A DISTRIBUTED CONSENSUS PLUS INNOVATION PARTICLE FILTER FOR NETWORKS WITH COMMUNICATION CONSTRAINTS

Arash Mohammadi and Amir Asif

Department of Electrical Engineering and Computer Science York University, Toronto, ON, Canada M3J 1P3. Email:{marash, asif}@cse.yorku.ca

ABSTRACT

Motivated by the problem of distributed signal processing in sensor networks, the paper considers the general problem of state estimation in geographically dispersed systems with nonlinear dynamics operating in an uncertain environment with communication constraints. Distributed particle filter implementations used as nonlinear state estimators introduce an additional consensus step, which must converge to achieve consistent values for local estimators' statistics in between two consecutive filter iterations. The number of consensus iterations per consensus run is high such that the consensus step may not converge in between two filter iterations especially in networks with intermittent connectivity. To reduce the consensus liability, we propose a consensus plus innovation based distributed implementation of the unscented particle filter (CI/DUPF), which extends the linear consensus and innovation framework to nonlinear distributed estimation. The CI/DUPF does not require the consensus step to converge and is suited for environments with intermittent connectivity. In our Monte Carlo simulations, the performance of the CI/DUPF follows that of its centralized counterpart even with a limited number of consensus iterations per consensus run.

Index Terms— Consensus protocols, Wireless sensor networks, Distributed estimation, Intermittent connectivity, and Particle filters.

1. INTRODUCTION

Widespread deployment of sensor agents has revolutionized our ability to monitor physical environments. Not only do these sensors observe an environment independently, they process measured data locally, and collaborate, thereby, providing a suitable framework for distributed estimation in nonlinear applications as diverse as source localization in robotics [1, 2], submarine tracking in sonars [3], and surveillance in radars [4]. For the sensor networks to reach their full potential, they must however be capable of operating in hostile conditions. Our goal in this paper is to propose a distributed particle filter implementation [5]-[23] for such unstructured environments with communication constraints and connectivity issues.

In recent years, there has been a surge of interest in consensusbased distributed particle filter implementations [5]-[20] for state estimation and tracking problems in systems with nonlinear dynamics. Based on the type of information exchanged between the sensor nodes, these implementations can broadly be classified into two categories. Category 1 [7]-[12] communicates predefined statistics of the local posteriors (e.g., the state estimates and/or their corresponding error covariances) between the neighboring nodes. Sharing such local posterior's statistics is resilient to packet losses especially in error prone or congested networks since any lost information in principle should be contained in the following posteriors and, therefore, can be recovered. Category 2 [13]-[20] instead communicates a predefined function of the local observations to reach a consensus on the global likelihood. Irrespective of which category an implementation belongs to, current state-of-the-art distributed implementations of the particle filter [7]-[20] require several consensus runs, each requiring a large number of iterations to achieve consistent values for local estimators' statistics through information exchanges between neighboring nodes. The performance of these distributed implementations degrades severely in environments with communication constraints, i.e., when only a limited number of consensus iterations is possible due to intermittent connectivity or other communication constraints. Consequently, the consensus step does not converge between two consecutive filter iterations. To date, distributed estimation with communication constraints [24]-[26] is limited to linear systems based on the Kalman filter and has not yet been extended to nonlinear systems. The paper addresses this gap and proposes the consensus plus innovation based distributed implementation of the unscented particle filter (CI/DUPF) to deal with such connectivity issues. Being a marriage between Categories 1 and 2, the CI/DUPF benefits from advantages of both categories and keeps the estimation error bounded even without the convergence of the consensus step.

Designed as a consensus plus innovation distributed nonlinear estimator, the CI/DUPF is the nonlinear (particle filter based) counterpart of the linear consensus and innovation filter [25]. The CI/DUPF is ideally suited for applications where the global likelihood function satisfies the sufficient statistics based factorization as is the case for bearing-only [15, 27], joint bearing/range [28] tracking problems and in scenarios where the local likelihood functions belong to the exponential family [14]. In our Monte Carlo simulations, the performance of the CI/DUPF follows that of the centralized filter even with a limited number of consensus iterations.

2. PROBLEM FORMULATION AND PARTICLE FILTER

The overall nonlinear state-space model is given by

State Model:

$$\mathbf{x}(k) = \mathbf{f}(\mathbf{x}(k-1)) + \mathbf{\xi}(k), \quad (1)$$
Observation Model:

$$\begin{bmatrix} \mathbf{z}^{(1)}(k) \\ \vdots \\ \mathbf{z}^{(N)}(k) \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{g}^{(1)}(\mathbf{x}(k)) \\ \vdots \\ \mathbf{g}^{(N)}(\mathbf{x}(k)) \end{bmatrix}}_{g(\mathbf{x}(k))} + \underbrace{\begin{bmatrix} \boldsymbol{\zeta}^{(1)}(k) \\ \vdots \\ \boldsymbol{\zeta}^{(N)}(k) \end{bmatrix}}_{\boldsymbol{\zeta}(k)}, \quad (2)$$

