Distributed Node Selection for Sequential Estimation over Noisy Communication Channels

Thakshila Wimalajeewa and Sudharman K. Jayaweera
Dept. of Electrical and Computer Engineering
University of New Mexico
Albuquerque, NM 87131-0001, USA
Email: {thakshilaw,jayaweera}@ece.unm.edu

Abstract

This paper proposes a framework for distributed sequential parameter estimation in wireless sensor networks. In the proposed scheme, the estimator is updated sequentially at the current node with its new measurement and the noisy corrupted estimator from the previous node. Since all nodes in the network may not carry useful information, methodologies to find the optimal set of nodes and the corresponding node ordering for the sequential estimation process are investigated. It is shown that the determining the optimal set of nodes that leads to the globally optimal performance is computationally complex when the network size is large. We develop two distributed greedy type node selection algorithms with reduced computational and communication complexities. In these algorithms, the next best node is selected at the current node such that it optimizes a certain objective function. It is shown that the performance of both proposed greed type schemes leads to exact, or close to exact, results to the optimal scheme computed via forward dynamic programming, under certain conditions. Moreover, contrast to existing methodologies, our work considers the node selection and inter-node communication noise jointly in the sequential estimation process.

I. Introduction

Distributed parameter estimation is one of the common tasks in many wireless sensor network (WSN) applications. Severe resource constraints such as node power and communication bandwidth in WSNs call for efficient algorithms for estimation that combine estimation accuracy with optimal resource consumption. In a static parameter estimation problem, the required parameter is to be estimated based on noise corrupted observations at local nodes. In most sensor network applications considered in the literature it is assumed that the spatially separated sensor nodes send their locally processed information to a fusion center to form the final decision or the estimator [1]–[4]. In some sensor network applications, however,

[1]This research was supported in part by the National Science foundation (NSF) under the grant CCF-0830545.

it is required that any distributed node has the ability to form the final decision or the estimator by collaborating with other nodes in the network. Distributed sequential estimation, in which nodes update the local estimators sequentially, is one way of achieving collaborative estimation without depending on a central fusion center [5], [6]. In a sequential estimation process, the estimator is updated sequentially with the information residing at the current node and the information received from the previous node. Contrast to estimation via consensus in which each node has to agree with the final estimator, in sequential estimation any node in the sequential estimation process has the ability to declare the final estimator if the desired performance level is reached. Since all nodes in the network may not have useful information regarding the Phenomenon of Interest (PoI), it is required to select nodes which carry useful information such that the sequential estimation process will be terminated with the minimum number of nodes. This ensures that only the nodes that contribute to the final estimation need to be in active mode while others may remain idle preserving their transmit energy.

The distributed sequential estimation problem was addressed in previous work [5]–[7]. According to [6], [7], a lead node sequentially queries sensor nodes and updates its estimator (based on the posterior distribution of the state of the PoI) until a desired performance level is reached. In these schemes, the lead node has to keep track of all nodes which have been participated in the estimation process at each step. In [5], the posterior distribution (belief) at the current node is transmitted to the next node where it updates the state of belief based on the current belief and the new measurement at that node. Note that in this scheme, if the belief (posterior distribution) cannot be represented by a parameterizable distribution, grid samples of the distribution should be transmitted to the next node leading to a considerable communication burden. However, when the belief cannot be represented by a standard parameterizable distribution, [5] proposed to approximate the belief by a parameterizable distribution and the corresponding parameters are transmitted to the next node. The communication complexity in transmitting belief then is determined by the number of parameters and their dimensions. To find the next best node in the sequential estimation process, several objective functions incorporated with information utility measures based on entropy and the network geometry were proposed in [7]. In [8] a node selection algorithm for target tracking based on the posterior Cramer-Rao Lower Bound (CRLB) was presented. In [9], a sequential data processing at the fusion center for estimation of multiple random sources is presented in which the node ordering is based on the mutual information. However, neither of these considered the noise in inter-node communication links and the node selection together. In [10], the sequential estimation of a non-random parameter over noisy correlated channels was considered. However, it did not consider the best ordering of the nodes for the sequential processing.

In this paper, we consider the distributed sequential estimation of a random static parameter in which the estimator is updated at each node sequentially based on its own measurement and the noisy corrupted estimator received from the previous node. In proposed schemes, each node in the estimation process

needs to transmit only two parameters, namely the updated estimator and the corresponding minimum mean squared error (MMSE). We propose two greedy algorithms to find the ordering of nodes for the estimation process such that the current node selects the next node as the one which maximizes a certain reward function. This reward function is assumed to be a combination of the information utility measure and the communication cost incurred between the current node and the next node. The two schemes are different from each other in terms of the search space; global search or a local search. We propose to use mutual information as the information utility measure and investigate the use of MMSE of the estimator as an alternative when it is difficult to compute the mutual information. Note that, the mutual information utility measure selects the node that provides the maximum amount of new information regarding the PoI given the current estimate, as the next processing node. In the global search based scheme, we assume that any two nodes in the network can communicate with each other and the next node is searched over all possible unvisited nodes to maximize the relevant reward function. In the scheme based on the local search, on the other hand, we assume that each node has a set of neighbors that it can communicate at an affordable communication cost. The candidate next nodes at each node are allowed to be selected only from these neighbors. The information utility measures of the candidate nodes are computed according to the current node's observation and the knowledge of positions of the neighboring sensor nodes and the target. In the proposed scheme based on local search, each node has to keep track of only its neighbors to determine which nodes have been participated in the estimation process, while in the scheme with global search, each node has to keep track of all unvisited nodes in the whole network.

From simulation results we see that the performance of the proposed scheme with local approach becomes closer to that with the global approach after processing a relatively small number of nodes. However, it should be noted that, with proposed greedy type node selection schemes a global optimal MMSE solution is not guaranteed. We further develop an algorithm to find the optimal sequence of nodes (with best ordering) which yields the global MMSE, based on forward dynamic programming with a higher computational complexity compared to that with greedy type algorithms. We compare the two proposed schemes with the global optimal solution obtained via dynamic programming and show that the performances of both proposed greedy type schemes are getting closer to that with the optimal scheme with a relatively small number of processing nodes. Moreover, the proposed greedy type node selection schemes result a significant performance improvement compared to that with the nearest node selection method in the context of number of nodes required to achieve a desired performance level.

The remainder of this paper is organized as follows: Section II presents the sensor network and observation models and formulates the distributed sequential estimation problem over noisy communication channels. In Section III, the sequential MMSE performance is evaluated. The node selection schemes are discussed in the Section IV. Section V presents the performance results of the noisy sequential estimation process with the node selection schemes. Final concluding remarks are given in Section VI.

II. SENSOR NETWORK MODEL

Consider a spatially distributed, sensor network consisting of n number of nodes. Denote by s_k the k-th node, for $k = 1, \cdots, n$. Note that, when there is no ambiguity, we use s_k and k to denote the k-th processing node interchangeably. The network is deployed to estimate the signal amplitude emitted by a possible target (e.g. a sound source) based on the following observation model at node s_k:

$$w'_k = \frac{\theta}{\|\mathbf{x}_k - \mathbf{x}_t\|^{\alpha/2}} + v'_k, \text{ for } k = 1, \cdots, n, \tag{1}$$

where θ is the parameter to be estimated (target amplitude) that is assumed to be Gaussian with zero mean and variance σ_θ^2, $\mathbf{x}_k$ and $\mathbf{x}_t$ denote the positions of sensor node s_k and the target, respectively, v'_k is the measurement noise that is assumed to be white Gaussian with zero mean and variance σ_0^2 and α is the path loss exponent that is determined by the propagation environment. This model can be used, for example, in applications in which acoustic sensors are used to estimate the amplitude of sound signals emitted by a target [7], [11]. By rearranging (1), we can re-write the observation at node s_k in the equivalent form of

$$w_k = \theta + v_k, \text{ for } k = 1, \cdots, n, \tag{2}$$

where now v_k is assumed to be independent but not identically distributed. In particular, v_k is Gaussian with mean zero and variance $\sigma_k^2 \propto \|\mathbf{x}_k - \mathbf{x}_t\|^\alpha$.

The idea is to estimate the parameter θ sequentially via inter node communication. Let s_1 be the starting node of the sequential estimation process. The starting node estimates the parameter based on its own observation, $z_1 = w_1 = \theta + v_1$. Denote $\hat{\theta}_1(z_1)$ to be the estimator at node s_1. The sequential estimation process is continued until either the desired performance level is reached or observations at all nodes are processed. When $k = 1$, if the MMSE does not meet the desired performance, the estimator $\hat{\theta}_1$ is transmitted to the next node, selected based on a certain criteria, over a noisy channel. The criteria for selection of next node is discussed in a later section. For $k > 1$, the k-th node estimates the parameter θ based on its own observation and the received estimator from the $(k-1)$-th node. The effective observation vector at node s_k (for $k > 1$) is

$$\mathbf{z_k} = \begin{bmatrix} w_k \\ y_k \end{bmatrix} = \begin{bmatrix} \theta + v_k \\ \hat{\theta}_{k-1} + n_k \end{bmatrix}, \text{ for } \quad k = 2, \cdots, n, \tag{3}$$

where y_k is the noise corrupted estimator from node s_{k-1}. The channel noise n_k, from node s_{k-1} to node s_k is assumed to be independent Gaussian with mean zero and variance $\sigma^2_{c(k-1,k)} \propto \|\mathbf{x}_k - \mathbf{x}_{k-1}\|^{\alpha'}$ for $k = 2, \cdots, n$ where α' is the path loss index of communication channels between nodes.

