

# Consensus Tracking of an Underwater Vehicle Using Weighted Harmonic Mean Density

Ved Prakash Dubey and Shovan Bhaumik

**Abstract**—This paper addresses an underwater target tracking problem in which a large number of sonobuoy sensors are deployed on a surveillance region. The region is divided into several sub-regions, where a single tracker, capable of generating track is installed. Each sonobuoy can measure the direction of arrival of acoustic signals (known as bearing angles) and communicate the measurements with the local tracker. Further, each local tracker can communicate with all other trackers, where each of them can exchange their estimate and finally a consensus is reached. We propose a weighted harmonic mean density (HMD) based tracking to reach a consensus and provide a solution for the fusion of Gaussian densities. In this approach, optimal weights are assigned by minimizing the Kullback-Leibler divergence measure. Performance of the proposed method is measured using root mean square error, percentage of track divergence, and normalized estimation error squared. Simulation results demonstrate that the optimized HMD-based fusion outperforms existing fusion methods during a distributed tracking.

**Index Terms**—Harmonic density fusion, distributed tracking, consensus filtering, Kullback-Leibler divergence, underwater target tracking.

## I. INTRODUCTION

In the last few decades, there have been substantial advances in communication and sensor technology, leading to the increased adoption of multi-sensor systems in the field of underwater tracking [1], [2], [3], [4]. In a single-sensor system, the received information becomes highly corrupted or lost for a long distance sensing [5], [6]. Therefore, sensors should be placed moderately close to the signal source to obtain more precise information. To address this issue, a multi-sensor tracking system was developed for underwater surveillance and tracking applications [7], [8], [9], [10].

The tracking process using multiple sensors can be categorized into three groups: centralized, decentralized, and distributed tracking [11], [12]. Among these, decentralized and distributed tracking are popular for their robustness in handling sensor failures and accommodating addition or removal of sensors and trackers [13], [11]. Distributed tracking over a sensor network involves two types of nodes: sensor nodes and processing nodes. Sensor nodes possess only sensing and communication capabilities, allowing them to sense a physical quantity from the environment and also be able to send their measured information to the processing nodes. On the other hand, processing nodes are equipped with both computational

and communication capabilities. They can process the information received from sensors to estimate the system states and subsequently share the estimated information with neighboring processing nodes [11], [14]. In tracking applications, the processing nodes are also called trackers.

Using prior knowledge of the target, each tracker identifies sensors located closer to the target that have more accurate and informative measurements [15]. The selected sensor then generates measurement samples of the suspected target and forwards them to their respective trackers. Each tracker utilizes only the measured information of its local sensors to determine the state of the target. However, it is well known that sensors located closer to the target provide measurements with greater variation in their bearing information. Consequently, only the tracker situated in the sub-region containing the target provides an accurate estimate. Therefore, it is necessary to fuse the estimated information across all trackers to achieve a consensus, ensuring that the state estimates of all trackers are accurate and consistent.

To reach a consensus [12] among the trackers, they share the probability density function (pdf) (also called tracks) with their neighboring trackers. Track fusion takes place at each tracker to reach consensus between their track densities. In the context of information shared between the tracking nodes, the consensus filters are broadly classified into three categories: consensus on the estimate (CE), consensus on measurement (CM), and consensus on information (CI) [16], [12]. These filters are restricted to the Bayesian framework and have limited effectiveness. To overcome this limitation, alternative consensus strategies are employed to merge density information from multiple trackers. These techniques are commonly known as track-to-track fusion, as they consolidate the probability density functions [17].

To reach a consensus, computation of mean plays a pivotal role. Three classical approaches for computing the mean of a data set are the arithmetic, geometric, and harmonic means. Among these, the arithmetic and geometric means together with their weighted generalizations have been extensively studied in the context of track density fusion and data association [18]. Among them, arithmetic density fusion is known to overestimate uncertainty, thereby degrading the estimation accuracy [19]. On the other hand, geometric mean density fusion does not admit a closed-form solution because of the non-integral exponent of the density function [20], [21]. To address this, weighted inverse covariance fusion, also known as Chernoff fusion, is proposed in the literature as an approximation of weighted geometric fusion [11], [22]. The harmonic mean density is free from both of these limitations.