for a sensor network comprising N nodes and observing a set of n_x states $\boldsymbol{x} = [X_1, X_2, \ldots, X_{n_x}]^T$. T denotes transposition. The observation vector is $\boldsymbol{z} = [\boldsymbol{z}^{(1)T}, \ldots, \boldsymbol{z}^{(N)T}]^T$ with $\boldsymbol{z}^{(l)}(k)$ denoting the observation at node $l, (1 \leq l \leq N)$, at time instant k. Functions $\{\boldsymbol{f}(\cdot), \boldsymbol{g}(\cdot)\}$ are nonlinear. $\{\boldsymbol{\xi}(\cdot), \boldsymbol{\zeta}(\cdot)\}$ are the global non-Gaussian uncertainties in the process and observation models respectively. The optimal Bayesian filtering recursion for iteration k is given by

$$P(\boldsymbol{x}(k)|\boldsymbol{z}(1:k-1)) = (3)$$

$$\int P(\boldsymbol{x}(k-1)|\boldsymbol{z}(1:k-1))f(\boldsymbol{x}(k)|\boldsymbol{x}(k-1))d\boldsymbol{x}(k-1)$$

and
$$P(\boldsymbol{x}(k)|\boldsymbol{z}(1:k)) = \frac{P(\boldsymbol{z}(k)|\boldsymbol{x}(k))P(\boldsymbol{x}(k)|\boldsymbol{z}(1:k-1))}{P(\boldsymbol{z}(k)|\boldsymbol{z}(1:k-1))}.$$
 (4)

The particle filter is based on the principle of sequential importance sampling [29]-[31], where the filtering distribution $P(\boldsymbol{x}(k)|\boldsymbol{z}(1:k))$ is represented by its samples (particles) $X_i(k), (1 \le i \le N_s)$, derived from a proposal distribution $q(\boldsymbol{x}(0:k)|\boldsymbol{z}(1:k))$ with weights

$$W_{i}(k) = \frac{P(\mathbb{X}_{i}(k)|\boldsymbol{z}(1:k))}{q(\mathbb{X}_{i}(0:k)|\boldsymbol{z}(1:k))}$$
(5)

associated with the vector particles. The particle filter implements the filter recursions by propagating the particle set $\{X_i(k), W_i(k)\}$ as

$$\mathbb{X}_{i}(k) \sim q(\mathbb{X}_{i}(k)|\mathbb{X}_{i}(0:k-1), \boldsymbol{z}(1:k))$$
(6)

$$W_{i}(k) \propto W_{i}(k-1) \frac{P(\boldsymbol{z}(k)|\mathbb{X}_{i}(k))P(\mathbb{X}_{i}(k)|\mathbb{X}_{i}(k-1))}{q(\mathbb{X}_{i}(k)|\mathbb{X}_{i}(0:k-1), \boldsymbol{z}(1:k))}.$$
(7)

3. CONSENSUS PLUS INNOVATION BASED DUPF

Designed as a consensus plus innovation distributed nonlinear estimator, the CI/DUPF is the particle filter counterpart of the linear consensus plus innovation filter [25]. The estimation error associated with the CI/DUPF remains bounded even when the number N_c of consensus iterations between two consecutive observations is less than the number $N_c(U)$ of iterations required for the convergence of the consensus step, i.e., $N_c < N_c(U)$.

3.1. Linear Consensus and Innovation Estimation

To provide insight into its extension to nonlinear estimation, we review the linear consensus plus innovation distributed filter [26] for a linear system with state model $\boldsymbol{x}(k) = \boldsymbol{F}\boldsymbol{x}(k-1) + \boldsymbol{\xi}(k)$ and observation model $\boldsymbol{z}^{(l)}(k) = \boldsymbol{G}^{(l)}\boldsymbol{x}(k) + \boldsymbol{\zeta}^{(l)}(k)$ at node *l*. The state estimate in the linear consensus plus innovation filter is updated as

$$\hat{\boldsymbol{x}}^{(l)}(k+1) = \boldsymbol{F} \underbrace{\sum_{i \in \mathbb{N}^{(l)}} U_{li} \hat{\boldsymbol{x}}^{(i)}(k)}_{\text{Consensus update}} + \boldsymbol{B}^{(l)} \underbrace{\sum_{i \in \mathbb{N}^{(l)}} [\boldsymbol{G}^{(i)}]^T \left(\boldsymbol{z}^{(i)}(k+1) - \boldsymbol{G}^{(i)} \hat{\boldsymbol{x}}^{(l)}(k) \right)}_{\text{Innovation correction}},$$
(8)

where $\aleph^{(l)}$ is the set of neighboring nodes for node l, $B^{(l)}$ the local innovation gain, and $U = \{U_{li}\}$ the consensus matrix. The consensus update step in (8) is equivalent to the local prediction step, where each node fuses its local predicted state estimate $F\hat{x}^{(l)}(k)$ with those of its neighbors ($F\hat{x}^{(i)}(k), i \in \aleph^{(l)}$) using multiple consensus runs. The innovation correction step in (8) is based on observations made at the nodes connected to node l. In (8), node lcollects the observations of its neighbors and computes their innovation based on its local state estimate. Please refer to [26] for details.

