# Particle Filter Based Average Consensus Target Tracking with Preservation of Network Connectivity

[†]Z. Kan, [‡]E. L. Pasiliao Jr, [‡]J. W. Curtis, [†]W. E. Dixon
[†]Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, FL 32611-6250, USA.
[‡]Air Force Research Laboratory, Munitions Directorate, Eglin AFB, FL 32542, USA.
Email: kanzhen0322@ufl.edu, pasiliao@eglin.af.mil, curtisjw@eglin.af.mil, wdixon@ufl.edu

***Abstract*— In this paper a mobile sensor network is tasked with cooperatively tracking a moving target. Each sensor runs a distributed particle filter to estimate the states of the target. Since some sensors may experience occasional failures or do not trust their own observations, a general case is considered, where only a portion of the sensors have the observations of the target. An average consensus algorithm is developed to allow each sensor to dynamically track the global mean of all local estimates about the target from the network by using communication with neighboring sensors only. Due to the limited communication capability on each sensor, the consensus algorithm and the cooperative objective of tracking the target must be accomplished while ensuring that the overall network remains connected. Based on the estimated global mean, a potential field based distributed motion control law is then designed to drive the sensors to track the target cooperatively, while preserving the network connectivity and avoiding collision during the mission operation. Simulation results verify the effectiveness of the proposed approach.**

## I. Introduction

Cooperative mobile sensor networks provide versatile platforms for both civilian and military applications. Collaborative target tracking by the mobile sensor network is an enabling technology for many of these applications. Compared with target tracking by a single sensor, a team of autonomous sensors operating in a coordinated fashion can provide benefits such as robustness in mission completion, accuracy and efficiency in information gathering, and applicability in complex and dynamic environment.

To estimate the trajectory of a moving target, Kalman filtering is one of the most computationally efficient algorithms [1]. However, Kalman filtering requires linear system dynamics and zero-mean Gaussian measurement noise. A particle filter makes no assumption on the linearity of system dynamics or the distribution of measurement noise, making it more suitable for tracking a target with nonlinear dynamics in a complex environment [2]–[4]. Several results are developed in the work of [5] and [6] using distributed particle filters, where global statistics of particles are accumulated by adding local statistics from each sensor in the network. However, transmitting the local statistics of particles among sensors can consume energy and communication bandwidth. To reduce the energy and communication bandwidth consumption, the global statistics of all particles are estimated in this work for each sensor by using a distributed average consensus algorithm, where each sensor updates its local estimate of the global statistics by communicating with neighboring sensors only.

The consensus problem has attracted significant research attention recently (see [7] and [8] for a comprehensive review and tutorial), where systems are generally required to agree upon certain quantities of interest. The average consensus problem is one useful form that requires systems to achieve an consensus on an average of quantities of interest. In contrast to static average consensus where individual states reach an agreement on the average of their initial states, dynamic average consensus requires the individual states agree on the average of time-varying reference signals, such as the results developed in [9]–[13]. By using standard frequency-domain techniques, a consensus algorithm is developed to track the average of a ramp reference in [9]. Based on the work of [9], proportional and proportional-integral consensus algorithms are developed to track the average of constant or slowly-varying reference inputs with sufficiently small steady-state error in [10]. The work of [11] studies the dynamic average consensus algorithm with an assumption that the nonlinear model which generates the time-varying reference is known, and the algorithm designed in [12] can track with some bounded steady-state error the average of a common reference with bounded derivative. The results developed in [9]–[12] were extended to a class of discrete-time systems to track the average of their reference inputs in [13].

In practice, some sensors may experience occasional failures during the mission operation (e.g., the target may move out of the field of view of a sensor when image feedback is used, or the sensor does not trust what it observes). However, the results developed in [9]–[13] are not applicable when only a portion of the sensors are able to sense the target. Hence, a novel consensus algorithm is needed to allow each sensor to dynamically estimate the global statistics by communicating with neighboring sensors when only some neighbors have target measurements. Moreover, network connectivity is a key assumption in [9]–[13], in the sense that network connectivity enables information exchange among sensors all the time. Since, in reality, wireless communication between sensors can be impacted due to path loss, shadowing, and/or multi-path

[1]This research is supported in part by a grant from the Air Force Research Laboratory, Munitions Directorate at Eglin AFB.

978-1-4673-3/12/$31.00 

fading, which depends on the distance between the sensors, desired quality of service may not be guaranteed all the time due to the motion of the sensors, resulting in a potential to partition the network. To ensure network connectivity, sensors are required to stay within a certain distance to maintain sufficient communication quality. In addition, the task of target tracking by a group of mobile sensors also requires the sensors to coordinate their motion by communicating with neighboring sensors. Therefore, it is paramount to design a cooperative motion control law for sensors to track the moving target while preserving the network connectivity.