In [23], a weighted harmonic mean fusion method was proposed to combine two density functions. To fuse multiple densities, a sequential fusion strategy was suggested in [23]. This approach is computationally not efficient and requires successive pairwise fusions with an optimization step at each stage to determine the optimal weight. To address this limitation, here we introduce a method to compute the harmonic mean density among multiple pdfs in a single step. The weights of the densities are found by minimizing Kullback-Leibler divergence measure. As in the proposed framework, the weights corresponding to all densities are obtained through a single optimization step, thereby yielding a more efficient and scalable solution for multi-track fusion.

The effectiveness of the proposed fusion method is evaluated by comparing its tracking performance with other fusion techniques in a distributed tracking environment. The comparison is conducted in terms of root mean square error (RMSE) and percentage of track divergence, while consistency is verified using the average normalized estimation error squared (ANEES). In simulation, the proposed harmonic mean density fusion demonstrates robust performance when deterministic sample point filters and the shifted Rayleigh filter are adopted to generate tracks.

## II. TRACKING PROBLEM IN A SURVEILLANCE REGION

As discussed earlier, we consider a scenario in which a large number of battery-powered sonobuoys are airdropped and uniformly distributed across the sea surface within the surveillance region. Each sensor is capable of measuring the direction of arrival of an acoustic signal emitted by an underwater target moving within the region. Unlike centralized approaches (e.g., [15]), we adopt a distributed architecture in which multiple trackers are uniformly deployed, thereby partitioning the surveillance region into several sub-regions. All sensors are synchronized with their nearest tracker and transmit their measurements to it. Each tracker then estimates the target state locally using the measurements from its associated sensors.

This distributed framework is inherently more robust, as the failure of a communication channel or an individual tracker has minimal impact on the overall tracking performance. Moreover, it offers reduced processing time compared to centralized tracking. An example of the distributed tracking architecture is illustrated in Fig. 1. In this figure, we can see the distribution of sensors and trackers in arbitrary surveillance area. The trackers are interconnected via wireless communication channels, allowing them to exchange their estimated information. The key assumptions underlying the considered scenario are summarized as follows.

- The sensors are stationary (unaffected by sea currents) and they are uniformly distributed throughout the surveillance region.
- The total area is subdivided in several regions, and a tracker is placed in each subregion where each sensor is linked to its nearest tracker.
- To prolong battery life, only a few sensors are activated during a specific interval to transmit their measurements.

- All sensors are identical, passive devices capable of measuring the bearing angle of an acoustic source.
- Each tracker performs local tracking using the available measurements and it generates a track.
- Each tracker communicates with its neighboring trackers, sharing its estimated track. Through track fusion, each node produces a consensus tracking result.

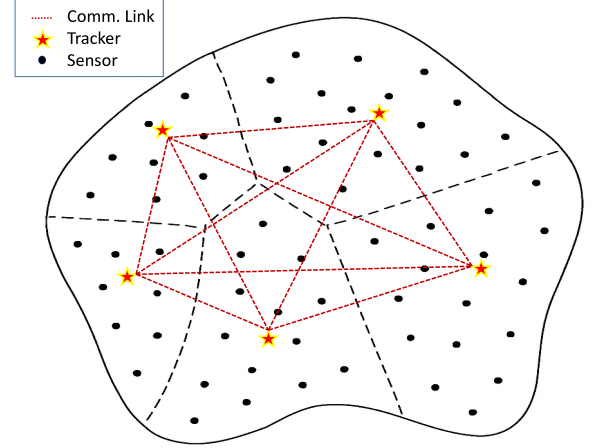


Fig. 1: Schematic representation of a distributed tracking architecture where the sensors are deployed uniformly. Here, the whole area is divided in several sub-regions, and trackers are placed inside each sub-region and they are connected with each other through a communication link.