3.2. The CI/DUPF Implementation

The CI/DUPF extends (8) to nonlinear systems and derives a consensus plus innovation based distributed particle filter capable of handling communication constraints using the following two steps.

Consensus update: As stated previously, the consensus update step in Eq. (8) is actually the local prediction step where each node fuses its local predicted state estimate with those of its neighboring nodes. Recall that the prediction step in the particle filter generates new particles from the proposal distribution. In order to run the consensus update in the CI/DUPF, node l forms the proposal distribution based on its local state estimate and those of its neighboring nodes. There is a complication in deriving the proposal distribution for the consensus plus innovation based CI/DUPF implementation, i.e., the proposal distribution needs to be as close as possible to the posterior distribution. In the CI/DUPF, after completion of iteration k, node l runs a local consensus iteration on the local state estimates in its neighborhood and computes $\sum_{i \in \aleph^{(l)}} \hat{x}^{(i)}(k)$. Then, node l computes a local statistic of the posterior at iteration k+1 based on $\sum_{i \in \mathbb{N}^{(l)}} \hat{x}^{(i)}(k)$ and its local observation $z^{(l)}(k+1)$. Another local consensus run fuses these local statistics which are then used as the statistics of the local proposal distribution.

Observation update: In the context of the CI/DUPF, the analogy of the innovation update is the weight update step. In order to run the innovation update, each node needs to form a weight update equation based on the local observations available in its neighborhood. In the CI/DUPF, this step is accomplished by first computing the local sufficient statistics (LSS) and then forming the global sufficient statistics (GSS) within the local neighborhoods. In other words, node *l* forms a local likelihood function based on the GSSs computed only within its immediate neighborhood.

Having described the principle, we present the CI/DUPF in more details below. The CI/DUPF is assumed to be in steady state at iteration k. The state estimates and the weights are computed cooperatively. Conditioned on the state variables, the observations made at different nodes are assumed to be independent (a common assumption in distributed particle filter implementations [6]). All nodes have updated the state estimates $(\hat{x}^{(l)}(k) \text{ and } \hat{P}^{(l)}(k))$ at time instant k. A new measurement $z^{(l)}(k+1)$ is available, which is used to perform the following consensus plus innovation steps at node l.

3.2.1. Consensus Update

The consensus update consists of the following three substeps.

Step 1. Local UPF: Node *l* generates a set of $(2n_x+1)$ deterministic samples $S = \{W_i^{(l)}, \chi_i^{(l)}(k)\}_{i=0}^{2n_x}$ (referred to as the sigma points) based on the following selection procedure

$$\boldsymbol{\chi}_{i}^{(l)}(k) = \hat{\boldsymbol{x}}^{(l)}(k) \pm \left\{ \sqrt{(n_{x} + \kappa) \hat{\boldsymbol{P}}^{(l)}(k)} \right\}_{i},$$
(9)

where the last term corresponds to column *i* of the square root of the enclosed matrix. The initial condition is given by $\chi_0^{(l)}(k) = \hat{x}^{(l)}(k)$. The corresponding weights for the Sigma points $\{W_i\}_{i=1}^{2n_x}$ are given by $W_0^{(l)} = \kappa/(n_x + \kappa)$ and $W_i^{(l)} = 1/(2(n_x + \kappa))$, where κ is a scaling parameter. The sigma points are propagated through the state model (Eq. (1)) to generate the predicted sigma points $\chi_i^{(l)}(k+1|k) = f(\chi_i^{(l)}(k))$, which are then propagated through the local observation model to generate the predicted observation sigma points $\mathcal{Z}_i^{(l)}(k+1|k) = g^{(l)}(\chi_i^{(l)}(k+1|k))$. Based on the predicted sigma points, node *l* computes an estimate of its local posterior as [9]

$$\begin{aligned} \boldsymbol{x}_{\text{UKF}}^{(l)}(k+1) &= \boldsymbol{x}_{\text{UKF}}^{(l)}(k+1|k) + \boldsymbol{\mathcal{K}}^{(l)}(k) \left(\boldsymbol{z}^{(l)}(k+1) - \boldsymbol{z}_{\text{UKF}}^{(l)}(k+1|k) \right) (10) \\ \boldsymbol{P}_{\text{UKF}}^{(l)}(k+1) &= \boldsymbol{P}_{\text{UKF}}^{(l)}(k+1|k) - \boldsymbol{\mathcal{K}}^{(l)}(k) \boldsymbol{P}_{zz}^{(l)}(k+1|k) \boldsymbol{\mathcal{K}}^{(l)}{}^{T}_{(k)} \quad (11) \\ \text{where the Kalman gain is } \boldsymbol{\mathcal{K}}^{(l)}(k) &= \boldsymbol{P}_{xz}^{(l)}(k+1|k) \boldsymbol{P}_{zz}^{(l)}(k+1|k) \boldsymbol{P}_{zz}^{(l)}(k+1|k)^{-1}. \end{aligned}$$