III. Sequential MMSE Performance

For $k = 1$, assuming that the parameter θ is independent of observation noise v_1, the optimal Minimum Mean Squared Error (MMSE) estimate at node s_1 based on z_1 is given by

$$\hat{\theta}_1(w_1) = \frac{\sigma_\theta^2}{\sigma_\theta^2 + \sigma_1^2} w_1, \tag{4}$$

and the corresponding MMSE, denoted by M_1, of the estimator (4) is

$$M_1 = \frac{\sigma_1^2 \sigma_\theta^2}{\sigma_1^2 + \sigma_\theta^2} = \left(\frac{1}{\sigma_1^2} + \frac{1}{\sigma_\theta^2} \right)^{-1}. \tag{5}$$

Equivalently, (4) can be expressed as $\hat{\theta}_1(w_1) = \frac{M_1}{\sigma_1^2} w_1$.

For $k > 1$, the MMSE estimator at node s_k is computed based on the observation vector (3).

Lemma 1: The MMSE estimator $\hat{\theta}_k(w_k, y_k)$ and the corresponding MMSE M_k at node s_k for $k > 1$ are given by,

$$\hat{\theta}_k(w_k, y_k) = \frac{M_k}{\sigma_k^2} w_k + \frac{M_k(\sigma_\theta^2 - M_{k-1})}{M_{k-1}(\sigma_\theta^2 - M_{k-1}) + \sigma_\theta^2 \sigma_{c(k-1,k)}^2} y_k, \tag{6}$$

and

$$M_k = \frac{\sigma_\theta^2}{\sigma_\theta^2 d_{k-1,k}^2 + 1}, \tag{7}$$

respectively, where $d_{k-1,k}^2 = \frac{1}{\sigma_k^2} + \frac{(\sigma_\theta^2 - M_{k-1})^2}{\sigma_\theta^2 \left[M_{k-1}(\sigma_\theta^2 - M_{k-1}) + \sigma_\theta^2 \sigma_{c(k-1,k)}^2 \right]}$ and M_{k-1} is the MMSE at the node s_{k-1} that is assumed to be available at node s_k.

Proof: (Proof of Lemma 1) Given θ, we can show that y_k and w_k are distributed as,

$$y_k|\theta \sim \mathcal{N}\left(\frac{(\sigma_\theta^2 - M_{k-1})}{\sigma_\theta^2} \theta, \frac{M_{k-1}(\sigma_\theta^2 - M_{k-1}) + \sigma_\theta^2 \sigma_{c(k-1,k)}^2}{\sigma_\theta^2} \right) \tag{8}$$

and $w_k|\theta \sim \mathcal{N}(\theta, \sigma_k^2)$, respectively, where we use $X \sim \mathcal{N}(\mu, \sigma^2)$ to denote that the random variable X is distributed as Gaussian with mean μ and the variance σ^2. Since given θ, y_k and w_k are independent, the effective observation vector at the node s_k, $\mathbf{z}_k$ is distributed as, $\mathbf{z}_k|\theta \sim \mathcal{N}(\boldsymbol{\mu}_k \theta, \boldsymbol{\Sigma}_k)$, where $\boldsymbol{\mu}_k = \begin{bmatrix} 1 & \frac{(\sigma_\theta^2 - M_{k-1})}{\sigma_\theta^2} \end{bmatrix}^T$ and $\boldsymbol{\Sigma}_k = \begin{bmatrix} \sigma_k^2 & 0 \\ 0 & \frac{M_{k-1}(\sigma_\theta^2 - M_{k-1}) + \sigma_\theta^2 \sigma_{c(k-1,k)}^2}{\sigma_\theta^2} \end{bmatrix}$. From [12] (page 150-151), it can be shown that the posterior density for θ, $p(\theta|\mathbf{z}_k)$ is given by,

$$\theta|\mathbf{z}_k \sim \mathcal{N}\left(\tilde{\mu_k}, \tilde{\sigma}_k^2 \right), \tag{9}$$

where $\tilde{\sigma}_k^2 = \left(d_{k-1,k}^2 + \frac{1}{\sigma_\theta^2} \right)$ with $d_{k-1,k}^2 = \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}_k^{-1} \boldsymbol{\mu}_k = \frac{1}{\sigma_k^2} + \frac{(\sigma_\theta^2 - M_{k-1})^2}{\sigma_\theta^2 \left[M_{k-1}(\sigma_\theta^2 - M_{k-1}) + \sigma_\theta^2 \sigma_{c(k-1,k)}^2 \right]}$ and $\tilde{\mu_k} = \tilde{\sigma}_k^2 \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}_k^{-1} \mathbf{z}_k$. Then the MMSE estimator of θ is given by,

$$\hat{\theta}_k(w_k, y_k) = \mathbb{E}\{\theta|\mathbf{z}_k\} = \tilde{\mu_k}$$

and the MMSE, M_k is given by, $M_k = \mathbb{E}\{Var(\theta|\mathbf{z}_k)\} = \tilde{\sigma}_k^2$. After manipulating, we get MMSE estimator and the corresponding MMSE as given by (6) and (7). ■

From (6), it can be seen that the MMSE estimator at node s_k is determined by its own observation, information from the node s_{k-1} and the channel noise quality. The MMSE at the s_k-th node is determined only by statistics of observations and channel noise and it is reasonable to assume that they can be made available at neighbors [10]. Note that, MMSE (7) can be expressed alternatively as,

$$M_k = \frac{\sigma_\theta^2}{1 + U_k + T_{k-1,k}} \tag{10}$$

where $U_k = \frac{\sigma_\theta^2}{\sigma_k^2}$ is the received SNR at the node s_k, and $T_{k-1,k} = \frac{(\sigma_\theta^2 - M_{k-1})^2}{M_{k-1}(\sigma_\theta^2 - M_{k-1}) + \sigma_\theta^2 \sigma_{c(k-1,k)}^2}$ represents the effect of the estimator at the node s_{k-1} and the channel noise between nodes s_{k-1} and s_k. As expected, from (10) it can be seen that by combining the previous node's estimator always reduces the MMSE at the node s_k rather than node s_k makes the estimator from its own observation only.

Since MMSE M_k in (7) depends on the channel noise of inter-node communication links, it is interesting to examine the behavior of M_k with respect to the corresponding channel quality. We consider following two extremes: Channel quality is good such that $\sigma_{c(k-1,k)}^2 \to 0$ and channel quality is poor such that $\sigma_{c(k-1,k)}^2 \to \infty$ for $k = 2, 3, \cdots, n$. In the first case, we have

$$\lim_{\sigma_{c(k-1,k)}^2 \to 0} M_k = \frac{M_{k-1}}{1 + \frac{M_{k-1}}{\sigma_k^2}}, \text{ for } k = 2, 3, \cdots, n.$$

Therefore, it is seen that when $\sigma_{c(k-1,k)}^2 \to 0$, $M_k \leq M_{k-1}$ for all k. That is, by sending the node s_{k-1}'s estimator to the node s_k always improves the MMSE performance at node s_k compared to that with at node s_{k-1}. On the other hand, if inter-node communication channel quality is poor, we have

$$\lim_{\sigma_{c(k-1,k)}^2 \to \infty} M_k = \frac{\sigma_\theta^2 \sigma_k^2}{\sigma_\theta^2 + \sigma_k^2}, \text{ for } k = 2, \cdots, n. \tag{11}$$

That is, when the quality of inter-node communication link is poor, the performance at node s_k does not depend on the estimator at node s_{k-1}, but is entirely determined by the observation quality at node s_k. It implies that there will be a certain threshold value for channel quality of inter-node communication links which ensures that $M_k \leq M_{k-1}$ for $k = 2, 3, \cdots, n$. Indeed, it can be shown that if $\sigma_{c(k-1,k)}^2$ satisfies the following inequality for $k = 2, 3, \cdots, n$

$$\sigma_{c(k-1,k)}^2 \leq \frac{M_{k-1}^2(\sigma_\theta^2 - M_{k-1})}{\sigma_k^2(\sigma_\theta^2 - M_{k-1}) - M_{k-1}\sigma_\theta^2}, \tag{12}$$

then $M_k \leq M_{k-1}$; i.e. sending the estimator at node s_{k-1} to node s_k improves the MMSE performance at s_k. This is further discussed in Section V.

If we assume that the node observations are i.i.d and the inter-node communication is noiseless such that $\sigma_k^2 = \sigma_0^2$ and $\sigma_{c(k-1,k)}^2 = 0$ for $k = 2, 3, \cdots, n$, it can be shown that the MMSE at node s_k (7) reduces to $M_k = \frac{\sigma_\theta^2 \sigma_0^2}{\sigma_v^2 + k\sigma_\theta^2}$, which is a monotonically decreasing function of k. It is also interesting to see that in this case the minimum number of nodes n_{min} required to achieve a required MMSE performance level ϵ is given by, $n_{min} = \sigma_0^2 \left(\frac{1}{\epsilon} - \frac{1}{\sigma_\theta^2} \right)$.