### A. Process model

*Scenario 1: constant velocity model* : In this scenario, the target follows a straight-line motion and it is modeled as a nearly constant velocity [24]. The process dynamics is defined by

$$\mathcal{X}_k = F\mathcal{X}_{k-1} + \eta_k, \quad (1)$$

where  $\mathcal{X}_k, \eta_k \in \mathbb{R}^{n_x}$ ,  $\mathcal{X}_k = [x_k \ y_k \ \dot{x}_k \ \dot{y}_k]^\top$  is a target state vector.  $x_k, y_k$  are positions along  $X$  and  $Y$  axis, respectively at time step  $k$ .  $F$  is the system matrix which can be expressed as  $F = \begin{bmatrix} I_2 & TI_2 \\ 0_2 & I_2 \end{bmatrix}$ , where  $I_2$  is the  $2 \times 2$  identity matrix,  $0_2$  the  $2 \times 2$  zero matrix, and  $T$  is the sampling interval.  $\eta_k$  is the process noise which is assumed to be white, Gaussian with zero mean and covariance  $Q_k$  i.e.  $\eta_k \sim \mathcal{N}(0, Q_k)$ . The expression of  $Q_k$  is

$$Q_k = q_1 \begin{bmatrix} \frac{T^3}{3} I_2 & \frac{T^2}{2} I_2 \\ \frac{T^2}{2} I_2 & TI_2 \end{bmatrix}, \quad (2)$$

where  $q_1$  is the intensity of process noise.

*Scenario 2: constant turn rate model*: In this scenario, the target motion is along a coordinated turn (CT) and is modeled in the  $X - Y$  plane with a nearly constant turn rate [16]. The process dynamics are defined by

$$\mathcal{X}_k = F_{k-1}\mathcal{X}_{k-1} + \eta_k, \quad (3)$$

where  $\mathcal{X}_k = [x_k \ y_k \ \dot{x}_k \ \dot{y}_k \ \Omega_k]^\top$ ,  $\Omega_k$  is the angular turn rate of the target, positive for counterclockwise rotation and negative for clockwise rotation, and the expression of  $F_k$  is given by

$$F_k = \begin{bmatrix} 1 & 0 & \frac{\sin \Omega_k T}{\Omega_k} & -\frac{1-\cos \Omega_k T}{\Omega_k} & 0 \\ 0 & 1 & \frac{1-\cos \Omega_k T}{\Omega_k} & \frac{\sin \Omega_k T}{\Omega_k} & 0 \\ 0 & 0 & \cos \Omega_k T & -\sin \Omega_k T & 0 \\ 0 & 0 & \sin \Omega_k T & \cos \Omega_k T & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (4)$$

The process noise covariance matrix  $Q_k$  corresponding to this system is given by  $Q_k = \text{diag}([q_1 Q_{1,k}, q_2 T])$ , where

$$Q_{1,k} = \begin{bmatrix} \frac{2s_{\Omega_k}}{\Omega_k^3} & 0 & \frac{c_{\Omega_k}}{\Omega_k^2} & -\frac{s_{\Omega_k}}{\Omega_k^2} \\ 0 & \frac{2s_{\Omega_k}}{\Omega_k^3} & \frac{s_{\Omega_k}}{\Omega_k^2} & \frac{c_{\Omega_k}}{\Omega_k^2} \\ \frac{c_{\Omega_k}}{\Omega_k^2} & \frac{s_{\Omega_k}}{\Omega_k^2} & T & 0 \\ -\frac{s_{\Omega_k}}{\Omega_k^2} & \frac{c_{\Omega_k}}{\Omega_k^2} & 0 & T \end{bmatrix}, \quad (5)$$

where  $s_{\Omega_k} = \Omega_k T - \sin \Omega_k T$ ,  $c_{\Omega_k} = 1 - \cos \Omega_k T$  and  $q_2$  is the process noise intensity corresponding to its turn rate.

