

The variance of \( \hat{N}_i \) is,
\[
\text{Var}[\hat{N}_i] = \frac{2\nu^2 \tau^4}{(\nu - 2)^2 (\nu - 4)}.
\]
(48)
\[
= \frac{2K^2}{(K - 2)^2 (K - 4)} \left( \frac{N \sigma_i^2 a^2}{N \sigma_i^2 + \sigma_i^2 \beta \sum_{i=1}^{N} d_i} \right)^2.
\]
(49)
\[
= \frac{2K^2}{(K - 2)^2 (K - 4)} \left( \frac{N \text{SNR}}{N \text{SNR} + \beta \sum_{i=1}^{N} d_i} \right)^2.
\]
(50)
Note that the variance is inversely proportional to \( K \) and \( K \text{Var}[\hat{N}_i] \sim 2 \left( \frac{N \sigma_i^2 \beta \sum_{i=1}^{N} d_i}{N \text{SNR} + \beta \sum_{i=1}^{N} d_i} \right)^2 \) for large \( K \).

The probability density function of \( \hat{N}_i \) provides guideline towards how to choose the design parameters. For example, from equation (45) and (47) we can see that when the SNR is small, we should choose the sum of squares of the step size to be small (\( \beta \) to be small) to make the bias small.

### B. Fisher Information

Fisher information in the absence of communication noise is discussed in Section VII-B1 and the Fisher information in the presence of communication noise is calculated in Section VII-B2 to pinpoint the effect of noise. How the design parameters affect the Fisher information and Cramer-Rao bounds is also presented. Finally, the conditions under which \( \hat{N}_i \) is asymptotically efficient (achieves the CRB) is discussed.

1) Absence of communication noise: First we discuss the Fisher information in the absence of communication noise. For nodes in the network, the \( k \)th element in the state vector converge by the central limit theorem to
\[
y_i^{(k)}(t) = \frac{1}{N} \sum_{i=1}^{N} r_i^{(k)} x_i \sim \mathcal{N} \left( 0, \frac{\sigma^2}{N} \right),
\]
(51)
where \( \sigma^2 = \frac{1}{N} \sum_{i=1}^{N} x_i^2 \) and we assume that \( \sigma^2 \) does not depend on \( N \). After a straightforward calculation, the Fisher information is given by
\[
\mathcal{I}(N) = E \left( \frac{\partial}{\partial N} \ln g(X; N)^2 \right) \mid N = \frac{1}{2N^2}.
\]
(52)
where \( g(X; N) \) is the probability density function of Gaussian distribution as in equation (51). From equation (52), we can conclude that when \( N \) is large, the choice of \( x_i \) will not affect the Fisher information.

Note that equation (52) is the Fisher information for one consensus run. For \( K \) consensus runs, the Fisher information will be \( K/2N^2 \). The Cramer-Rao bound is the inverse of Fisher information. Therefore, for \( K \) consensus run, a lower bound on the estimation variance for any unbiased network size estimator can be expressed as,
\[
\text{Var} \left[ \hat{N}_i \right] \geq \frac{2N^2}{K}.
\]
(53)
2) Presence of communication noise: In the presence of communication noise, elements in the state vectors converge to a noisy version of the sample mean of the initial states. We use central limit theorem to approximate the average consensus results with Gaussian distribution. The \( k \)th element in the state vector can be expressed as,
\[
y_i^{(k)}(t) = \frac{1}{N} \sum_{i=1}^{N} r_i^{(k)} x_i + n_i^{(k)} \sim \mathcal{N} \left( 0, \frac{\sigma^2}{N} + \sigma_e^2 \right),
\]
(54)
where \( n_i^{(k)} \) is the accumulated noise at node \( i \). We assume that \( n_i^{(k)} \) is Gaussian distributed, \( n_i^{(k)} \sim \mathcal{N}(0, \sigma_e^2) \). Note that the noise term \( n_i^{(k)} \) can be viewed as a more general error caused...
by communication noise and lack of convergence due to finite number of iterations.