In the case of i.i.d. observation noise such that $\sigma_k^2 = \sigma_0^2$ for all k, Fig. 1 shows the MMSE performance of the sequential estimation process with different channel noise qualities on inter-node communication links. In Fig. 1, we have let $\sigma_0^2 = 1$ and $\sigma_\theta^2 = 1$. In the special case when channel noise is also i.i.d. such that $\sigma_{c(k-1,k)}^2 = \sigma_c^2$ for all k, from Fig. 1 it can be seen that $M_k \leq M_{k-1}$ holds for all k. Moreover, as expected from (11) the MMSE performance converges to 0.5 as σ_c^2 increases. It is expected that when both observations and channel noise are i.i.d., the performance of the MMSE estimator is independent of the order of the processing nodes. Figure 1 also shows the performance of the MMSE estimator when channel noise is not identical (still the observation noise is i.i.d.). We have considered two cases: In the first, $\sigma_{c(k-1,k)}^2$'s are drawn randomly from a uniform distribution in $[0, 1]$ without any order. In the second case, these random $\sigma_{c(k-1,k)}^2$'s are arranged in an ascending order. From Fig. 1 it can be seen that whenever the condition (12) is satisfied at node k, $M_k \leq M_{k-1}$. In this case, to find the node where the minimum MMSE is achieved, the process should be continued for all nodes. On the other hand, in case 2, where nodes are selected with minimum distance from the current node, we observe that after a certain node the MMSE starts to monotonically increase. Therefore, it is enough to continue the sequential estimation process only until this specific node, thereby, saving the network power.

Figure 2 shows the MMSE performance of the sequential distributed estimation process with non-identical observations and channel noise. Dashed line corresponds to channel noise variance drawn from a uniform distribution without any order while the solid line corresponds to channel noise variance in ascending order with k. In both cases, the observation noise variances are drawn from a uniform distribution on $[0, 1]$. As can be observed from Fig. 2, when observations are not i.i.d., just selecting the nearest node as the next node does not always improve the performance. Therefore, when observations are not identical, it is required to have an information driven approach to select the nodes with higher information gain as well as lower communication cost.

IV. Sensor Node Selection

As can be seen from Fig. 2, when observation noise and the channel noise are both non identical, (which is the most realistic scenario) it is inefficient to continue the sequential estimation process without considering the node ordering or selection. This motivates us to find the optimal distinct ordered sequence of nodes which contains the minimum number of nodes required to reach a desired performance level. In the following we propose greedy type sequential algorithms to find the best ordering of nodes to achieve a desired performance level where each node in the estimation process determines its next best node based on a certain reward function. The reward function is considered to be a combination of an information utility measure, which reflects the information gain that can be achieved by selecting a node and the communication cost which accounts for the communication burden including bandwidth, latency, when communicating with the selected node.

A. Distributed node selection: global approach

In the following we determine the best ordering of nodes sequentially that would complete the estimation process by reaching at the desired performance level with a minimum number of processing nodes as a trade-off between the information gain and the communication cost. Let $\bar{S}_n = \{s_1, s_2, \cdots, s_n\}$ be the set of nodes in the network. Let $\mathcal{V}_j$ denote the set of nodes that have been participated in the sequential estimation process up to step j. Let $s_j \in \bar{S}_n$ be the selected processing node at step j. Then the next node s_{j+1} at step $(j+1)$ is chosen as,

$$s_{j+1} = \underset{s_k \in \mathcal{V}_j^c}{\operatorname{argmax}} \; R(s_j, s_k) \tag{13}$$

where $\mathcal{V}_j^c$ denotes the set complement of $\mathcal{V}_j$ with respect to $\bar{S}_n$ and the objective function $R(s_j, s_k)$ is defined as

$$R(s_j, s_k) = \beta R_I(\theta, w_k, y_{j,k}) + (1-\beta) R_c(s_j, s_k). \tag{14}$$

The first term in (14), $R_I(\theta, w_k, y_{j,k})$, represents a measure which reflects the information gain achieved by selecting node s_k when the current node is s_j and the second term in (14), R_c, accounts for the cost of communication between node s_j and s_k including bandwidth and latency. $\beta \in [0, 1]$ is a trade-off parameter that balances the contributions from the two terms in (14) and $y_{j,k} = \hat{\theta}_j + n_{j,k}$ is the received signal at node s_k if it is chosen to be the next node when the current node is s_j and $n_{j,k}$ is the channel noise between nodes s_j and s_k. The choice of β will depend on the required information gain and the tolerable communications cost. Note that when $\beta = 1$, the next node is selected as the one which provides the best information gain and which will provide a faster reduction of the estimation uncertainty. When $\beta = 0$, the next node is selected as the one which minimizes the communication cost while the information gain is not taken into account. Note that in this scheme, when the current processing node is s_j, the next best node is selected from the set of all unvisited nodes in the network up to step j.

There are several possible information utility measures that can be used to quantify the information gain provided by a sensor measurement. For example, [7], [5] provided a detailed description of entropy- and geometry-based information utility measures. In this paper, we consider two measures for information utility: (1). the conditional mutual information $I(\theta; w_{j+1} | y_{j,j+1} = \hat{\theta}_j + n_{j,j+1})$ which is considered to be the measure that provides the greatest amount of new information by selecting node s_{j+1} when the current estimate is $\hat{\theta}_j$ (2). the negative of MMSE $M_{j+1|j}$ at the s_{j+1}-th node, when the current node is s_j, for $s_j, s_{j+1} \in \mathcal{V}$. We explore the use of (-)MMSE as an alternate information measure if its computation is easier. In the following we find a relationship between the conditional mutual information and the sequential MMSE of our sequential estimation process.

MMSE and the conditional mutual information: The conditional mutual information between θ and w_k when selecting node s_k as the next node given that the current node is s_j, $I(\theta; w_k|y_{j,k} = \hat{\theta}_j + n_{j,k})$ is given by,

$$I(\theta; w_k|y_{j,k} = \hat{\theta}_j + n_{j,k}) = h(\theta|\hat{\theta}_j + n_{j,k}) - h(\theta|\hat{\theta}_j + n_{j,k}, w_k) \tag{15}$$

where $h(.)$ denotes the differential entropy [13]. Using (9), the second term in (15) can be shown as,

$$h(\theta|\hat{\theta}_j + n_{j,k}, w_k) = \frac{1}{2}\ln 2\pi e M_{j,k},\ nats \tag{16}$$

where $M_{j,k}$ is the MMSE at node s_k when the current node is s_j. From (8) and using results in [12] (pages 150-151), it can be shown that $\theta|y_{j,k} \sim \mathcal{N}\left(\frac{M_{j,k}(\sigma_\theta^2 - M_j)}{M_j(\sigma_\theta^2 - M_j) + \sigma_\theta^2\sigma^2_{c(j,k)}} y_{j,k},\ \frac{M_{j,k}\sigma_k^2}{\sigma_k^2 - M_{j,k}}\right)$. Then the first term in (15) is given by,

$$h(\theta|\hat{\theta}_j + n_{j,k}) = \frac{1}{2}\ln 2\pi e \frac{M_{j,k}\sigma_k^2}{\sigma_k^2 - M_{j,k}}\ nats. \tag{17}$$

From (16) and (17), we get the condition mutual information as,

$$I(\theta; w_k|y_{j,k}) = \frac{1}{2}\ln\left[\frac{\sigma_k^2}{\sigma_k^2 - M_{j,k}}\right]\ nats. \tag{18}$$

From (7), it can be shown that $M_{j,k} \leq \sigma_k^2$ for all $s_j, s_k \in \bar{\mathcal{S}}_n$ so that (18) is valid for all $s_j, s_k \in \bar{\mathcal{S}}_n$. We use natural logarithm for mutual information and differential entropy measures such that they are in *nats* and we omit writing the unit when there is no ambiguity.

In [14], authors present a relationship between input-output mutual information and the MMSE achievable by optimal estimation of the input given the output through an additive Gaussian noise channel for different variations (such as discrete time, continues time, scalar and vector Gaussian channel models). Similarly, it can be shown that the following relationship holds between the sequential MMSE at the node s_k when the current node is s_j, $M_{j,k}$ and the conditional mutual information $I(\theta; w_k|y_{j,k})$:

$$\frac{\partial I_{j,k}(U_k, T_{j,k})}{\partial U_k} = \frac{M_{j,k}}{2\sigma_\theta^2} \tag{19}$$

where $U_k = \frac{\sigma_\theta^2}{\sigma_k^2}$ and $T_{j,k} = \frac{(\sigma_\theta^2 - M_j)^2}{M_j(\sigma_\theta^2 - M_j) + \sigma_\theta^2\sigma^2_{c(j,k)}}$ as defined before, and we write $I_{j,k}$ to represent $I(\theta; w_k|y_{j,k})$.