### B. Measurement model

The sensors are deployed in the surveillance area and they are capable to measure the bearing angle of the target at their location. The mathematical expression representing the measurement acquired by the  $i$ th sensor in the  $j$ th sub-region at the  $k$ th instant is denoted by

$$Z_{i,k}^j = \theta_{i,k}^j + v_{i,k}^j, \quad (6)$$

where

$$\theta_{i,k}^j = h(\mathcal{X}_k, \mathcal{X}_i^{s,j}) = \tan^{-1} \left( \frac{x_k - x_i^{s,j}}{y_k - y_i^{s,j}} \right), \quad (7)$$

where  $(x_i^{s,j}, y_i^{s,j})$  is the  $i$ th sensor coordinate in  $j$ th subregion,  $\theta_{i,k}$  is the bearing angle of the sensor,  $\tan^{-1}(\cdot)$  denotes the four-quadrant inverse tangent function, and the angles  $\{Z_{i,k}^j, \theta_{i,k}^j, v_{i,k}^j\} \in (-\pi, \pi]$ . The term  $v_{i,k}$  represents the measurement noise, assumed to be white, Gaussian with zero mean and covariance  $\mathbb{E}[v_{i,k}^2] = R_{i,k}$ .

At a particular time instant, a local tracker selects a subset of sensors (at least two) from the available set and receives their measurements. The selection of sensors for transmitting their measurements is carried out by minimizing the trace of the Fisher information matrix (FIM). For a detailed discussion on the optimization procedure, readers are referred to [25]. Thus, at any instant  $k$ , the measurements received by the  $j$ th local tracker are given by

$$\mathcal{Y}_k^j = \{Z_{i,k}^j \mid \mathcal{S}_{i,k}^j = 1\}, \quad (8)$$

where  $\mathcal{S}_{i,k}^j \in \{0, 1\}$  denotes the status of the  $i$ th sensor for the  $j$ th tracker at time  $k$ . Specifically,  $\mathcal{S}_{i,k}^j = 1$  if the  $j$ th tracker selects the measurement from the  $i$ th sensor, and  $\mathcal{S}_{i,k}^j = 0$  otherwise.

In the described scenario, when a target is suspected within the surveillance area, all distributed trackers activate a subset of sensors based on the detected target location. Each tracker

then estimates the target's track using the measurements from its activated sensors. These tracks are generated locally, and their estimation accuracy depends on the proximity of the target to the activated sensors; hence, trackers closer to the target provide better estimates than those farther away. Each tracker shares its estimated track density with its neighboring trackers, enabling them to fuse the densities and reach a consensus estimate.

The objective is to find a consensus density function so that we can get accurate estimation results with minimal error. Our approach employs optimal weighted mean harmonic density fusion, where the weights are determined by formulating a cost function based on the symmetrized Kullback–Leibler divergence measure.

## III. HARMONIC MEAN DENSITY FUSION

As stated above, the local tracker estimates the target state using the measurements of the sensors located in the corresponding subregion. Let us represent the estimated pdf of  $j$ th tracker as  $p_j(\mathcal{X}_k | \mathcal{Y}_k^j)$  or briefly by  $p_j(\mathcal{X}_k)$ . Our objective is to fuse all such pdfs and obtain an overall consensus pdf of the target states. The generalized weighted mean density  $p_\omega(\mathcal{X}_k)$  for mixing pdfs obtained from  $N$  number of trackers viz.  $p_1(\mathcal{X}_k), \dots, p_N(\mathcal{X}_k)$  can be represented as [22]

$$p_\omega(\mathcal{X}_k) = \frac{\mathcal{M}_\omega(p_1(\mathcal{X}_k), \dots, p_N(\mathcal{X}_k))}{\zeta_{\mathcal{M}_\omega}(p_1(\mathcal{X}_k), \dots, p_N(\mathcal{X}_k))}, \quad (9)$$

where  $\mathcal{M}_\omega(\cdot)$  is the weighted mean function,  $\omega$  is the weight vector, and the normalization constant  $\zeta_{\mathcal{M}_\omega}(\cdot)$  ensures fusion density to a valid probability density function. The fused density can be obtained using arithmetic mean, geometric mean, or harmonic mean formulations. In this section, we derive the harmonic mean density (HMD) function for the general case of combining an arbitrary number of densities. For the fusion process, we adopt a one-step batch fusion of multiple pdfs, which significantly reduces computation time for both weight optimization and density fusion compared to the recursive procedure suggested in [23].