In this paper, each mobile sensor runs a local particle filter to estimate the position of a moving target. To improve the accuracy of the estimation and reduce the inter-sensor communication, a distributed average consensus algorithm is developed to allow each sensor to dynamically estimate the time-varying global statistics (i.e., the average of local estimates from all sensors) by communicating with neighboring sensors only. The consensus algorithm developed in this work is distinct from results such as [9]–[13] since it is applicable to a more general case where only a portion of the sensors are able to sense the target, which accounts for occasional missing information due to occlusions or obstacles in the complex environment. This new feature also provides robustness in the sense that failures of any sensor will not lead to failures of the consensus algorithm. Since maintaining a certain relative position from the target may result in the best observation of the target while reducing the probability of being detected [14], to improve the target tracking performance, a potential field based distributed control law is then developed to position the mobile sensors at a prespecified position relative to the target based on our previous work in [15]. In addition to tracking the target, the developed control law also ensures network connectivity and collision avoidance among sensors during the mission operation.

## II. Problem Formulation

Consider a network of $N$ mobile sensors tracking a moving target cooperatively, where the target moves in a workspace $\mathcal{F}$ according to the following kinematics:

$$\dot{x}_T = f\left(x_T, \nu_T\right), \tag{1}$$

where $x_T\left(t\right) \in \mathbb{R}^2$ denotes the target position, $f\left(\cdot\right)$ is a time-varying nonlinear function describing the motion of the target, and $\nu_T \in \mathbb{R}^2$ denotes a known noise process. Each sensor is represented as a point mass, moving according to the kinematics

$$\dot{x}_i = u_i, \ i = 1, \cdots, N \tag{2}$$

where $x_i\left(t\right) \in \mathbb{R}^2$ denotes the position of sensor $i$, and $u_i\left(t\right) \in \mathbb{R}^2$ represents its control input. The observation of the target is obtained by sensor $i$ as

$$z_i\left(t\right) = x_T\left(t\right) + w_i(t), \tag{3}$$

where $z_i\left(t\right) \in \mathbb{R}^2$ denotes the target measurements, and $w_i(t)$ represents a known measurement noise.

To perform the mission of target tracking, the sensors coordinate their motion collaboratively and the interaction of the networked sensors is modeled as an undirected graph $\mathcal{G}(t) = (\mathcal{V}, \mathcal{E}(t))$, where $\mathcal{V} = \{1, \cdots, N\}$ denotes the set of sensors, and $\mathcal{E}(t)$ denotes a set of time-varying edges. In graph $\mathcal{G}(t)$, each node $i \in \mathcal{V}$ represents a sensor located at a position $x_i$. Since the communication quality depends on the relative distance between nodes, $d_{ij} \triangleq \|x_i - x_j\| \in \mathbb{R}^+$, each sensor is assumed to have a limited communication capability encoded by a disk area with radius $R$, which implies that two sensors are able to exchange local estimates about the target through communication as long as they stay within a distance of $R$. The sensor is assumed to have a perfect knowledge of its own position states. An undirected edge $(i, j) \in \mathcal{E}$ exists between node $i$ and $j$ in $\mathcal{G}(t)$ if $d_{ij} \leq R$. The neighbor set of node $i$ is denoted as $\mathcal{N}_i = \{j \mid (i, j) \in \mathcal{E}\}$, which includes nodes that can be reached for communication. Since the graph $\mathcal{G}\left(t\right)$ is undirected, $i \in \mathcal{N}_j \Longleftrightarrow j \in \mathcal{N}_i$ for $\forall i, j \in \mathcal{V}, i \neq j$. The adjacency matrix $A = [a_{ij}] \in \mathbb{R}^{N \times N}$ is defined such that $a_{ij}$ is a positive weight if $(i, j) \in \mathcal{E}$ and $a_{ij} = 0$ if $(i, j) \notin \mathcal{E}$. A graph is balanced if $\sum_{j=1}^{N} a_{ij} = \sum_{j=1}^{N} a_{ji}$. The Laplacian matrix is then defined as $L = [l_{ij}] \in \mathbb{R}^{N \times N}$, where $l_{ii} = \sum_{j \neq i} a_{ij}$, and $l_{ij} = -a_{ij}$ when $i \neq j$. For an undirected graph, the Laplacian matrix $L$ is a symmetric positive semidefinite matrix. Since $L$ has zero row sums, $\mathbf{1} = [1, \cdots, 1]^T \in \mathbb{R}^N$ is always a right eigenvector of $L$ corresponding to the eigenvalue of 0 (i.e., $L\mathbf{1} = 0$).