We consider both measures MMSE and the mutual information as information utility measures in the reward function (14). When mutual information is used as the information utility measure, the first term $R_I(.)$ in (14) is given by,

$$R_I(\theta, w_k, y_{j,k}) = \frac{1}{2}\log\left[\frac{\sigma_k^2}{\sigma_k^2 - M_{j,k}}\right],\ j = 1, \cdots, n,\ k = 2, \cdots, n. \tag{20}$$

When MMSE is used as the information utility measure, we have

$$R_I(\theta, w_k, y_{j,k}) = -M_{j,k} = -\frac{\sigma_\theta^2}{\sigma_\theta^2 d_{j,k}^2 + 1},\quad j = 1, \cdots, n,\ k = 2, \cdots, n \tag{21}$$

where $d_{j,k}^2 = \frac{1}{\sigma_k^2} + \frac{(\sigma_\theta^2 - M_j)^2}{\sigma_\theta^2\left[M_j(\sigma_\theta^2 - M_j) + \sigma_\theta^2\sigma_{c(j,k)}^2\right]}$. The communication cost function between current node s_j and the possible next node s_k is taken to be $R_c(s_j, s_k) = \frac{1}{d_{max}}(\mathbf{x}_j - \mathbf{x}_k)^T(\mathbf{x}_j - \mathbf{x}_k)$ where d_{max} is the maximum distance between any two nodes in the network. Then the composite objective function (14) can be written as,

$$R(s_j, s_k) \;\; = \;\; \beta R_I(\theta, w_k, y_{j,k}) - \frac{(1-\beta)}{d_{max}}(\mathbf{x}_j - \mathbf{x}_k)^T(\mathbf{x}_j - \mathbf{x}_k), \tag{22}$$

where $R_I(\theta, w_k, y_{j,k})$ is as given in (20) or (21), for the mutual information or the MMSE, respectively. To find the next best processing node, the node s_j has to compute the reward function (22) for all candidate nodes in $\mathcal{V}_j^c$. At the worst case, node s_j has to compute the reward function for $n-1$ candidate nodes. On the other hand, estimation process may consist of n nodes at the worst case, thus resulting in a worst case computational complexity of order $O(n^2)$ for the whole network. Note that in this scheme, in general the computational complexity is much more reduced (compared to the worst case complexity) since as the process continues, the number of nodes to be queried by the current processing node is decreased. However, in this scheme each node s_j has to keep track of nodes that have already been participated in the estimation process up to step j, which requires a certain communication among nodes. Due to these factors, implementing this scheme distributively might be difficult. Thus, in the following, we propose a distributed algorithm for sensor node selection with reduced computational and communication complexities in which each node only needs to keep track of its neighboring nodes to perform the sequential estimation. As we will see with numerical examples, the performance of the proposed scheme based on local approach (in next subsection) shows closer performance to that with the global approach discussed in this subsection, after processing a small number of nodes.

B. Distributed node selection scheme: local approach

Assume that k-th node in the network has a set of neighbors $\mathcal{N}_k$ for $k = 1, \cdots, n$ where the neighbors are determined based on a node's effective communication range. We assume that each node has the same effective communication range, r_c, so that the criteria for selection of neighbors is the same for all nodes. In other words, each node s_k selects its neighbors as the nodes located inside a disk with an area of πr_c^2 centered at location $\mathbf{x}_k$ of node s_k, for $k = 1, 2, \cdots, n$. Moreover, if node s_k is a neighbor of node s_i, for $i \neq k$, then node s_i is also a neighbor of node s_k. Let s_j be the current processing node at step j. The node s_j selects the next node based on the objective function (13) from the set of candidate sensor nodes $\mathcal{C}_j^{s_j}$, which is the set of its neighbor nodes who have have not been participated in the estimation process previously. Note that each node $s_j \in \bar{\mathcal{S}}_n$ updates its set of candidate nodes $\mathcal{C}_j^{s_j}$ based on the information received from its neighbors. Thus in this scheme, each node has to communicate and keep track of the nodes participated in the estimation process only within its neighborhood. According to this approach,

the next node s_{j+1} at step $(j+1)$ is chosen as,

$$s_{j+1} = \underset{s_k \in \mathcal{C}_j^{s_j}}{\operatorname{argmax}} \; R(s_j, s_k), \tag{23}$$

where $R(s_j, s_k)$ is as given by (22). To find the next best node according to (23), the node s_j has to compute the reward function (22) only for candidate nodes in the set $\mathcal{C}_j^{s_j}$. Denote $m = \max_{k \in \{1,2,\cdots,n\}} \{|\mathcal{N}_k|\}$ to be the maximum size of the set of neighbors for any node s_k in the network where $m \leq n-1$. Each node s_j in the estimation process has to compute the reward function only for a maximum of m nodes. Since there is a maximum of n nodes, this leads to a worst case computational complexity of order $O(mn)$ for the whole network. Since, in this scheme a node has to keep track of only its neighbors, the communication complexity is reduced compared to the scheme presented in subsection IV-A whenever $m \ll n-1$. In distributed sensor networks, network architectures where nodes only communicate with their neighbors to make local estimators are desirable due to network resource constraints. For example, in [15], each mobile node communicates with its one-hop neighbors at a given time to make a local estimate of the target state where the one-hop neighborhood at each node is dynamically changing. The proposed distributed sequential estimation process based on local search is summarized in Algorithm 1 and described in detail in Fig. 4.

Algorithm 1 Sequential estimation process at step j at node s_j

while ($j \geq$)1 **do**

Compute estimate $\hat{\theta}_j$ from (6)

Compute MMSE M_j from (7)

if ($M_j <$ Desired performance or $\mathcal{C}_j^{s_j} = \emptyset$) **then**

1. Make the final estimator
2. Go to sleep mode

else

1. Select next node from $\mathcal{C}_j^{s_j}$ according to (23)
2. Send the current estimate to the node selected
3. Broadcast signal to nodes in $\mathcal{C}_j^{s_j}$ implying node s_j has been participated in estimation process
4. Go to sleep mode

end if

end while

Note that since node s_j selects the next node from the candidate set $\mathcal{C}_j^{s_j}$, node s_j only needs to perform $|\mathcal{C}_j^{s_j}|$ number of computations. Also, node s_j needs to keep track of the nodes which are not participated in the estimation process in its neighborhood only. Once the final estimator is made, a signal is broadcast implying the final estimator is made. Then all unprocessed nodes go to sleep mode, until the

next event occurs. Also it is to be noted that, when the effective communication range r_c is sufficiently large ($r_c \to \infty$), this scheme based on local approach converges to the scheme described in subsection IV-A. Thus the scheme described in subsection IV-A can be considered as a special case of the proposed scheme in this subsection when $r_c \to \infty$.

Updating candidate set at the k-th processing node: Denote $\mathcal{C}_j^{s_k}$ to be the candidate set of node s_k at the step j. Algorithm for updating the candidate set at node s_k is explained in Algorithm 2.

Algorithm 2 Updating candidate set at k-th node

NOTATION

s_k: k-th node, s_j: processing node at step j, $\mathcal{C}_j^{s_j}$: candidate set of the processing node s_j at step j, $\mathcal{C}_j^{s_k}$: candidate set of the node s_k at step j

INITIALIZATION

$\mathcal{C}_0^{s_k} = \mathcal{N}_k$

UPDATING

while ($j \geq 1$) **do**

  $\mathcal{C}_j^{s_j} = \mathcal{C}_{j-1}^{s_j}$

  if $s_k = s_j$ (i.e. node s_k becomes the current processing node at step j) **then**

    $\mathcal{C}_j^{s_k} = \mathcal{C}_j^{s_j}$

  else $\{s_k \in \mathcal{C}_j^{s_j}$ (i.e. node s_k belongs to the candidate set of the current processing node at step $j)\}$

    $\mathcal{C}_j^{s_k} = \mathcal{C}_{j-1}^{s_k} \setminus s_j$

  else

    $\mathcal{C}_j^{s_k} = \mathcal{C}_{j-1}^{s_k}$

  end if

end while

Note that, the node s_j is not a neighboring node for any node in the network except for those that are in $\mathcal{N}_j$ itself. Thus it is not necessary for nodes that are not in $\mathcal{N}_j$ to keep track of node s_j. However, this process can be terminated when the current node does not have any candidate neighboring nodes (i.e. $\mathcal{C}_j^{s_j} = \emptyset$), where $\emptyset$ is the null set, irrespective of whether the desired performance level is reached or not, eventhough there might be remaining nodes in other neighborhoods of the network. However, as observed from simulations, this does not seem to cause a significant performance loss. To further illustrate this scenario consider the sensor network as shown in Fig. 5. Assume that the process is started at the node s_1. The corresponding node s_j at step j and the perimeter of the disk centered at the location of s_j are shown by the same color. The neighborhood size of the node s_1, $|\mathcal{N}_{s_1}| = 7$. According to the algorithm based on local search, assume that the consecutive 7 nodes followed by node s_1 in the node sequence are happened to be the neighbors of the node s_1, as shown in the Fig. 5. If the node s_8 (the last node of the

8-length node sequence) and the node s_1 have the same neighborhood, it can be seen that the process is terminated at the node s_8, since $C_j^{s_j} = \emptyset$ for $j = 8$ although there are unprocessed nodes in the network. Denote $\bar{m} = \min_{k\in\{1,2,\cdots,n\}} \{|\mathcal{N}_k|\}$ to be the minimum size of the set of neighbors for any node s_k in the network where $\bar{m} > 0$. It can be seen that, this phenomenon occurs after, $\bar{m} + 1$ number of nodes are processed, at the worst case. Note that for the worst case scenario to be occurred, the following factors should be satisfied: (i). $|\mathcal{N}_{s_1}| = \bar{m}$ and (ii). $\mathcal{N}_{s_{\bar{m}+1}} \subseteq \mathcal{N}_{s_1}$. Moreover, with the node selection algorithm presented in this subsection, this worst case scenario will most likely happen to occur only if the starting node locates closer to the target position. Otherwise, according to (23) nodes located far from the node s_1 will be queried as the process continues allowing more nodes ($\gg \bar{m}$) are participated in the decision process. Thus, at the worst case, eventhough there is possibility for the process to be terminated after processing $\bar{m} + 1$ number of nodes, since those node are located closer to the target position, they will carry rich information on the PoI and will result a considerable performance (compared to processing all allowable nodes) with a relatively small number of nodes. Thus as also observed from simulations, this phenomenon does not seem to cause a significant performance loss even when varying β and r_c.