Step 2. Consensus Step: Based on the Chong-Mori-Chang track-fusion [32], the local statistics $(\{\boldsymbol{x}_{\text{UKF}}^{(l)}(k+1), \boldsymbol{P}_{\text{UKF}}^{(l)}(k+1)\}_{l=1}^{N})$ computed in Step 1 are fused cooperatively to compute the statistics of the local proposal distribution (assumed Gaussian) as follows

$$\hat{\boldsymbol{P}}_{\text{UKF}}^{(l,\text{Global})^{-1}}(k+1) = \sum_{j=1}^{N} [\boldsymbol{P}_{\text{UKF}}^{(j)}(k+1)]^{-1} - [\boldsymbol{P}_{\text{UKF}}^{(j)}(k+1|k)]^{-1} + [\boldsymbol{P}_{\text{UKF}}^{(l)}(k+1|k)]^{-1}$$
(12)

$$\hat{\boldsymbol{x}}_{\text{UKF}}^{(l.\,\text{Global})}(k+1) = \tag{13}$$

$$\begin{split} & \left[\boldsymbol{P}_{\text{UKF}}^{(l,\text{thodal})}(k\!+\!1) \right]^{-1} \left[\left[\boldsymbol{P}_{\text{UKF}}^{(l)}(k+1|k) \right]^{-1} \boldsymbol{x}_{\text{UKF}}^{(l)}(k+1|k) \right] \\ & + \sum_{j=1}^{N} \left[\boldsymbol{P}_{\text{UKF}}^{(j)}(k\!+\!1) \right]^{-1} \boldsymbol{x}_{\text{UKF}}^{(j)}(k\!+\!1) - \left[\boldsymbol{P}_{\text{UKF}}^{(j)}(k\!+\!1|k) \right]^{-1} \boldsymbol{x}_{\text{UKF}}^{(j)}(k\!+\!1|k) \right], \end{split}$$

where the summation terms are obtained by iterating the following average consensus [21]-[23] equations

$$\boldsymbol{x}_{c}^{(l)}(t+1) = \boldsymbol{x}_{c}^{(l)}(t) + \epsilon \sum_{j \in \aleph^{(l)}} (\boldsymbol{x}_{c}^{(j)}(t) - \boldsymbol{x}_{c}^{(l)}(t)) \quad (14)$$

$$\boldsymbol{P}_{c}^{(l)}(t+1) = \boldsymbol{P}_{c}^{(l)}(t) + \epsilon \sum_{j \in \aleph^{(l)}} (\boldsymbol{P}_{c}^{(j)}(t) - \boldsymbol{P}_{c}^{(l)}(t)), (15)$$

with $\epsilon \in (0, 1/\Delta_{\mathcal{G}})$ [21] and t denotes the consensus iteration index. The initial conditions for the consensus runs are

$$\begin{aligned} \boldsymbol{P}_{c}^{(l)}(t=0) &= \left[\boldsymbol{P}_{\text{UKF}}^{(j)}(k+1)\right]^{-1} - \left[\boldsymbol{P}_{\text{UKF}}^{(j)}(k+1|k)\right]^{-1} \ (16) \\ \boldsymbol{x}_{c}^{(l)}(t=0) &= \left[\boldsymbol{P}_{\text{UKF}}^{(j)}(k+1)\right]^{-1} \boldsymbol{x}_{\text{UKF}}^{(j)}(k+1) \end{aligned}$$

$$P_{c}^{(j)}(t=0) = [\boldsymbol{P}_{\text{UKF}}^{(j)}(k+1)]^{-1}\boldsymbol{x}_{\text{UKF}}^{(j)}(k+1) - [\boldsymbol{P}_{\text{UKF}}^{(j)}(k+1|k)]^{-1}\boldsymbol{x}_{\text{UKF}}^{(j)}(k+1|k).$$
(17)

The consensus converges asymptotically [21] with $\{P_c^{(l)}(t), x_c^{(l)}(t)\}$ converging to N times the respective summation terms in Eqs. (12) and (13), respectively. In networks with intermittent connectivity, the number of consensus iterations t is limited and convergence of the consensus step is not guaranteed. Designed as a consensus plus innovation distributed estimator, the CI/DUPF does not require the consensus step to converge and even with a limited number of consensus iterations, the CI/DUPF performs reasonably well as is illustrated in the Monte Carlo simulations. However, when consensus is allowed to converge, the CI/DUPF improves on its performance and overlaps with the centralized particle filter [15].