Each sensor runs a local particle filter to estimate the state of the moving target. Since the local estimate can be corrupted by noise, each sensor dynamically estimates the target position by averaging all local estimates from the group of sensors to obtain an accurate estimate of the target. In addition, when performing in a complex environment, some sensors may experience occasional failure, and thus have no access to the measurement of the target. In this work, a general case is considered where only a subset of the sensors, defined as $\mathcal{V}_l \subseteq \mathcal{V}$, can sense the target. Therefore, one objective in this work is to design a distributed dynamic average consensus algorithm for each sensor to locally estimate the global mean by using communication within neighboring sensors in which only a portion have the observations of the target.

It is stated in [14] that maintaining a certain distance from the target may result in the best observation of the target while reducing the probability of being detected. To obtain the best observation of the target, each sensor $i$ is required to maintain a prespecified relative position $s_i \in \mathbb{R}^2$ to the target. Moreover, due to limited communication capability, node $i$ can only communicate with nodes within its communication range. Once node $j$ moves out of the sensing and communication zone of node $i$, node $i$ will no longer share information with node $j$ directly, which may lead to mission failure. As a result, maintaining connectivity of the underlying graph is necessary. Hence, another objective is to design a cooperative control law to maintain the sensors at $s_i$, while ensuring the network connectivity and collision avoidance among sensors.

To achieve these goals, the following assumptions are required in the subsequent development.

**Assumption 1**: It is assumed that the subset $\mathcal{V}_l$ forms a balanced graph and each failure sensor $i$ has at least one neighboring sensor $j$ such that $j \in \mathcal{N}_i \cap \mathcal{V}_l$.

**Assumption 2**: The target state $x_T(t)$ and its derivative $\dot{x}_T(t)$ are assumed to be bounded. It is unrealistic to achieve consensus for a target whose states change too rapidly, since it takes time for the local estimates about the target states to flow across the network.

**Assumption 3**: The $s_i$ assigned to the sensors is achievable, which indicates that the destination for each sensor will not meet any constraints, i.e., lead to the partition of the underlying graph or collision with other sensors.

## III. Consensus based Distributed Particle Filter

### A. Distributed Particle Filter

A standard particle filter [2]–[4] is applied on each sensor to estimate the states of the moving target. The key idea of a particle filter is to use a set of independent random samples, called particles, to represent the posterior distribution of the target, where these particles are modeled using the same dynamics as the target. As new sequence of sensor measurements arrives, the particles are re-allocated to update the state estimate.

To estimate the target state $x_T(t)$, each sensor $i \in \mathcal{V}$ applies a distributed particle filter to approximate the probability density function (pdf) of $x_T(t)$ by using a set of $M$ particles $\zeta_i^{(m)}(t) \in \mathbb{R}^2$, $m \in \{1, \cdots, M\}$. Each particle $\zeta_i^{(m)}(t)$ is associated with a weight $\omega_i^{(m)}(t)$, satisfying that $\sum_{m=1}^{M} \omega_i^{(m)} = 1$. The set of particles $\zeta_i^{(m)}(t)$ and their associated weights $\omega_i^{(m)}(t)$ can be used to approximate the target density $p(x_T(t)|z_i(t))$ by the following weighted sum of delta functions,

$$p(x_T(t)|z_i(t)) \approx \sum_{m=1}^{M} \omega_i^{(m)}(t)\,\delta\left(x_T(t) - \zeta_i^{(m)}(t)\right), \quad (4)$$

where $\delta(\cdot)$ denotes the delta measure. The particles $\zeta_i^{(m)}$ and the weights $\omega_i^{(m)}$ are continuously updated to represent the target density in real time, where particles $\zeta_i^{(m)}$ are propagated using the the dynamics of the target in (1), and weights $\omega_i^{(m)}$ are updated using the new sequence of sensor measurements $z_i(t)$ in (3) by an approach called importance sampling [3]. Since the variance of the importance weights increasing over time will lead to the degeneracy phenomenon, resampling [4] is used to eliminate particles with low importance weights by replicating the samples corresponding to higher importance weights. Further details are available in [2]–[4].

Each sensor $i$ maintains a set of $M$ particles $\omega_i^{(m)}$ and the associated weights $\zeta_i^{(m)}$ that are evolving with the sequence of measurements $z_i(t)$. Since the target density can be described by the set $\{\omega_i^{(m)}, \zeta_i^{(m)}\}$ in (4), the target state $x_T$ can be estimated for each sensor $i$ by computing the local mean $\bar{\zeta}_i(t) \in \mathbb{R}^2$ of the weighted particles as

$$\bar{\zeta}_i(t) = \sum_{m=1}^{M} \omega_i^{(m)}(t)\,\zeta_i^{(m)}(t). \quad (5)$$

Using (5), a local estimate of the target states is obtained on each sensor $i$.