In both schemes discussed above in subsections IV-A and IV-B, a global minimum is not guaranteed in general since in both schemes current node selects the next best node from only all unvisited nodes in the network (in scheme discussed in IV-A), or in neighborhood (in scheme discussed in IV-B), sequentially. In subsection IV-C, a dynamic programming approach is presented to find the k-length optimal node sequence which yields the global minimum, with a higher computational complexity compared to that with the greedy type algorithms. We observe (see Section V) that when there is no channel noise, the global scheme discussed in subsection IV-A coincides with the optimal scheme that yields the global minimum (computed based on dynamic programming as in subsection IV-C) and the local scheme proposed in subsection IV-B performs close to the optimal scheme after processing relatively small number of nodes. Even when there is channel noise, we will see that both schemes perform fairly close to the optimal scheme after processing relatively a small number of nodes. We refer to the node selection scheme presented in subsection IV-A with global search as the scheme 1 and the proposed scheme with local search presented in subsection IV-B as the scheme 2, in the rest of the paper.

C. Optimal node sequence via dynamic programming

Denote s_k to be the k-th processing node, for $k = 1, 2, \cdots, n$ and s_1 to be the initial node in the sequential estimation process, as before. Also let $\bar{\mathcal{S}}_n \equiv \{s_1, s_2, \cdots, s_n\}$ be the set containing all nodes in the network. Let ϵ be the desired MMSE performance level and $\mathcal{S}_k = \{\hat{s}_1, \hat{s}_2, \cdots, \hat{s}_k\} \subseteq \mathcal{S}_n$ be a k-length sequence of distinct nodes with $\hat{s}_1 = s_1$. Then the optimal node ordering problem can be formulated as,

$$\begin{aligned} &\min\ |\mathcal{S}_k| \\ &\text{such that } M_k \leq \epsilon, \end{aligned} \tag{24}$$

where M_k is as defined in (7) and the minimization is over all possible distinct node sequences of length k (including ordering of the nodes), for $k = 1, \cdots, n$, starting at node s_1. Assume that the starting node s_1 is able to access the information regarding the observation noise variance and position information at each node without any additional cost. To obtain the optimal k-length distinct sequence which will yield the minimum MMSE at k-th node, one has to consider $\frac{(n-1)!}{((n-1)-(k-1))!} = (n-1)...(n-k)$ possible distinct sequences for each k and compute the minimum over all these sequences. At the worst case, if $k = n$, this results computing $\sum_{k=1}^{n}(n-1)...(n-k)$ distinct sequences which will cause a high computational burden. In the following we present an algorithm to compute the optimal node sequence which satisfies (24) based on forward dynamic programming with the best case complexity of $O(n^3)$.

The basic idea of the proposed algorithm is to compute the optimal k-length distinct node sequence denoted by $\hat{\mathcal{S}}_k^j$, which gives the minimum MMSE when the k-th node is s_j, for $j = 2, \cdots, n$. Then the best k-length node sequence which results the minimum MMSE (over all possible sequences) at the k-th node is given by, $\hat{\mathcal{S}}_k = \underset{\hat{\mathcal{S}}_k^j, j=2,\cdots,n}{\operatorname{argmin}} \{M_k(\hat{\mathcal{S}}_k^j)\}$, where $M_k(\hat{\mathcal{S}}_k^j)$ is the MMSE at the k-th node of the sequential estimation process when the node sequence is $\hat{\mathcal{S}}_k^j$. The process is continued over k, if the condition in (24) is not satisfied. To compute $\hat{\mathcal{S}}_k^j$ recursively for $k = 2, 3, \cdots$, $j = 2, \cdots, n$, we propose the following procedure based on the forward dynamic programming.

Let $\mathcal{D}_k$ be the state space for each $k = 2, \cdots, n$, where $\mathcal{D}_k \equiv \{s_2, \cdots, s_n\}$. We construct the trellis at each k for $k = 2, \cdots, n$, by concatenating of $n-1$ copies of the state space followed by a dummy terminating node s_t where $\mathcal{D}_{n+1} \equiv \{s_t\}$ as shown in Fig. 3. At each k, branch connections and the cost of transition from node $s_i \in \mathcal{D}_k, i = 2, \cdots, n$ to node $s_j \in \mathcal{D}_{k+1}, j = 2, \cdots, n$ are performed as k grows and explained in the following.

To explain the state transitions and the metric assignments of the trellis in Fig. 3, we introduce following notations. Denote $\mathcal{S}_k^j$ be a k-length sequence of distinct nodes in which the starting node is s_1 and the k-th node is s_j. Also denote $J_k(\mathcal{S}_k^j)$, to be the MMSE (or the cost in the context of dynamic programming) incurred at the k-th node when the k-length node sequence terminating at node s_j is $\mathcal{S}_k^j$ for $s_j \in \mathcal{D}_k$. When $k = 1$, since the process is started at node s_1, let $J_1(s_1)$ be MMSE at node s_1 as given by (5). For $k > 2$, depending on the sequence, the quantity $J_k(\mathcal{S}_k^j)$ can be computed from (7) after substituting parameters with appropriate notations.

Computing $J_k(\hat{\mathcal{S}}_k^j)$ via dynamic programming:

- Transition from $k = 1$ to $k = 2$

By $k = 2$, we need to find the optimal 2-length sequence starting from s_1 which will result the minimum MMSE. We assign the metric $f(J_1(s_1), \sigma_{s_j}^2, \sigma_{c(s_1,s_j)}^2) = \frac{\sigma_\theta^2}{\sigma_\theta^2 \tilde{d}_{s_1,s_j}^2 + 1}$ with $\tilde{d}_{s_1,s_j}^2 = \frac{1}{\sigma_{s_j}^2} + \frac{(\sigma_\theta^2 - J_1(s_1))^2}{\sigma_\theta^2 \left[J_1(s_1)(\sigma_\theta^2 - J_1(s_1)) + \sigma_\theta^2 \sigma_{c(s_1,s_j)}^2 \right]}$ for the arc connecting node s_1 to node $s_j \in \mathcal{D}_2$. Note that for clarity we use $\sigma_{s_j}^2$ and $\sigma_{c(s_i,s_j)}^2$ to denote observation noise variance at node s_j and the variance of the channel noise between nodes s_i and s_j for

$s_i \neq s_j$, respectively. Then the minimum MMSE at the 2-nd node when the 2-nd node of the 2-length distinct sequence is s_j, is,

$$J_2(\hat{\mathcal{S}}_2^j) = f(J_1(s_1), \sigma_{s_j}^2, \sigma_{c(s_1,s_j)}^2). \tag{25}$$

The optimal 2-length node sequence (over all 2-length node) is given by, $\hat{\mathcal{S}}_2 = \min_{s_j \in \mathcal{D}_2} \{J_2(\hat{\mathcal{S}}_2^j)\}$ and the corresponding minimum MMSE is $M_2(\hat{\mathcal{S}}_2)$. If $M_2(\hat{\mathcal{S}}_2)$ does not satisfy the condition in (24), the processes is continued over k.

- Transition from $k = 2$ to $k = 3$

For transition of stages at $k = 2$ to $k = 3$, node $s_i \in \mathcal{D}_2$ is connected to node $s_j \in \mathcal{D}_3$ for $s_i \neq s_j$ and for $i, j = 2, \cdots, n$ with the metric $f(J_2(\hat{\mathcal{S}}_2^j), \sigma_{s_j}^2, \sigma_{c(s_i,s_j)}^2) = \frac{\sigma_\theta^2}{\sigma_\theta^2 \tilde{d}_{s_i,s_j}^2 + 1}$ with $\tilde{d}_{s_i,s_j}^2 = \frac{1}{\sigma_{s_j}^2} + \frac{(\sigma_\theta^2 - J_2(\hat{\mathcal{S}}_2^i))^2}{\sigma_\theta^2 \left[J_2(\hat{\mathcal{S}}_2^i)(\sigma_\theta^2 - J_2(\hat{\mathcal{S}}_2^i)) + \sigma_\theta^2 \sigma_{c(s_i,s_j)}^2 \right]}$. Note that, the node $s_i \in \mathcal{D}_2$ is not connected to the node $s_j \in \mathcal{D}_3$ if $s_i = s_j$, since no node is visited more than once. Then the minimum MMSE at the 3-rd node when the 3-rd node of the 3-length distinct sequence is s_j, is,

$$J_3(\hat{\mathcal{S}}_3^j) = \min_{s_i \in \{\mathcal{D}_k \setminus s_j\}} f(J_2(\hat{\mathcal{S}}_2^i), \sigma_{s_j}^2, \sigma_{c(s_i,s_j)}^2), \tag{26}$$

where we use the notation $\mathcal{D}_k \setminus s_j$ to represent all the node elements of $\mathcal{D}_k$ except node s_j. The optimal 3-length node sequence (over all 3-length node) is given by, $\hat{\mathcal{S}}_3 = \min_{s_j \in \mathcal{D}_3} \{J_3(\hat{\mathcal{S}}_3^j)\}$.