Let \( n_i^{(k)} \) capture the error caused by communication noise and lack of convergence. If we assume \( \sigma_i^2 \) is not a function of \( N \), the Fisher information can be calculated as,

\[
\mathcal{I}(N) = E \left[ \left( \frac{\partial}{\partial N} \ln g(X; N) \right)^2 \right] \left( \frac{\sigma_i^4}{\sigma_i^2 + \sigma_i^2} \right) \quad (55)
\]  

\[
= \left( \frac{1}{2N^4} \right) \left( \frac{\sigma_i^4}{\sigma_i^2 + \sigma_i^2} \right)^2 \quad (56)
\]

\[
\approx \left( \frac{1}{2N^4} \right) \sigma_i^2 \quad (57)
\]

\[
= \left( \frac{1}{2N^4} \right) \text{SNR}^2, \quad (58)
\]

where \( g(X; N) \) is the Gaussian probability density function and the distribution is given in equation (54). Equality in (57) holds when \( N \) is large. By comparing Equation (52) and (58), we can observe that in the absence of noise, the Fisher information behaves like \( O(1/N^2) \), while in the presence of noise it behaves like \( O(1/N^4) \) if the SNR does not depend on \( N \). The Fisher information in equation (58) also shows that the SNR affects the performance.

If we assume that the consensus is reached and the error \( n_i^{(k)} \) is caused by Gaussian noise, then \( n_i^{(k)} \) is Gaussian distributed, \( n_i^{(k)} \sim N \left( 0, \left( \sum_{i=1}^{N} \frac{d_i}{K} \right) \sigma_i^2 \right) \). From Theorem 1 and it depends on \( N \). We have an interesting finding that if the nodes in the network have same degree \( d_i = d \), then \( \sigma_i^2 = \frac{d \sigma_i^2}{N K} \). We have \( y_i^{(k)}(t) = \left( \frac{1}{N} \sum_{i=1}^{N} r_i^{(k)} x_i + n_i^{(k)} \right) \sim N \left( 0, \frac{\sigma_i^2 + d \sigma_i^2}{N K} \right) \), and the Fisher information \( \mathcal{I}(N) = 1/(2N^2) \) which is same as equation (54). This is because after defining \( \sigma_i^2 = \sigma_i^2 + d \sigma_i^2 \), the Fisher information calculation will be same as in equation (55). The above result suggests that if all nodes have the same degree and consensus is reached, the SNR will not affect the Fisher information.

Finally, we compare the estimation variance in equation (50) with the Cramer-Rao bound in equation (53). For large SNR, the estimation bias in equation (47) is negligible so that

\[
\text{Var} [ \hat{N} ] = \frac{2K^2}{(K-2)^2(K-4)} \left( \frac{N^2 \text{SNR}}{N \text{SNR} + \beta \sum_{i=1}^{N} d_i} \right)^2 \approx \frac{2K^2 N^2}{(K-2)^2(K-4)}. \quad (59)
\]

The estimation variance in equation (59) is always larger than the CRB result in equation (52), and it achieves the CRB, in the sense that the ratio of the right hand side of (53) to (59) converges to 1 as \( K \to \infty \). This shows that the proposed estimator is asymptotically efficient.

VII. SIMULATION RESULTS

A. Convergence of the Algorithm

The graph of the sensor network is fixed as shown in Figure 1 with \( N = 75 \). In Figure 2(a) - 2(c) we set \( K = 1000 \), noise variance \( \sigma_n^2 = 1 \) and \( \alpha(t) = 0.1/(t+1) \). In Figure 2(a) the initial values \( x_i \) are fixed and generated from a Gaussian distribution with zero mean and variance 25, and \( r_i^{(k)} \) are Bernoulli distributed with \( \pm 1 \) so that \( \text{SNR} = 13.98 \text{ dB} \). In Figure 2(b) the initial values are set to be fixed to \( x_i = 5 \) and \( r_i^{(k)} \) are Bernoulli distributed with \( \pm 1 \), and \( \text{SNR} = 13.98 \text{ dB} \). In Figure 2(c) the initial values are set to be fixed \( x_i = 5 \) and \( r_i^{(k)} \) are chosen as \( r_i^{(k)} \sim N(0, 1) \), and \( \text{SNR} = 13.98 \text{ dB} \). In Figure 2(a) node counting algorithm described in Section IV-B is used, and method in Section IV-C is used in the simulations in Figure 2(b) and 2(c). From Figure 2(a) - 2(c) we see that the number of nodes can be estimated using the proposed node counting algorithm in the presence of communication noise.

In Figure 3 the mean square error for \( \hat{N}(t) \), denoted as \( \text{E} \left[ \left( \hat{N}(t) - N \right)^2 \right] \), is plotted. The graph is the same as in Figure 1 with \( N = 75 \) and \( x_i \) and \( r_i^{(k)} \) are chosen to be different values as shown in the figure. We assume noisy communication with \( \sigma_n^2 = 1 \) and \( K = 1000 \). From Figure 3 we can see that in the presence of communication noise, larger \( x_i \) values result in a better performance since the signal to noise ratio is larger. We can also see from Figure 3 that for the same \( x_i \) value, MSE are almost the same for different \( r_i^{(k)} \) distributions. This is because that when \( N \) is large, equation (51) and (53) are almost equal.