### B. Dynamic Average Consensus

As described in Section III-A, each sensor $i$ estimates the target states using the local mean $\bar{\zeta}_i(t)$ in (5). Since a control decision made based on such local estimates may lead to an accumulated error, and hence deteriorate the target tracking performance, it is beneficial to take into account all the local estimates from the entire network of sensors. To improve the accuracy of the estimate, the goal in this section is to design a dynamic average consensus algorithm that allows each sensor to estimate the average of the local estimates from all sensors, called global mean, by using information exchange with its neighbors only, where the global mean $\bar{\zeta}(t)$ can be computed as $\bar{\zeta}(t) = \frac{1}{N}\sum_{i=1}^{N} \bar{\zeta}_i(t)$. Instead of transmitting the set of $M$ particles and associated weights across the entire network, only locally computed $\bar{\zeta}_i(t)$ is exchanged among sensors, which reduces power demands and communication bandwidth.

To estimate the time-varying global mean $\bar{\zeta}(t)$, a local estimator $p_i(t)$ is developed for each sensor $i$ such that the error

$$e_i(t) = p_i(t) - \bar{\zeta}(t) \quad (6)$$

approaches zero as $t \to \infty$ for $\forall i \in \mathcal{V}$. To regulate the tracking error in (6), a local estimator $p_i(t)$ for sensor $i$ is developed via the distributed dynamic average consensus algorithm

$$\begin{aligned} \dot{\xi}_i(t) &= -\gamma k_i \xi_i(t) - \sum_{j \in \mathcal{N}_i} a_{ij} k_j (p_i(t) - p_j(t)) \\ &\quad -(1-k_i)\alpha sgn\left[\sum_{j \in \mathcal{N}_i} a_{ij} k_j (p_i - p_j)\right] \quad (7) \\ p_i(t) &= \xi_i(t) + k_i \bar{\zeta}_i(t), \quad (8) \end{aligned}$$

where $\xi_i(t)$ is the internal state, $\bar{\zeta}_i(t)$ is the local mean of the target state which can be considered as an exogenous input to the system, $p_i(t)$ denotes a local estimator used to estimate the global mean $\bar{\zeta}(t)$, $\gamma$, $\alpha \geq 0$ are control gains, and $a_{ij}$ denotes the neighborhood between sensor $i$ and $j$ (i.e., $a_{ij} > 0$ if sensor $i$ is able to communicate with sensor $j$, and $a_{ij} = 0$ otherwise). The consensus algorithm in (7) and (8) only requires communication among neighboring nodes about their local mean $\bar{\zeta}_i(t)$ and a local estimate $p_i(t)$, and thus, is a distributed algorithm.

The developed algorithm in (7) and (8) is inspired by the work of [9] and [10]. However, a more general case is considered in our approach, where only a portion of the sensors observe about the target. Due to target motion or the random failure of sensors, some sensors may not be able to sense the target all the time or do not trust what it measured during some interval. The trust weight $k_i \in \{0, 1\}$

is introduced in (7) and (8) to indicate whether sensor $i$ can sense the target or trust its local observations. That is, $k_i = 0$ if sensor $i$ has no measurement of the target (i.e., $i \notin \mathcal{V}_l$), where $\mathcal{V}_l$ denotes the set of sensors that can sense the target, and $k_i = 1$ otherwise. Note that the proportional dynamic consensus developed in [10] is a particular case of algorithm (7) and (8), where every sensor can observe the target (i.e., $k_i = 1$ for $\forall i$, and $\mathcal{V}_l = \mathcal{V}$). Moreover, if $\gamma = 0$ and $\mathcal{V}_l = \mathcal{V}$, the consensus algorithm in (7) and (8) can be further reduced to the case discussed in [9].

The consensus algorithm in (7) and (8) can be separated into two parts based on whether the target can be sensed by sensor $i$ or not. Specifically, if sensor $i$ observes the target, then

$$\left\{ \begin{array}{c} \dot{\xi}_i = -\gamma \xi_i - \sum_{j \in \mathcal{N}_i} a_{ij} k_j \left(p_i - p_j\right) \\ p_i = \xi_i + \bar{\zeta}_i \end{array} \right. \quad \text{for } i \in \mathcal{V}_l, \quad (9)$$

for $i \notin \mathcal{V}_l$ otherwise,

$$\dot{p}_i = -\sum_{j \in \mathcal{N}_i} a_{ij} k_j \left(p_i - p_j\right) - \alpha sgn \left[ \sum_{j \in \mathcal{N}_i} a_{ij} k_j \left(p_i - p_j\right) \right]. \quad (10)$$

To show that $e_i(t)$ in (6) approaches zero as $t \to \infty$ for $\forall i$, the strategy is to first show that all sensors within the set $\mathcal{V}_l$ track the global mean $\bar{\zeta}(t)$ and then to show that the rest of the sensors $i$, $i \notin \mathcal{V}_l$, achieve consensus to those sensors within the set $\mathcal{V}_l$. For the simplicity of presentation, only one-dimensional state (e.g., the position of the target along a line) is considered in the following analysis. However, the results are valid for an $m$ dimensional case ($m > 1$) by introduction of the Kronecker product.