- Transition from k to $k + 1$ for $k \geq 3$

When connecting arcs and assigning costs for transitions from k to $k+1$ for $k \geq 3$, additional constraints are to be considered. If $J_k(\hat{\mathcal{S}}_k^j)$ is computed at $k \geq 3$ and the condition (24) is not satisfied, the process should be continued to $k + 1$. When connecting branches and assigning costs from node $s_i \in \mathcal{D}_k$ to $s_j \in \mathcal{D}_{k+1}$, it should be noted that, any node is processed only once in the sequential estimation process. For $k \geq 3$, the optimal sequences $\hat{\mathcal{S}}_k^j$ computed for each $s_j \in \mathcal{D}_k$, contain more than 2 distinct nodes except s_1. Thus when $s_i \in \mathcal{D}_k$ is connected to $s_j \in \mathcal{D}_{k+1}$ for $s_i \neq s_j$ and $k \geq 3$, the associated cost can not be computed based on $J_k(\hat{\mathcal{S}}_k^i)$ if $s_j \in \hat{\mathcal{S}}_k^i$. When connecting node $s_i \in \mathcal{D}_k$ to node $s_j \in \mathcal{D}_{k+1}$ for $k \geq 3$, we follow steps given below:

- if $s_i = s_j$ for $s_i \in \mathcal{D}_k$, $s_j \in \mathcal{D}_{k+1}$, no arc is connected between s_i and s_j.
- if $s_i \neq s_j$, for $s_i \in \mathcal{D}_k$, $s_j \in \mathcal{D}_{k+1}$, arcs are connected from $s_i \in \mathcal{D}_k$ $s_j \in \mathcal{D}_{k+1}$. Computing of the corresponding metrics associated with the transitions are done as follows. Since no node is visited more than once, it requires to check whether $s_j \in \hat{\mathcal{S}}_k^i$: that is check whether the node s_j has already been participated in the optimal sequence $\hat{\mathcal{S}}_k^i$. If so, find the k-length optimal sequence terminating at node $s_i \in \mathcal{D}_k$ in which the node s_j has not been participated. Denote $\tilde{J}_k(\hat{\mathcal{S}}_k^i \setminus s_j)$ to be the minimum MMSE resulted at node $s_i \in \mathcal{D}_k$ without visiting node s_j at any stage $l < k$. If $s_j \in \hat{\mathcal{S}}_k^i$, $\tilde{J}_k(\hat{\mathcal{S}}_k^i \setminus s_j)$ should be computed going backward in the trellis from the current stage k, which requires some

additional computation. If $s_j \notin \hat{\mathcal{S}}_k^i$, then $\tilde{J}_k(\hat{\mathcal{S}}_k^i \setminus s_j) = J_k(\hat{\mathcal{S}}_k^i)$ and does not require any additional computation. The metric corresponding to the transition from node $s_i \in \mathcal{D}_k$ to $s_j \in \mathcal{D}_{k+1}$ is given by $f(\tilde{J}_k(\hat{\mathcal{S}}_k^i \setminus s_j), \sigma_{s_j}^2, \sigma_{c(s_i,s_j)}^2) = \frac{\sigma_\theta^2}{\sigma_\theta^2 \tilde{d}_{s_i,s_j}^2 + 1}$ with $\tilde{d}_{s_i,s_j}^2 = \frac{1}{\sigma_{s_j}^2} + \frac{(\sigma_\theta^2 - \tilde{J}_k(\hat{\mathcal{S}}_k^i \setminus s_j))^2}{\sigma_\theta^2 \left[\tilde{J}_k(\hat{\mathcal{S}}_k^i \setminus s_j)(\sigma_\theta^2 - \tilde{J}_k(\hat{\mathcal{S}}_k^i \setminus s_j)) + \sigma_\theta^2 \sigma_{c(s_i,s_j)}^2\right]}$. Then the forward DP recursion is given by,

$$J_{k+1}(\hat{\mathcal{S}}_{k+1}^j) \;=\; \min_{s_i \in \mathcal{D}_k \setminus s_j} \{f(\tilde{J}_k(\hat{\mathcal{S}}_k^i \setminus s_j), \sigma_{s_j}^2, \sigma_{c(s_i,s_j)}^2)\}, s_j \in \mathcal{D}_k, k = 3, \cdots, n+1, \tag{27}$$

and the optimal $(k+1)$-length distinct node sequence which results the global MMSE at the $(k+1)$-th node is given by, $\hat{\mathcal{S}}_{k+1} = \min_{s_j \in \mathcal{D}_{k+1}} \{J_{k+1}(\hat{\mathcal{S}}_{k+1}^j)\}$.

From (25), (26) and (27), we observe that the objective function of the dynamic program is not additive but is a known function which can be computed recursively as the process continues. It is also worth mentioning that, under this formulation, branch connections from stage k to $k+1$ are performed by taking the optimal node sequences up to stage k corresponding to $J_k(\mathcal{S}_k^j), s_j \in \mathcal{D}_k$ into account. Thus, the branch connection and finding the optimal node sequence should be performed simultaneously as the process continues (k increases).

However, with a large network size n, the above dynamic programming approach would be computationally complex. If the optimization were to be performed at the initial node, it also requires additional costs (corresponding to latency and bandwidth) for communication. It is noted that both distributed schemes proposed in subsections IV-A and IV-B can be implemented with reduced computational and communication complexities compared to that with the optimal scheme. We also show that the algorithms discussed in subsections IV-A and IV-B can lead to the exact or close to exact results to the optimal scheme (computed based on dynamic programming) under certain conditions. Details are given in Section V.

V. Performance Analysis

We consider a 2D square sensor network of area A on $X \times Y$ plane. The locations of the k-th node and the target are denoted by $\mathbf{x}_k = (x_k, y_k)$, for $k = 1, \cdots, n$, and $\mathbf{x}_t = (x_t, y_t)$, respectively as before. In the following we analyze the performance of a fixed 2D network when the target location is known exactly.

We assume that the node s_j has knowledge of its own position, target location and the positions of its neighbors $\mathcal{N}_j$. Then the observation noise variance at the k-th node according to the model (2) can be expressed as

$$\sigma_k^2 = \left(\frac{r_{kt}}{r_0}\right)^\alpha \sigma_0^2, \tag{28}$$

where $r_{kt} = \sqrt{(x_k - x_t)^2 + (y_k - y_t)^2}$ is the distance between the k-th node and the target, α is the path loss index and r_0 and σ_0^2 are constants. The channel noise variance between $(k-1)$-th node and the k-th

node is given by

$$\sigma^2_{c(k-1,k)} = \left(\frac{r_{k-1,k}}{r'_0}\right)^{\alpha'} \sigma^2_c, \tag{29}$$

where $r_{k-1,k} = \sqrt{(x_k - x_{k-1})^2 + (y_k - y_{k-1})^2}$ is the distance between the k-th node and the $(k-1)$-th node, α' is the path loss index and r'_0 and σ^2_c are constants.

In the proposed schemes, node s_j computes the estimator and the MMSE according to (6) and (7). If the desired MMSE threshold is not met, node s_j sends its information to the node s_{j+1}, where the node s_{j+1} is selected from the candidate set $\mathcal{V}^c_j$ according to (22) in the scheme 1 and from $\mathcal{C}^{s_j}_j$ according to (13) in the scheme 2. We assume that there are 40 sensor nodes deployed in a square region of 10×10 square units. The target is assumed to be at the origin and the initial node is selected randomly and assumed same for all plots. Neighbors at each node are selected as the set of nodes located within a disk of radius $r_c = 3$ units.

In the first experiment, we investigate performance of the greedy based sequential estimation processes when both mutual information and the MMSE are considered as information utility measures. Figure 6 shows the required number of nodes to be processed to achieve a desired MMSE performance for different values of β with both information utility measures. The top plot of Fig. 6 corresponds to the global scheme (scheme 1) while the bottom plot corresponds to the local approach (scheme 2) with $r_c = 3$ units. In Fig. 6, $\sigma^2_c = 0$. When $\beta = 1$ and $\beta = 0$ (nearest node selection method), it can be seen from Fig. 6 that the node selection using both utility measures depict similar performance in the sequential estimation process. For intermediate value of $\beta = 0.5$, it can be seen from Fig. 6 that the performance of the sequential estimation process with node selection based on mutual information gives relatively better performance compared to that with MMSE being the information utility measure for the smaller region of ϵ. However, it can be seen that when ϵ is moderate, the performance of both node selection scheme coincides with one another. Thus it can be seen that for both global and local approaches, choosing MMSE as an information utility measure would be a good alternative to mutual information especially when the desired MMSE takes moderate values.

Figure 7 shows the MMSE performance at the k-th node with no channel noise such that $\sigma^2_{c(k-1,k)} = \sigma^2_c = 0$ (in Fig. 7(a)) and with channel noise with $\sigma^2_{c(k-1,k)}$ as given in (29) (in Fig. 7(b)), with the MMSE as the information utility measure.

With no channel noise, it can be seen that when $\beta = 1$, the performance with node ordering based on proposed scheme 1 coincides with that of the optimal scheme which results in the global minimum. In that case, from Fig. 7(a) it can be seen that the performance of the proposed scheme 2 converges to that of scheme 1 (as well as to that with optimal scheme) after a relatively small number of processing nodes. For $\beta = 0.5$ and $\beta = 0$, it is seen that proposed scheme 1 and scheme 2 give similar performance. For example, with $\beta = 1$, to achieve a required performance level of an MMSE of 0.05, scheme 1 requires 2

nodes, while scheme 2 requires 4 nodes. On the other hand, to achieve the same performance level, both scheme 1 and the scheme 2 require 8 and 12 nodes with $\beta = 0.5$ and $\beta = 0$, respectively. It is noted that the proposed scheme 2 is terminated at node 22, 32 and 25 with $\beta = 1$, $\beta = 0.5$ and $\beta = 0$, respectively, due to the reason discussed in subsection IV-B. However, it is seen that once such a number of nodes are processed node ordering does not affect the overall performance level. This implies that when the sequential estimation process is continued among a large number of sensors, the performance converges to the same value irrespective of how the nodes are selected, which of course is not desirable in many resource constrained sensor networks.

On the other hand, when there is channel noise, it is seen that continuing the sequential processing after some point does not yield improved performance as can be seen from Fig. 7(b). This essentially is due to the fact observed in (12). However, in this case, from Fig. 7(b) it can be seen that the proposed scheme 2 (local approach) with $\beta = 1$ gives closer performance to that of with the optimal scheme. Also we can see that the performance of proposed scheme 1 (global approach) and that of the scheme 2 is almost the same for $\beta = 0.5$ and $\beta = 0$. When $\beta = 1$, from Fig. 7(b) it can be seen that the proposed scheme 2 yields a lower MMSE compared to the scheme 1 after processing a certain number of nodes. This can be explained by noting the fact that both proposed schemes 1 and 2 are greedy-type algorithms. Thus they would not necessarily result in the same global minimum after completing the same number of processing stages.