Step 3. Generating the Particles: Given the proposal distribution $\tilde{\mathbb{X}}_{i}^{(l)}(k+1) \sim \mathcal{N}(\hat{\boldsymbol{x}}_{\mathrm{UKF}}^{(l,\mathrm{Global})}(k+1), \hat{\boldsymbol{P}}_{\mathrm{UKF}}^{(l,\mathrm{Global})}(k+1))$, node l, generates $N_{p}^{(l)}$ random particles $\tilde{\mathbb{X}}_{i}^{(l)}(k+1)$, which completes the consensus update of the CI/DUPF.

3.2.2. Innovation Update

The innovation update step computes the weights $W_i^{(l)}(k+1)$ corresponding to the local particles $\tilde{\mathbb{X}}_i^{(l)}(k+1)$ generated in the consensus update step. The global weight update equation

$$W_{i}^{(l)}(k+1) \propto \frac{P(\boldsymbol{z}(k+1)|\tilde{\mathbb{X}}_{i}^{(l)}(k+1))P(\tilde{\mathbb{X}}_{i}^{(l)}(k+1)|\mathbb{X}_{i}^{(l)}(k))}{\mathcal{N}(\boldsymbol{x}_{\mathsf{UKF}}^{(l,\mathsf{Global})}(k+1), \boldsymbol{P}_{\mathsf{UKF}}^{(l,\mathsf{Global})}(k+1))},$$
(18)

requires pointwise evaluation of the global likelihood function $P(\mathbf{z}(k+1)|\tilde{\mathbb{X}}_i^{(l)}(k+1))$ for each local particle $\tilde{\mathbb{X}}_i^{(l)}(k+1)$. Since the global observation vector $\mathbf{z}(k+1)$ is not available locally, an alternative weight update approach is required. Two separate cases are considered for the innovation update step.

Case 1 – Single Consensus Iteration: Each node can communicate only once with its neighboring nodes between two consecutive iterations of the CI/DUPF. In other words, only one consensus iteration is allowed, $t \leq 1$. As in Eq. (8) for linear systems [26], node *l* collects the observations of its neighboring nodes (i.e., $z^{(j)}(k+1)$, for $j \in \mathbb{N}^{(l)}$) and updates its local weights as

$$W_{i}^{(l)}(k+1) \propto \frac{P(\boldsymbol{z}^{(j \in \aleph^{(l)})}(k+1) | \tilde{\mathbb{X}}_{i}^{(l)}(k+1)) P(\tilde{\mathbb{X}}_{i}^{(l)}(k+1) | \mathbb{X}_{i}^{(l)}(k))}{\mathcal{N}(\boldsymbol{x}_{\mathsf{UKF}}^{(l,\mathsf{Global})}(k+1), \boldsymbol{P}_{\mathsf{UKF}}^{(l,\mathsf{Global})}(k+1))}.$$
(19)

In this case, node l needs to evaluate the likelihood function of its neighboring nodes and therefore requires prior knowledge of the observation model $g^{(j)}(\cdot), j \in \aleph^{(l)}$, of its neighboring nodes.

Case 2 – Multiple Consensus Iterations: Each node can communicate with its neighbors more than once between two consecutive iterations of the CI/DUPF. Still, a limited number of consensus iterations t are allowed. In the Monte Carlo simulations, we consider t = 2 and t = 3. The global likelihood function required to compute Eq. (18) is factorized as

$$P(\boldsymbol{z}(k)|\boldsymbol{x}(k)) \triangleq \mathcal{T}_1(\boldsymbol{z}(k))\mathcal{T}_2(\mathcal{S}(\boldsymbol{z}(k)), \boldsymbol{x}(k)), \qquad (20)$$

such that it can be evaluated distributively using average consensus. Since Term $\mathcal{T}_1(\boldsymbol{z}(k))$ in (20) is independent of the state $\boldsymbol{x}(k)$, it is considered as the normalization constant. Term $\mathcal{T}_2(\mathcal{S}(\boldsymbol{z}(k)), \boldsymbol{x}(k))$ depends on state $\boldsymbol{x}(k)$ and the global sufficient statistics (GSS) $\mathcal{S}(\boldsymbol{z}(k)) = \sum_{l=1}^{N} \mathcal{Y}^{(l)}(k)$ with $\mathcal{Y}^{(l)}(k)$ denoting the local sufficient statistics (LSS). The innovation update in the CI/DUPF is, therefore, a two step procedure.

- The LSSs are first computed locally. The means of the LSSs at the local nodes are then computed by running average consensus algorithms and used to derive the corresponding GSSs.
- (ii) Given the GSSs S(z(k)), Eq. (20) is used to compute P(z(k+1)|X̃^(l)_i(k+1)) for each particle X̃^(l)_i(k+1). Each node then updates the weights of the localized particle filters based on the updated GSSs using (19).