**Lemma 1:** *Since a target with constant or slowly varying velocity is considered, suppose that the exogenous input $\bar{\zeta}_i$ in (9) is bounded as $\left|\gamma \bar{\zeta}_i + \dot{\bar{\zeta}}_i\right| \leq \mu \in R^+$. Provided that the set $V_l$ forms a connected undirected balanced graph, then the consensus algorithm in (9) is Input to State Stable (ISS) and ensures that $p_i(t)$ tracks the global mean $\bar{\zeta}(t)$ for $\forall i \in V_l$.*

The proof of this result, omitted here, is based on the work of [10]. As shown in (9), when considering the sensors $i \in \mathcal{V}_l$ only, the estimator $p_i$ takes its local measurement $\bar{\zeta}_i$ as an input and updates the internal state $\xi_i$ by exchanging information with other neighboring sensors $j$ who can also observe the target (i.e., $k_j = 1$, for $j \in \mathcal{N}_i \cap \mathcal{V}_l$). Note that $p_i$ will not be affected by sensor $j$ which can not see the target, since $k_j = 0$ if $j \notin \mathcal{V}_l$. Hence, the consensus algorithm in (9) has a similar form to the proportional consensus estimator designed in [10]. By following a similar procedure in [10], it can be shown that the consensus algorithm in (9) is ISS and $p_i(t) \to \bar{\zeta}(t)$ as $t \to \infty$ for $\forall i \in \mathcal{V}_l$.

**Theorem 1**: *Suppose that the sensor network is connected in a way that the sensors $i \in V_l$ form a connected balanced graph, and each sensor $i \in \bar{V}_l$ has at least one neighboring sensor within the set $V_l$, where $\bar{V}_l$ denotes the complement of $V_l$ (i.e., $\bar{V}_l \cup V_l = V$). The consensus algorithm designed in (7) and (8) ensures that $e_i(t)$ in (6) approaches zero as $t \to \infty$ for $\forall i$ provided that $\alpha > \mu$.*

*Proof:* Given that the set $\mathcal{V}_l$ forms a connected balanced graph, it has been shown that sensor $i$ tracks the global mean $\bar{\zeta}(t)$ for $\forall i \in \mathcal{V}_l$ in Lemma 1. The rest of the sensors $i \in \bar{\mathcal{V}}_l$ also must be proven to track $\bar{\zeta}(t)$. Towards this end, let $\eta_i = p_i - \bar{\zeta}(t)$ for $i \in \bar{\mathcal{V}}_l$, where (10) can be written as

$$\begin{aligned} \dot{\eta}_i &= -\sum_{j \in \mathcal{N}_i} a_{ij} k_j \left(\eta_i - \eta_j\right) \\ &\quad -\alpha sgn \left[ \sum_{j \in \mathcal{N}_i} a_{ij} k_j \left(\eta_i - \eta_j\right) \right] - \dot{\bar{\zeta}}. \end{aligned} \quad (11)$$

The term $\sum_{j \in \mathcal{N}_i} a_{ij} k_j \left(\eta_i - \eta_j\right)$ in (11) can be further separated as

$$\begin{aligned} \sum_{j \in \mathcal{N}_i} a_{ij} k_j \left(\eta_i - \eta_j\right) &= \sum_{j \in \mathcal{N}_i \cap \bar{\mathcal{V}}_l} a_{ij} k_j \left(\eta_i - \eta_j\right) \\ &\quad + \sum_{j \in \mathcal{N}_i \cap \mathcal{V}_l} a_{ij} k_j \left(\eta_i - \eta_j\right) \end{aligned} \quad (12)$$

Substituting (12) into (11) and using the fact that $\eta_j = p_j - \bar{\zeta}(t) = 0$ if $j \in \mathcal{V}_l$, (11) can then be written in a matrix form as $\dot{\eta} = -M\eta - \alpha sgn(M\eta) - \mathbf{1}\dot{\bar{\zeta}}$, where $\eta$ is the column stack vector of $\eta_i$, and $M = L + D$ is a matrix with $L$ being the Laplacian matrix and $D$ being a diagonal matrix with the $i$th diagonal entry as $\sum_{j \in \mathcal{N}_i \cap \mathcal{V}_l} a_{ij} k_j$. Due to the properties of a Laplacian matrix, $L$ is a positive semi-definite matrix. Also, since each sensor $i \in \bar{\mathcal{V}}_l$ has at least one neighboring sensor $j$ within the set $\mathcal{V}_l$ (i.e., $j \in \mathcal{N}_i \cap \mathcal{V}_l$), at least one $a_{ij} k_j > 0$. Hence, $M$ is a symmetric positive definite matrix.