From these performance results, we can see that in proposed sequential estimation processes, the proposed greedy-type algorithms essentially result in a near-optimal solution in finding the best ordering of nodes compared to that with the optimal scheme which yields the global minimum at a high computational and communication cost.

Figure 8 shows the performance of the scheme 2 (local search with $r_c = 3$ units) with the trade-off parameter β in the reward function. Figs. 8(a) and 8(b) correspond to the performance with mutual information and MMSE as information utility measures, respectively. From Fig. 8, it is observed that the effect of the values of β is more significant to achievable MMSE after processing a small number of nodes (compared to that with a large number of processing nodes) for both information utility measures and all considered values of channel noise qualities. However, after processing a relatively a large number of nodes, the effect of selecting β does not seem to cause a large difference in the MMSE performance. Since the most desirable scenario in a sensor network is to make the final estimator with a small number of nodes and with a high accuracy, the trade-off parameter value β should be selected carefully to achieve the best possible performance considering both the number of nodes and the communication costs.

In Fig. 9, we investigate the minimum achievable MMSE with varying neighborhood size, r_c, in the scheme 2 for $\sigma_c^2 = 0.001$. Note that as pointed out in subsection IV-B, the proposed scheme 2 terminates at a certain point if all candidate nodes in the current node have already been participated in the estimation

process although there might be unvisited nodes in the network. Fig. 9 (a) shows the minimum achievable MMSE when r_c is varying for different values of β while Fig. 9 (b) shows the number of nodes in the estimation process before the process terminates in scheme 2. Compared to the achievable minimum MMSE with the scheme 1 (global approach), it can be seen from Fig. 9 (a) that although the process terminates before all nodes are processed, the scheme 2 (with moderate values of r_c) does not seem to result a severe performance loss compared to that with scheme 1. From Fig. 9 (b), it can be seen that before terminating, the scheme 2 processes a considerable number of nodes (compared to total nodes) for all considered values of r_c.

VI. Conclusion

This paper presents a sequential estimation methodology for parameter estimation in an autonomous distributed sensor network. The proposed work considers the inter-node communication noise and the sensor node selection jointly for the sequential estimation contrast to most existing sequential schemes for parameter estimation. In the proposed sequential estimation framework, each node makes a local estimate by combining its own observation and the estimator from the previous node. To update the estimator at the next node, the current node's estimator is sent to the next node via a noisy communication channel. To find the optimal number of nodes with the best ordering that should be participated in the estimation process, we develop an algorithm based on forward dynamic programming which might be computationally complex when the network size is large. We proposed two distributed greedy type node selection schemes to select the best node ordering in the sequential estimation process based on a certain reward function. To select the next best node, in the scheme 1 the current node maximizes the reward function over all unvisited nodes in the network, while in the scheme 2 the current node maximizes the reward function over nodes in only its neighborhood. To implement the scheme 1, each node has to keep track of all nodes in the network which are already been participated in the estimation process. In the scheme 2, to perform the distributed sequential estimation process, each node has to only keep track of its neighboring nodes. We show that the performance with the scheme based on the local search gets closer to that with the scheme with global search with a relatively small number of nodes. Thus the proposed scheme 2 can be considered as a better trade-off between the performance gain and the communication and other costs incurred in message transmission in the sequential estimation process. Since both proposed distributed node selection schemes are greedy algorithms, they do not guarantee a global minimum MMSE. We compare the performance of two greedy type schemes with the optimal scheme computed based on forward dynamic programming, and show that the performance of two greedy type schemes gets very closer to the optimal solution after processing relatively a small number of nodes.

REFERENCES

[1] Z.-Q. Luo and J.-J. Xiao, "Universal decentralized estimation in a bandwidth constrained sensor network," *IEEE Trans. Inf. Theory*, vol. 51, no. 6, pp. 2210–2219, June 2005.

[2] G. Mergen and L. Tong, "Type based estimation over multiaccess channels," *IEEE Trans. Signal Processing.*, vol. 54, no. 2, pp. 613–626, Feb. 2006.

[3] V. Aravinthan, S. K. Jayaweera, and K. Al Tarazi, "Distributed estimation in a power constrained sensor network," in *IEEE 63rd Vehicular Technology Conference (VTC) 2006 Spring*, Melbourne, Australia, May 2006.

[4] T. Wimalajeewa and S. K. Jayaweera, "Power efficient distributed estimation in a bandlimited wireless sensor network," in *41st Asilomar Conference on Signals, Systems and Computers (ACSSC)*, Pacific Grove, CA, Nov. 2007, pp. 2156–2160.

[5] T. Zhao and A. Nehorai, "Distributed sequential bayesian estimation of a diffusive source in wireless sensor networks," *IEEE Trans. Signal Processing.*, vol. 55, no. 4, pp. 1511–1524, Apr. 2007.

[6] F. Zhao, J. Shin, and J. Reich, "Information-driven dynamic sensor collaboration," *IEEE Signal Processing Magazine*, vol. 19, no. 2, pp. 61–72, Mar. 2002.

[7] M. Chu, H. Haussecker, and F. Zhao, "Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks," *Int. J. High-Perform. Comput. Applicat.*, vol. 16, no. 3, pp. 90–110, 2002.

[8] L. Zuo, R. Niu, and P. K. Varshney, "Posterior CRLB based sensor selection for target tracking in sensor networks," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Honolulu, Hawaii, USA, April 2007, vol. 2, pp. II–1041–II–1044.

[9] A. Vosoughi and R. Gang, "Scalar quantizers for decentralized estimation of multiple random sources," in *Proc. Military Communications Conference (MILCOM),*, Nov. 2008.

[10] S. K. Jayaweera and C. Mosquera, "Distributed sequential estimation with noisy, correlated observations," *IEEE Signal Processing Letters*, vol. 15, pp. 741–744, 2008.

[11] R. Niu and P. K. Varshney, "Performance analysis of distributed detection in a random sensor field," *IEEE Trans. on Signal Processing*, vol. 56, no. 1, pp. 339 – 349, Jan. 2008.

[12] H. V. Poor, *An Introduction to Signal Detection and Estimation*, Springer-Verlag, New York, 1994.

[13] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, John Willey and Sons, Inc., NY, 2006.

[14] D. Guo, S. Shamai, and S. Verd, "Mutual information and minimum mean-square error in gaussian channels," *IEEE Trans. Info. Theory*, vol. 51, no. 4, pp. 1261–1282, Apr. 2005.

[15] Y. Zou and K. Chakrabarty, "Distributed mobility management for target tracking in mobile sensor networks," *IEEE. Trans. on Mobile Computing*, vol. 6, no. 8, pp. 872–887, Aug. 2007.

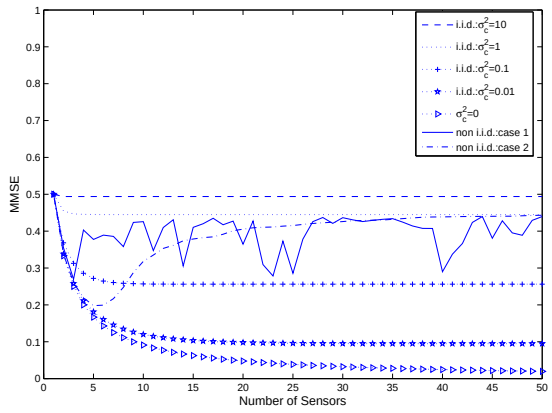


Fig. 1. MMSE vs. number of sensor when observation noise is i.i.d.

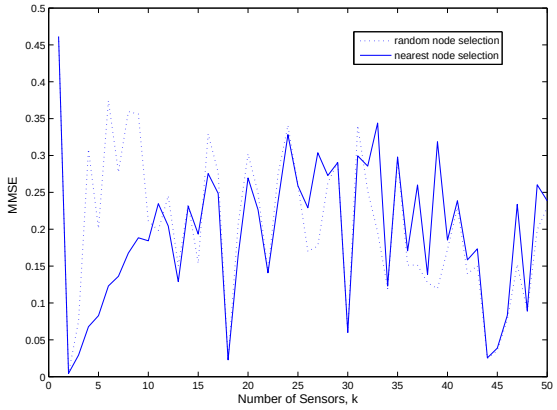


Fig. 2. MMSE vs number of sensors when observation noise is non-i.i.d.