Ideally, the consensus runs computing the GSSs from LSSs should be allowed to converge. In networks with intermittent connectivity, we limit the GSS consensus iterations per consensus run implying that the likelihood function at each node is based on its own observation and those from neighboring nodes within a *t* hops radius. Intuitively, this is a reasonable assumption in geographically dispersed networks with localized state models and sparse observations. Finally, we note that the CI/DUPF is restricted to applications where the global likelihood function is factorizable and can be expressed in the form (20). Several tracking applications, e.g., bearing only tracking (BOT) and joint range/bearing tracking satisfy factorization (20). See [14, 15] where the former reference with exponential likelihood approximates the observation model to satisfy (20).

Example: To provide insight into the nature of the LSSs and GSSs, consider the over-simplified sensor network included for illustration with identical sensors and observations corrupted by Gaussian noise

$$Z^{(l)}(k) = \boldsymbol{g}(\boldsymbol{x}(k)) + \boldsymbol{\zeta}^{(l)}(k), \qquad (21)$$

for $(1 \le l \le N)$, where $\boldsymbol{\zeta}^{(l)}(k) \sim \mathcal{N}(0, \sigma^{(l)^2}(k))$. Expressing the global likelihood as

$$P(\boldsymbol{z}(k)|\boldsymbol{x}(k)) \propto \exp\Big\{-\sum_{l=1}^{N} \frac{Z^{(l)^{2}}(k)}{2\sigma^{(l)^{2}}(k)}-$$

$$\boldsymbol{g}^{2}(\boldsymbol{x}(k))\sum_{l=1}^{N} \frac{1}{2\sigma^{(l)^{2}}(k)}+\boldsymbol{g}(\boldsymbol{x}(k))\sum_{l=1}^{N} \frac{Z^{(l)}(k)}{\sigma^{(l)^{2}}(k)}\Big\},$$
(22)



Fig. 1. (a) Sensor placement along with target's trajectory. (b) Comparison between the centralized particle filter, DPF [15], CI/DUPF, distributed UKF [12], and Gu *et al.* [10] along with the PCRLB [5].

node l has three LSSs leading to the following GSSs

$$G_{1}(k) = \sum_{l=1}^{N} \underbrace{\frac{Z^{(l)^{2}}(k)}{2\sigma^{(l)^{2}}(k)}}_{\mathcal{Y}_{1}^{(l)}(k)}, \quad G_{2}(k) = \sum_{l=1}^{N} \underbrace{\frac{1}{2\sigma^{(l)^{2}}(k)}}_{\mathcal{Y}_{2}^{(l)}(k)}$$

and $G_{3}(k) = \sum_{l=1}^{N} \underbrace{\frac{Z^{(l)}(k)}{2\sigma^{(l)^{2}}(k)}}_{\mathcal{Y}_{2}^{(l)}(k)}.$

The GSSs are computed by running three average consensus runs on the corresponding LSSs across the network from which the global likelihood is computed. Derivation of LSSs and GSSs for more complex tracking applications, e.g. BOT, are covered in [14, 15].

4. EXPERIMENTAL RESULTS

In this section, several distributed BOT problems [15, 27] are simulated to quantify the performance of the proposed CI/DUPF. The global likelihood in the BOT problem consists of six GSSs which are computed from the summation of their corresponding LSSs. Please refer to [15] for details. We consider an agent network comprising of N nodes distributed randomly in a $(15 \times 15)m^2$ region. Each node only communicates directly with its neighboring nodes within a connectivity radius of $\sqrt{2\log(N)/N}$ meters. In addition, the network is assumed to be connected with each node linked to at least one other node in the network. The target's motion model f(x(k))is considered to be a nonlinear clockwise coordinated turn kinematic motion model [27]. In each run, the target starts its track from coordinates {10, 10}, with the initial course set at -110° with the standard deviation $\sigma_{\boldsymbol{\xi}(k)} = 1.6 \times 10^{-2}$ meter for the process noise. Both process and observation noises are assumed to be normally distributed. Furthermore, the observation noise is assumed to be state dependent such that the bearing noise variance $\sigma^2_{\zeta^{(l)}}(k)$ at node l depends on the distance $D^{(l)}(\boldsymbol{x}(k))$ between the observer and target, i.e., $\sigma_{\zeta^{(l)}}^{2}(k) = B_m [D^{(l)}(\boldsymbol{x}(k))]^2 + 0.11 D^{(l)}(\boldsymbol{x}(k))$. Initialization performed at the distance $D^{(l)}(\boldsymbol{x}(k)) = 0.11 D^{(l)}(\boldsymbol{x}(k))$. tion performed at each node follows [27] with local filters initialized with observation noise variances $\sigma_{\theta} = 2.5^{\circ}$. In all implementations, $N_s = 1000$ particles are used. The performance metric used is the root mean square position error (RMSE) [27] given by

$$\mathbf{RMSE}(k) = \sqrt{\frac{1}{n_{\mathrm{MC}}} \frac{1}{N} \sum_{j=1}^{n_{\mathrm{MC}}} \sum_{l=1}^{N} (X_j(k) - \hat{X}_j^{(l)}(k))^2 + (Y_j(k) - \hat{Y}_j^{(l)}(k))^2}$$

Fig. 2. (a) Comparison between the CI/DUPF and centralized particle filter for a reduced number of consensus iterations. (b) RMS error at iteration k = 20 as a function of the network size for consensus iterations t = 1, 2, 3.

where $n_{\rm MC}$ is the number of Monte Carlo simulations.