Consider a Lyapunov function candidate $V = \frac{1}{2}\eta^T M \eta$. Taking the time derivative of $V$ and using the assumption that $\left|\gamma \bar{\zeta}_i + \dot{\bar{\zeta}}_i\right| \leq \mu$ yields

$$\begin{aligned} \dot{V} &= \eta^T M \left( -M\eta - \alpha sgn(M\eta) - \mathbf{1}\dot{\bar{\zeta}} \right) \\ &\leq -\eta^T M M \eta - (\alpha - \mu) \|M\eta\|. \end{aligned}$$

Since the gain $\alpha$ is chosen such that $\alpha > \mu$, and $M$ is a positive definite matrix, it follows that $\dot{V}$ is negative definite, which ensures that $p_i(t) \to \bar{\zeta}(t)$ as $t \to \infty$ for $\forall i \notin \mathcal{V}_l$. Based on the results in Lemma 1, $p_i(t) \to \bar{\zeta}(t)$ as $t \to \infty$ for $\forall i \in \mathcal{V}$. ■

## IV. Motion Control

A distributed motion control law is developed for each mobile sensor to cooperatively track the target. Since maintaining a certain distance from the target may result in the best observation of the target while reducing the probability of being detected [14], it is desired to have each sensor $i$ maintain a prespecified relative position $s_i$ to the target. To track the target cooperatively, mobile sensors exchange information with neighboring sensors to estimate the target position by using the consensus algorithm designed in Section III. Since mobile sensors with limited communication capabilities are considered, network connectivity can be impacted due to the motion of the sensors. An artificial potential field based motion control law is developed to maintain each sensor at a prespecified $s_i$ with respect to the moving target while preserving the network connectivity and ensuring collision avoidance among sensors during the task operation.

Based on our previous work [15] and [16], a decentralized potential function $\varphi_i(t) : \mathcal{F} \to [0,1]$ for each sensor $i$ is designed as

$$\varphi_i(t) = \frac{\gamma_i}{\left(\gamma_i^k + \beta_i\right)^{1/k}}, \tag{13}$$

where $k \in \mathbb{R}^+$ is a tuning parameter, $\gamma_i : \mathbb{R}^2 \to \mathbb{R}^+$ is the goal function, and $\beta_i : \mathbb{R}^2 \to [0,1]$ is a constraint function for node $i$. The goal function $\gamma_i(t)$ in (13) encodes the control objective of positioning the sensor $i$ at $s_i$ and is designed as

$$\gamma_i(t) = \left\| x_i(t) - \bar{\zeta}(t) - s_i \right\|^2. \tag{14}$$

Similar to [15] and [16], the constraint function $\beta_i(t)$ in (13) is designed as

$$\beta_i(t) = \prod\nolimits_{j \in \mathcal{N}_i} b_{ij} \prod\nolimits_{j \in \mathcal{V}} B_{ij}, \tag{15}$$

to ensure collision avoidance and network connectivity by only accounting for nodes located within its communication area each time instant. In (15), $b_{ij} \triangleq b(x_i, x_j) : \mathbb{R}^2 \to [0,1]$ ensures connectivity of the network graph (i.e., guarantees that node $j \in \mathcal{N}_i$ will never leave the communication zone if node $i$ and $j$ are neighbors initially) and is designed as

$$b_{ij} = \begin{cases} 1 & d_{ij} \le R - \delta \\ -\frac{1}{\delta^2}(d_{ij} + 2\delta - R)^2 \\ +\frac{2}{\delta}(d_{ij} + 2\delta - R) & R - \delta < d_{ij} < R \\ 0 & d_{ij} \ge R. \end{cases} \tag{16}$$

Also in (15), $B_{ij} \triangleq B(x_i, x_j) : \mathbb{R}^2 \to [0,1]$ ensures that node $i$ is repulsed from any node $j$ to prevent a collision when they are close enough, and is designed as

$$B_{ik} = \begin{cases} -\frac{1}{\delta^2} d_{ik}^2 + \frac{2}{\delta} d_{ik} & d_{ik} < \delta \\ 1 & d_{ik} \ge \delta. \end{cases} \tag{17}$$

In (16) and (17), $R$ denotes the radius of the limited communication zone, and $\delta$ represents a buffer region, which indicates that a repulsive force will be generated to avoid collision for sensors $i$ and $k$ if $d_{ik} < \delta$, and an attractive force will be generated to enforce connectivity between sensors $i$ and $j$ if $R - \delta < d_{ij} < R$. Based on the designed potential function in (13), the distributed controller for each sensor is computed as

$$u_i = -K_i \nabla_{x_i} \varphi_i, \tag{18}$$

where $K_i$ is a positive gain, and $\nabla_{x_i}\varphi_i$ is the gradient of the $\varphi_i$ with respect to $x_i$.