Note that existing consensus-based algorithms in the literature, such as the Bernoulli trials method in [17] and uniform + maximum + ML algorithm in [20] works in the absence of communication noise. However those algorithms are sensitive to communication noise: the uniform + maximum + ML algorithm fails to converge with noise since max consensus diverges in the presence of communication noise [13]. and Bernoulli trials method is sensitive to communication noise since least common multiple (LCM) function is used in the algorithm and LCM results fluctuate in the presence of noise.

B. PDF of \( \hat{N} \)

In Figure 4 the probability density function of \( \hat{N} \) is plotted based on equation (41). The network is the same as Figure 1 in Figure 4(a) we fix the SNR and the figure shows how the value of \( K \) affects the distribution of \( \hat{N} \). While in Figure 4(b) we fix
System size estimation at different nodes

(a) Entries of node counting result versus number of iterations \( t \). \( x_i(0) \sim \mathcal{N}(0, 25), \sigma_n^2 = 1 \) and \( r_i^{(k)} \) Bernoulli distributed with \( \pm 1. \alpha(t) = 0.1/(t + 1) \) and \( K = 1000 \).

(b) Entries of node counting result versus number of iterations \( t \). \( x_i(0) = a = 5, \sigma_n^2 = 1 \) and \( r_i^{(k)} \) Bernoulli distributed with \( \pm 1. \alpha(t) = 0.1/(t + 1) \) and \( K = 1000 \).

(c) Entries of node counting result versus number of iterations \( t \). \( x_i(0) = a = 5, \sigma_n^2 = 1 \) and \( r_i^{(k)} \) \( \sim \mathcal{N}(0, 1) \). \( \alpha(t) = 0.1/(t + 1) \) and \( K = 1000 \).

Fig. 2: Simulation Results for Node Counting Algorithm for Graph with \( N = 75 \).

\( K \) and the figure shows how the SNR affects the distribution of \( \hat{N} \).

From Figure 4(a), we can conclude the bias and the variance result in Section VI-A2. When \( K \) gets larger, the bias and variance of the estimator get smaller. From Figure 4(b), we see that when the SNR is larger, the bias of the estimator gets smaller, however the variance of the estimator gets larger. We also see from Figure 4(b) that for fixed \( K \) value and large enough SNR, the distribution of \( \hat{N} \) will almost be the same as SNR increases (by comparing the probability density function for SNR = 13.01dB and SNR = 13.98dB).

C. Special Initial Values \( x_i \) as in (36)

In Figure 5, a special case mentioned in Section V-C is considered. The initial values of \( x_i \) are chosen as in equation (36) and we assume the absence of noise, that is, \( \sigma_n^2 = 0 \).
The design parameters \( r_i^{(k)} \) are chosen to be Bernoulli and Gaussian distributed respectively.

From the figure we have the following observations: i) In the absence of communication noise, the MSE achieves 0 as \( t \to \infty \) if \( r_i^{(k)} \) is Bernoulli distributed; and ii) The MSE will be small when \( r_i^{(k)} \) is Gaussian distributed, but does not achieve 0.

However, choosing the initial values as in equation \( 30 \) is difficult in practice (especially in the presence of noise) since this is not a distributed way to choose the initial values, as mentioned in Section V-C.

### D. Bernoulli \( r_i^{(k)} \) outperforms Gaussian \( r_i^{(k)} \)

In this subsection, a network of 4 nodes with star topology is considered. In Figure 6 we set \( x_i = 0 \), \( r_i^{(k)} \) are chosen to be Gaussian and Bernoulli distributed and \( \sigma_n^2 = 1 \). MSE versus the number of iterations is plotted.

We see from the Figure 6 that when \( N \) is small, choosing \( r_i^{(k)} \) to be Bernoulli yields a better performance since the variance of the \( L_2 \) norm estimation result will be smaller from equation \( 32 \) and \( 33 \).

### VIII. DISCUSSION AND CONCLUSIONS

A practical approach for reliable estimation of the number of nodes over autonomous distributed sensor networks in the presence of communication noise is proposed. \( L_2 \) norm estimation is used, together with the average consensus algorithm. Different sources of error is described, and we show there is a trade-off between the estimation accuracy and the storage at sensor nodes. The Fisher information about the estimate of number of nodes in the network is calculated. How the noise and initial values at nodes affect the Cramer-Rao bound is shown. The distribution of the final estimator is also calculated to show how the design parameter affect the estimation performance. The results provide guideline towards algorithm parameter settings to trade-off between more accurate network size estimation and required storage at nodes.