Scenario 1: The consensus step is allowed to converge between two iterations of the localized filters. Scenario 1 compares the performance of the CI/DUPF with: (i) Centralized approach; (ii) DPF implementation from [15], referred to as the CSS/DPF; (iii) Distributed unscented Kalman filter [12], and; (iv) Distributed particle filter proposed in [10], referred to as Gu et al. Fig. 1(a) shows one realization of the sensor placement along with the target trajectory. Due to the state-dependent noise variance, the signal to noise ratio (SNR) is time-varying and differs from one node to the other depending on the location of the target. The SNRs at different nodes varies from 16dB to 29dB. In Fig. 1(b) the RMS error corresponding to the CI/DUPF is compared to schemes (i) to (iv), and with the Posterior Cramer Rao lower bound (PCRLB) [5]. It is observed that the performance of the CI/DUPF is fairly close to that of the centralized particle filter and approaching the PCRLB.

Scenario 2: The performance of the proposed CI/DUPF using a limited number $\{2, 3\}$ of consensus iterations is compared with that of the centralized particle filter. The consensus runs are stopped abruptly after a fixed number of iterations without allowing them to converge. The three remaining distributed implementations diverge if the consensus algorithm is not allowed to converge and are not plotted here since their RMS errors go out of scale. The results are plotted in Fig. 2(a). The CI/DUPF follows the centralized particle filter in all cases. The RMS errors of the CI/DUPF remains bounded and approaches that of the centralized filter as t increases.

Scenario 3: evaluates the performance of the CI/DUPF as a function of number of active nodes in the network. Several networks with a different number $N = \{10, 20, 30, 40, 50\}$ of sensor nodes are simulated. The centralized particle filter, the CI/DUPF with two and three iterations per consensus run are compared in Fig. 2(b). In the simulated networks, it is observed that the CI/DUPF follows the centralized particle filter with a limited number of iterations per consensus run for all networks.

5. SUMMARY

A consensus plus innovation based distributed particle filter CI/DUPF is proposed that extends the linear consensus plus innovation framework to nonlinear estimators. The CI/DUPF does not require the consensus step to converge making it suitable for environments with intermittent network connectivity. In our Monte Carlo simulations, the RMS error of the CI/DUPF remains bounded and approaches that of the centralized particle filter as the number of consensus iterations are increased. The role that the sparsity of the network plays in the CI/DUPF performance is being pursued as future work.

6. REFERENCES

- T. Zhao and A. Nehorai, "Distributed Sequential Bayesian Estimation of a Diffusive Source in Wireless Sensor Networks," *IEEE Tran. Sig. Proc.*, vol. 55, no. 4, pp. 1511-1524, 2007.
- [2] K. Zhou, S.I. Roumeliotis, "Optimal Motion Strategies for Range-Only Constrained Multisensor Target Tracking," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1168-1185, 2008.
- [3] J.C. Hassab, Underwater Signal and Data Processing, CRC Press, Boca Raton, FL 1989.
- [4] S. Blackman, R. Popoli, "Design and Analysis of Modern Tracking Systems," Artech House, 1999.
- [5] A. Mohammadi and A.Asif, "Distributed Particle Filter Implementation with Intermittent/Irregular Consensus Convergence," *IEEE Transactions on Signal Processing*, vol. 61, no. 10, pp. 2572-2587, 2013.
- [6] O. Hlinka, F. Hlawatsch, and P.M. Djuric, "Distributed Particle Filtering in Agent Networks: A Survey, Classification, and Comparison," *IEEE Signal Processing Magazine*, vol. 30, no. 1, pp. 61-81, Jan. 2013.
- [7] A. Mohammadi and A.Asif, "Full Order Nonlinear Distributed Estimation in Intermittently Connected Networks", *IEEE Inter. Con. on Acoustics, Speech, & Signal Processing (ICASSP)*, pp. 6327-6331, 2013.
- [8] A. Mohammadi and A.Asif, "Consensus-based Particle Filter Implementations for Distributed nonlinear Systems", *Nonlin*ear Estimation and Applications to Industrial Systems Control, Editor G. Rigatos, 2012.
- [9] A. Mohammadi and A. Asif, "Consensus-Based Distributed Unscented Particle Filter," in *IEEE Workshop on Statistical Signal Processing (SSP)*, pp. 237-240, 2011.
- [10] D. Gu, J. Sun, Z. Hu, and H. Li, "Consensus based distributed particle filter in sensor networks," *IEEE Int. Con. on Information and Automation*, pp. 302-307, 2008.
- [11] M.J. Coates and B.N. Oreshkin, "Asynchronous distributed particle filter via decentralized evaluation of gaussian products," *Proc. IEEE Int. Con. on Info. Fusion*, 2010.
- [12] A. Simonetto, T. Keviczky, R. Babuska, "Distributed nonlinear Estimation for Robot Localization using Weighted Consensus", in *IEEE Inter. Con. on Robotics and Automation*, pp. 3026-3031, 2010.
- [13] S. Farahmand, S. I. Roumeliotis, and G. B. Giannakis, "Setmembership constrained particle filter: Distributed adaptation for sensor networks, *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4122-4138, 2011.
- [14] O. Hlinka, O. Slucciak, F. Hlawatsch, P.M. Djuric, and M. Rupp, "Likelihood Consensus and Its Application to Distributed Particle Filtering," *IEEE Transactions on Signal Processing*, vol. 60, no. 8, pp. 4334-4349, 2012.
- [15] A. Mohammadi and A. Asif, "A Constraint Sufficient Statistics based Distributed Particle Filter for Bearing Only Tracking," *IEEE Int. Con. on Communications (ICC)*, pp. 3670-3675, 2012.
- [16] D. Ustebay, M. Coates, and M. Rabbat, "Distributed auxiliary particle filters using selective gossip," *IEEE Int. Con. on Acoustics, Speech, & Signal Processing (ICASSP)*, pp. 3296-3299, 2011.