**Theorem 2**: *Given an initially connected graph G composed of mobile sensors with kinematics given by (2), the controller developed in (18) ensures the graph remains connected and collision avoidance among sensors during the target tracking.*

*Proof:* Consider node $i$ located at a point $p_0$ that causes $\prod_{j \in \mathcal{N}_i} b_{ij} = 0$, which will be true when either only one neighboring node $j$ is about to disconnect from node $i$ or when multiple nodes are about to disconnect with node $i$ simultaneously. From (15), $\beta_i = 0$ , which indicates that the potential function designed in (13) achieves its maximum value. Since $\varphi_i$ is maximized at $q_0$, no open set of initial conditions can be attracted to $q_0$ under the negative gradient control law designed in (18). Therefore, the existing edge between node $i$ and node $j \in \mathcal{N}_i$ will be maintained for all time. If two sensors are about to collide with each other, $\beta_i$ will also achieve zero from (15) and lead to the maximum value of the potential function. Similar to the connectivity preservation, collision avoidance is guaranteed among sensors. ∎

**Theorem 3**: *Given an initially connected graph G, the controller developed in (18) ensures that the group of mobile sensors keep tracking the target at their desired $s_i$.*

*Proof:* Two nodes are guaranteed to achieve a desired relative position by using a potential function similar to (13), see [15] and [16]. To enable target tracking, the goal function $\gamma_i$ in (13) is modified from [15] and [16] to position each sensor $i$ at a desired $s_i$ with respect to the target. Note that $\gamma_i$ achieves its unique minimal of 0 when $x_i(t) - \bar{\zeta}(t) = s_i$, which implies that sensor $i$ is positioned at the desired relative position to the target. Since Assumption 3 guarantees that $\gamma_i$ and $\beta_i$ will not be zero simultaneously, the potential function designed in (13) achieves its minimum of 0 when $\gamma_i = 0$ and achieves its maximum of 1 when $\beta_i = 0$. Following a similar procedure in [15] and [16], it can be shown that the group of sensors will converge to $s_i$ driven by the negative gradient control law in (18). Due to space limitations, the proof detail is omitted here. ∎

## V. SIMULATION

A group of 5 mobile sensors are initially deployed in a workspace with the goal of tracking a moving target as shown in Fig. 1. Each sensor is assumed to have a limited communication zone of $R = 5\ m$ and $\delta = 1\ m$. In Fig. 1, the group of sensors form a connected graph, where each node represents a mobile sensor and the dashed line connecting two nodes indicates the available communication link. The target is represented by a square moving along a trajectory. The dash-dot lines connecting the target and sensors indicate where the sensor is able to sense the targets. It is assumed that only a portion of the sensors (i.e., node {2,3,4,5}) are able to sense the target and node 1 can not sense the target. The simulation results are presented in Fig. 2 and Fig. 3. Fig. 2 indicates convergence of the target tracking errors, which implies that every sensor dynamically updates its local estimate to track the time varying target states, even if some nodes (e.g., node 1 in this case) may not be able to sense the target. The navigation function based motion controller in (18) keeps positioning the sensors at $s_i$ with respect to the target. The trajectory evolution of the sensors and the target are shown in Fig. 3, where the triangle and '*' denotes the initial position of each sensor and

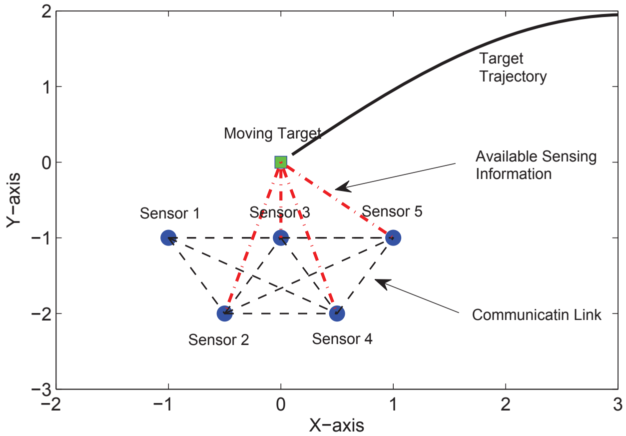


Fig. 1. A group of 5 nodes are deployed in the workspace with the goal of tracking the target. The target is represented by a square, and the sensors are denoted by dots. The sensors form a connected graph, where the dashed link indicates the available communication links. The dashed-dot line connecting the target and the sensor implies whether the sensor is able to measure the target position.

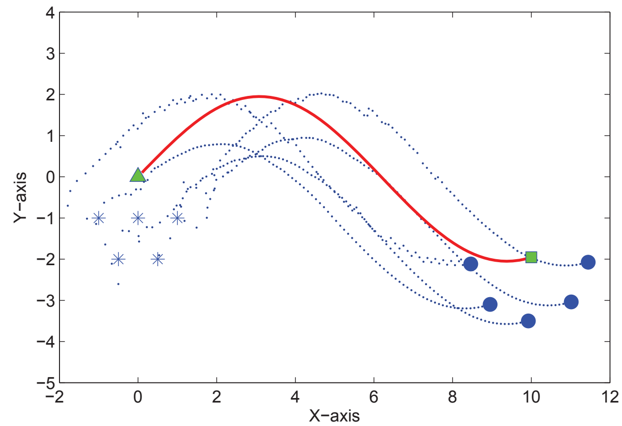


Fig. 3. The trajectory of the sensors and the target, where '*' and triangle denote the initial position of the sensors and the target respectively. The final position of the target and sensors are represented by the dots and square.