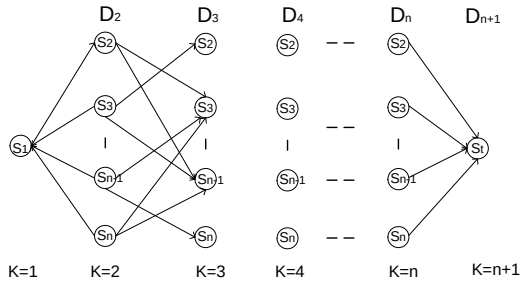


Fig. 3. Illustration of the shortest path formulation of the optimization problem (24)

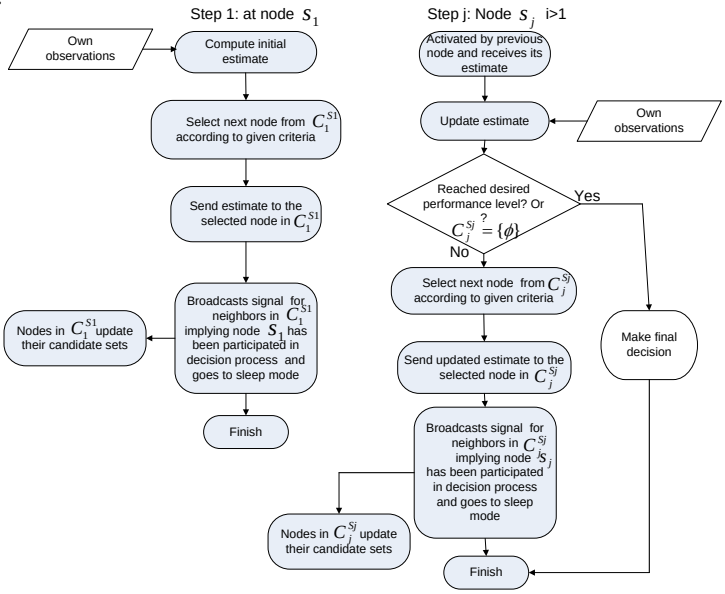


Fig. 4. Distributed Sequential Estimation Process at Node s_j for $j = 1, 2, \cdots$

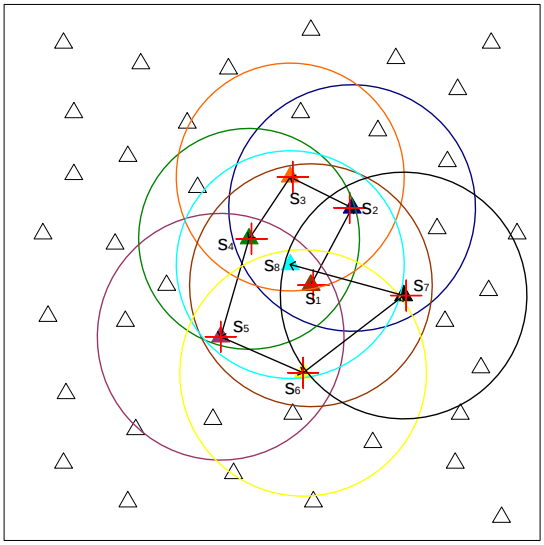


Fig. 5. Illustration of next node selection: $s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8$ is the sequence of nodes (with starting node s_1) in the estimation process before the process terminates at node s_8

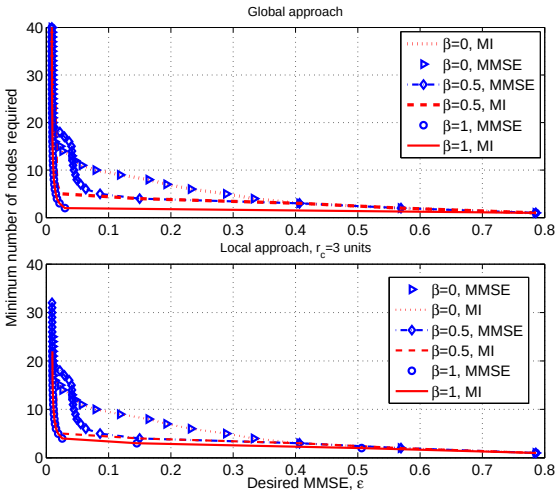
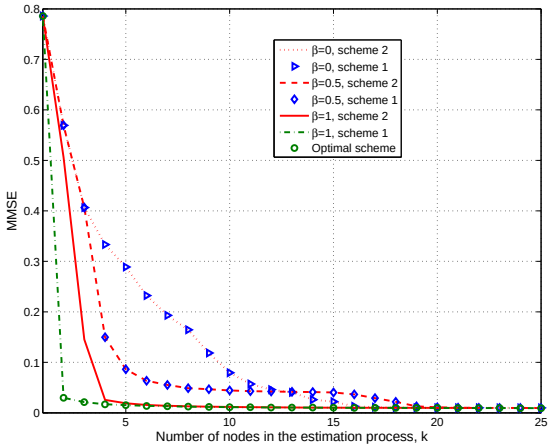
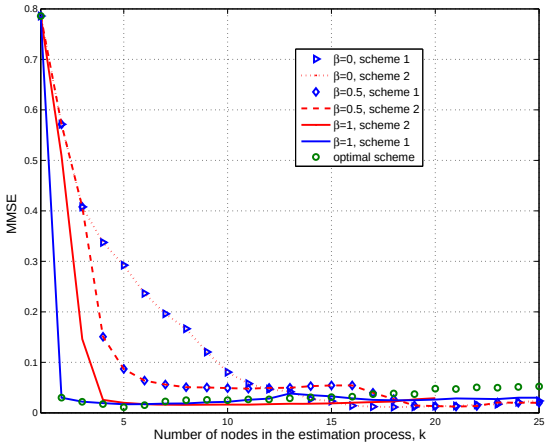


Fig. 6. Required minimum number of nodes to achieve a desired MMSE ϵ with both mutual information and MMSE as information utility measures: top plot- greedy based global approach, bottom plot- greedy based local approach with $r_c = 3$ units ($\sigma_c^2 = 0$, $\alpha' = 2$, $r_0 = 1$, $\sigma_0^2 = 0.1$).

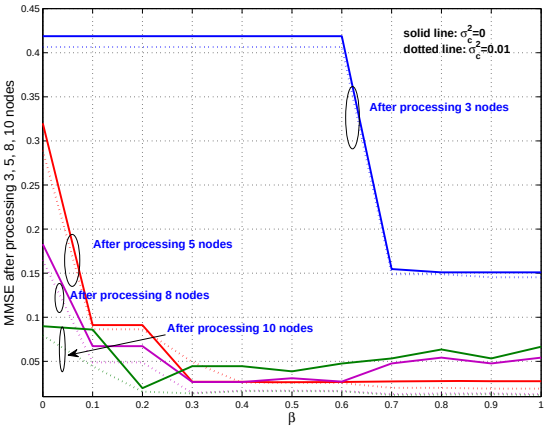


(a)

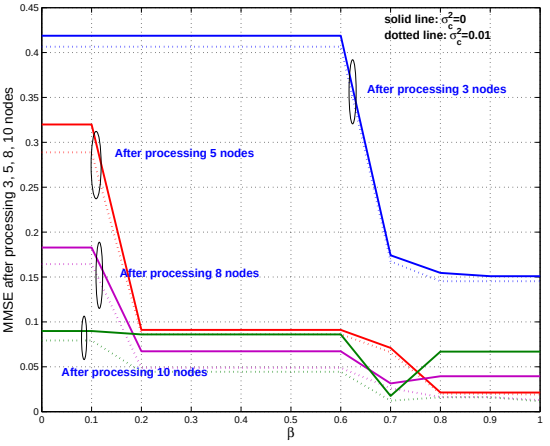


(b)

Fig. 7. MMSE at the k-th node with exact target location with MMSE as the information utility measure: (a). No channel noise (b). With channel noise: $\sigma^2_{c(k-1,k)}$'s are as given in (29),$n = 40$, $\sigma_c^2 = 0.001$, $\alpha' = 2$, $r_0 = 1$

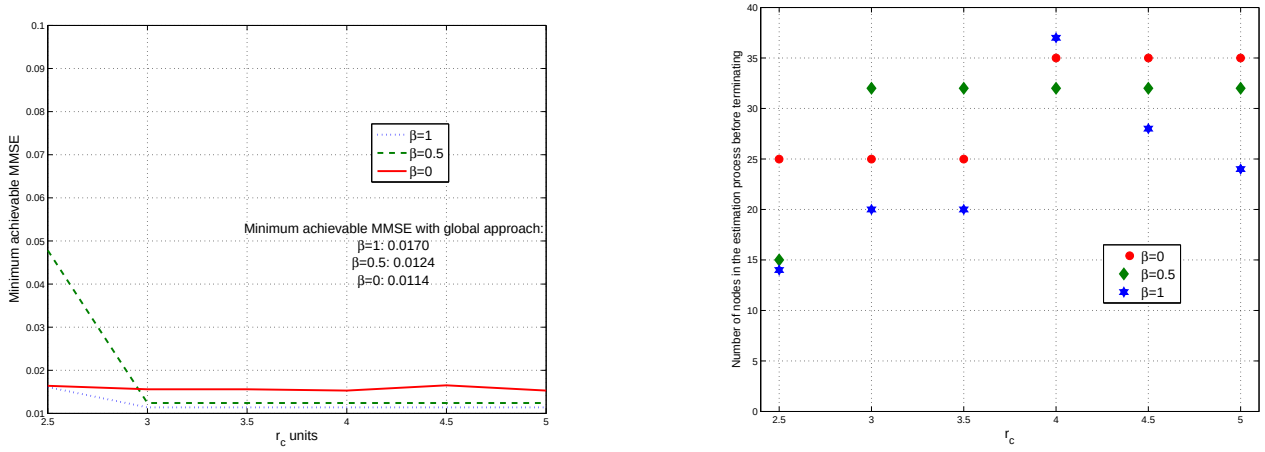


(a)



(b)

Fig. 8. MMSE Vs β for scheme 2 with $r_c = 3$ units with varying channel noise qualities: (a). with MI as information utility measure (b). with MMSE as information utility measure



(a) (b)

Fig. 9. Effect of the sequential estimation process terminating before processing all nodes in the network in scheme 2: $\sigma_c^2 = 0.001$, $n = 40$ (a). Minimum achievable MMSE for different values of β for varying r_c (b). Number of nodes in the sequential estimation process before the process terminates for varying r_c