- [17] C.J. Bordin, M.G.S. Bruno, "Consensus-based distributed particle filtering algorithms for cooperative blind equalization in receiver networks," *IEEE Int. Con. on Acoustics, Speech, & Signal Processing (ICASSP)*, pp. 3968-3971, 2011.
- [18] O. Hlinka and F. Hlawatsch, "Distributed particle filtering in the presence of mutually correlated sensor noises," *IEEE Int. Con. on Acoustics, Speech, & Signal Processing (ICASSP)*, pp. 6269-6273, 2013.
- [19] C.J. Bordin and M.G.S. Bruno, "A new minimum-consensus distributed particle filter for blind equalization in receiver networks," *IEEE Int. Con. on Acoustics, Speech, & Signal Processing (ICASSP)*, pp. 6323-6326, 2013.
- [20] D. Gu, "Distributed particle filter for target tracking," in *IEEE Int. Conference on Robotics and Automation*, pp. 3856–3861, 2007.
- [21] R. Olfati-Saber J. A. Fax and R. M. Murry, "Consensus and cooperation in networked multi-agent systems," in *Proceedings* of the IEEE, 2007.
- [22] R. Carli, A. Chiuso, L. Schenato, and S. Zampieri, "Distributed Kalman filtering based on consensus strategies," *IEEE Journal* on Selected Areas in Communications, vol. 26, no. 4, pp. 622-633, 2008.
- [23] A. G. Dimakis, S. Kar, J. M. F. Moura, M. G. Rabbat and A. Scaglione, "Gossip algorithms for distributed signal processing," *Proceedings of the IEEE*, vol.98, pp. 1847-1864, 2010.
- [24] S. Kar, J.M.F. Moura, K. Ramanan, "Distributed Parameter Estimation in Sensor Networks: Nonlinear Observation Models and Imperfect Communication," *IEEE Transactions on Information Theory*, vol. 58, no. 6, pp. 3575-3605, 2012.
- [25] S. Kar, J.M.F. Moura, "Consensus + innovations distributed inference over networks: cooperation and sensing in networked systems," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 99-109, 2013.
- [26] U.A. Khan and A. Jadbabaie, "Networked estimation under information constraints," arXiv preprint arXiv:1111.4580, 2011.
- [27] M.S. Arulampalam, B. Ristic, N. Gordon, and T. Mansell, "Bearings-only Tracking of Maneuvering Targets using Particle Filters," *EURASIP App. Sig. Proc.*, vol. 15, pp. 2351-2365, 2004.
- [28] K. Zhou, S.I. Roumeliotis, "Multirobot Active Target Tracking With Combinations of Relative Observations," *IEEE Transactions on Robotics*, vol. 27, no. 4, pp. 678-695, 2011.
- [29] P. m. Djuric, J. h. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. f. Bugallo, J. Miguez, "Particle Filtering", *IEEE Signal Processing Magazine*, vol. 20, no. 5, pp. 19-38, 2003.
- [30] N. Gordon, M. Sanjeev, S. Maskell, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, pp. 174-187, 2002.
- [31] R. Van der Merwe, A. Doucet, N. de Freitas and E. Wan, "The unscented particle filter," Tech. Rep., Cambridge University, 2000.
- [32] C.Y. Chong, S. Mori, and K.C. Chang, *Multi-target Multi-sensor Tracking*, chapter Distributed Multi-target Multi-sensor Tracking, Artech House, pp. 248–295, 1990.