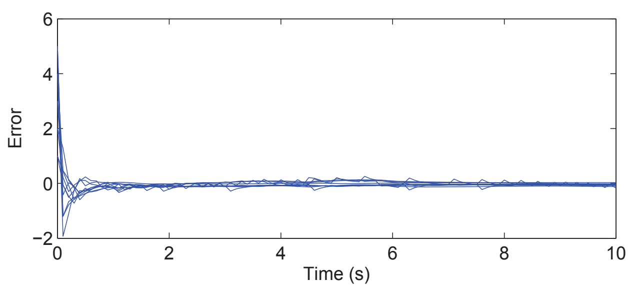


Fig. 2. The error plot of the local estimate $p_i$ and the target position.

the target respectively, and the square and dots denote their final position. The dotted line and solid line in Fig. 3 represent the generated trajectory of sensors and the target.

## VI. Conclusion

A novel dynamic average consensus algorithm is developed that enables each sensor to estimate the global mean of the measurements from all sensors by only communication with neighboring sensors. Based on the estimated global mean, a potential field based distributed motion control law is designed for each sensor to track the target cooperatively, while ensuring the network connectivity and collision avoidance among sensors. The developed approach achieves dynamic average consensus in a network where only a portion of the sensors observe the target, while actively preserving network connectivity.

## References

[1] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Trans. ASME J. Basic. Eng.*, vol. 82D, pp. 35–45, 1960.

[2] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, 2002.

[3] A. Doucet, N. De Freitas, and N. Gordon, *Sequential Monte Carlo methods in practice*. Springer Verlag, 2001.

[4] N. Gordon, D. Salmond, and A. Smith, "Novel approach to nonlinear/non-gaussian bayesian state estimation," in *IEE Proc. Radar and Signal Process.*, vol. 140, no. 2. IET, 1993, pp. 107–113.

[5] M. Coates, "Distributed particle filters for sensor networks," in *Proc. of Int. Symp. Information Processing in Sensor Networks*. ACM, 2004, pp. 99–107.

[6] X. Sheng, Y. Hu, and P. Ramanathan, "Distributed particle filter with gmm approximation for multiple targets localization and tracking in wireless sensor network," in *Proc. of Int. Symp. Information Processing in Sensor Networks*. IEEE Press, 2005, p. 24.

[7] R. Olfati-Saber and R. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1520–1533, Sept. 2004.

[8] W. Ren, R. W. Beard, and E. M. Atkins, "Information consensus in multivehicle cooperative control," *IEEE Contr. Syst. Mag.*, vol. 27, pp. 71–82, April 2007.

[9] D. Spanos, R. Olfati-Saber, and R. Murray, "Dynamic consensus on mobile networks," in *IFAC world congress*. Prague Czech Republic, 2005.

[10] R. Freeman, P. Yang, and K. Lynch, "Stability and convergence properties of dynamic average consensus estimators," in *Proc. IEEE Conf. Decis. Control*. IEEE, 2006, pp. 338–343.

[11] W. Ren, "Multi-vehicle consensus with a time-varying reference state," *Systems & Control Letters*, vol. 56, no. 7-8, pp. 474–483, 2007.

[12] R. Olfati-Saber and J. Shamma, "Consensus filters for sensor networks and distributed sensor fusion," in *Proc. IEEE Conf. Decis. Control and Eur. Control Conf.*, 2005, pp. 6698–6703.

[13] M. Zhu and S. Martínez, "Discrete-time dynamic average consensus," *Automatica*, vol. 46, no. 2, pp. 322–329, 2010.

[14] Y. Mostofi, T. Chung, R. Murray, and J. Burdick, "Communication and sensing trade-offs in decentralized mobile sensor networks: a cross-layer design approach," in *Proc. of Int. Symp. Inf. process. in Sensor Networks*. IEEE Press, 2005, p. 16.

[15] Z. Kan, A. Dani, J. M. Shea, and W. E. Dixon, "Network connectivity preserving formation stabilization and obstacle avoidance via a decentralized controller," *IEEE Trans. Automat. Control*, vol. 57, no. 7, pp. 1827– 1832, 2012.

[16] ——, "Ensuring network connectivity during formation control using a decentralized navigation function," in *Proc. IEEE Mil. Commun. Conf.*, San Jose, CA, 2010, pp. 954–959.