Compared to existing works in the literature, there are both advantages and disadvantages to our algorithm. The proposed approach is fully distributed: nodes do not need to be labeled and they communicate only with their neighbors. Moreover, the proposed algorithm is robust to communication noise. However, there are also disadvantages. For example, nodes in the network need to have a large storage (large \( K \) value) to obtain an accurate estimate of total number of nodes in the network. Another disadvantage is that the convergence time will be long compared to max consensus based algorithms since average consensus is used. However, max consensus based algorithms are not suitable in the presence of noise since max consensus diverges with noise. Note that methods to accelerate average consensus such as optimal step size can be used to accelerate the proposed distributed node counting algorithm.

### APPENDIX

#### Proof of Theorem 3

In this proof, we will first use central limit theorem to approximate the distribution of the denominator and numerator of equation \( 37 \) using Gaussian distribution. Then, the distribution of \( \hat{N} \) is obtained by using the ratio distribution results in \( 30 \) and \( 31 \).

First, let the numerator in equation \( 37 \)

\[
\left( n' + \frac{1}{N} \sum_{i=1}^{N} x_i^2 \right) = A. \quad \text{Since } x_i \text{ are constants and } n' \sim \mathcal{N} \left( 0, \frac{1}{N^2} \sum_{i=1}^{N} d_i \sigma_n^2 \beta \right), \text{ the numerator is Gaussian distributed, we have,}
\]

\[
A \sim \mathcal{N} \left( \frac{1}{N} \sum_{i=1}^{N} x_i^2, \frac{1}{N^2} \left( \sum_{i=1}^{N} d_i \right) \sigma_n^2 \beta \right). \quad (60)
\]

We use central limit theorem to calculate the distribution of the denominator of \( 37 \). Since \( r_i^{(k)} \) are i.i.d. random variables with mean 0 and variance 1, we can use central limit theorem to approximate the term:

\[
\frac{1}{N} \sum_{i=1}^{N} r_i^{(k)} x_i \sim \mathcal{N} \left( 0, \frac{1}{N} \sum_{i=1}^{N} x_i^2 \right). \quad (61)
\]

Also, we know that \( n_i^{(k)} \sim \mathcal{N} \left( 0, \frac{1}{N} \sum_{i=1}^{N} d_i \sigma_n^2 \beta \right) \), as a result, we have,

\[
\left( n_i^{(k)} + \frac{1}{N} \sum_{i=1}^{N} r_i^{(k)} x_i \right) \sim \mathcal{N} \left( 0, \frac{\sum_{i=1}^{N} x_i^2 + \left( \sum_{i=1}^{N} d_i \right) \sigma_n^2 \beta}{N^2} \right). \quad (62)
\]
Therefore, the denominator of (37) is the sample mean of the square of $K$ i.i.d. Gaussian random variables. Note that square of Gaussian distribution is scaled chi-squared distribution with degrees of freedom equals to 1, and its mean equals to $\left( \sum_{i=1}^{N} x_i^2 \right) / N^2$ and variance equals to $\left( \sum_{i=1}^{N} x_i^2 \right) / N^2$. By using central limit theorem, we can approximate the denominator of (37) with Gaussian distribution, let $\frac{1}{N} \sum_{i=1}^{K} n_i^{(k)} + \frac{1}{N} \sum_{i=1}^{N} r_i^{(k)} x_i = B$, we have,

$$B \sim N(\mu_B, \sigma_B^2)$$

$$\mu_B = \frac{\sum_{i=1}^{N} x_i^2 + \sum_{i=1}^{N} d_i \sigma_i^{2/\beta}}{N^2}$$

$$\sigma_B^2 = \frac{2}{K} \frac{\sum_{i=1}^{N} x_i^2 + \sum_{i=1}^{N} d_i \sigma_i^{2/\beta}}{N^2}.$$

Finally, since both numerator and denominator are Gaussian distributed as shown in equation (60) and (63). The results for Gaussian ratio distribution proposed in [11 eqn(1)] can be used to calculate the distribution for $\bar{N}$ in equation (37), and Theorem 3 is proved. Note that the numerator and denominator are independent since the noise are i.i.d. and $x_i$ are constants.

REFERENCES