### A. Arithmetic and geometric mean density

The solution of the mixing probabilities in Eqn. (9) for arithmetic mean density (AMD) fusion is represented as a convex combination of probability densities [19]. In the case of geometric mean density (GMD) fusion, the fused density is obtained as an exponential mixture of the individual densities [20]. These fused densities can be expressed as:

$$\begin{aligned} \mathcal{M}_\omega^a(\mathcal{X}_k) &= \omega_1 p_1(\mathcal{X}_k) + \dots + \omega_N p_N(\mathcal{X}_k), \\ &= \sum_{j=1}^N \omega_j p_j(\mathcal{X}_k), \end{aligned} \quad (10)$$

$$\begin{aligned} \mathcal{M}_\omega^g(\mathcal{X}_k) &= p_1^{\omega_1}(\mathcal{X}_k) \times \dots \times p_N^{\omega_N}(\mathcal{X}_k), \\ &= \prod_{j=1}^N p_j^{\omega_j}(\mathcal{X}_k), \end{aligned} \quad (11)$$

where weights are  $\{\omega_1, \dots, \omega_N\} \in [0, 1]$ , and  $\sum_{j=1}^N \omega_j = 1$ .

The weight represents the contribution of each tracker's density in the fused density.

### B. Harmonic mean density function

Previous studies [19], [20], [21] have explored the use of AMD and GMD functions for probabilistic fusion. However, both approaches exhibit notable limitations. AMD fusion tends to overestimate uncertainty, which can compromise the consistency and informativeness of the resulting estimates. GMD fusion, while theoretically appealing, lacks a closed-form solution due to the non-integral exponent in its density formulation. To address this, Chernoff fusion-also known as exponential mixture density fusion, is often employed as an approximation to GMD, typically by matching the first two moments with a Gaussian surrogate [26]. On the other hand, harmonic mean density (HMD) fusion avoids these problems, offering a closed-form, consistent, and reliable framework for producing a consensus probability density function.

The mixture of weighted HMD fusion mixing probabilities provided in Eqn. (9) is given by [23] and is defined by

$$\begin{aligned} \mathcal{M}_\omega^h(\mathcal{X}_k) &= \left( \frac{\omega_j}{p_j(\mathcal{X}_k)} + \dots + \frac{\omega_j}{p_j(\mathcal{X}_k)} \right)^{-1}, \\ &= \left( \sum_{j=1}^N \frac{\omega_j}{p_j(\mathcal{X}_k)} \right)^{-1}. \end{aligned} \quad (12)$$

The harmonic pdf is proportional to its mixing probability and simplified expression is represented by

$$p_\omega^h(\mathcal{X}_k) \propto \mathcal{M}_\omega^h(\mathcal{X}_k) = \frac{\prod_{j=1}^N p_j(\mathcal{X}_k)}{\sum_{j=1}^N \omega_j \prod_{i=1, i \neq j}^N p_i(\mathcal{X}_k)}, \quad (13)$$

where  $\omega_j$  is the weight associated with the  $j$ -th component of the distribution. Let  $\{p_j(\mathcal{X}_k)\}_{j=1}^N$  denote  $N$  multivariate Gaussian densities, where  $p_j(\mathcal{X}_k) = \mathcal{N}(\hat{\mathcal{X}}_{k|k}^j, P_{k|k}^j)$ . Substituting into (13), the weighted HMD mixture can be expressed as

$$\begin{aligned} p_\omega^h(\mathcal{X}_k) &\propto \frac{\prod_{j=1}^N \mathcal{N}(\hat{\mathcal{X}}_{k|k}^j, P_{k|k}^j)}{\sum_{j=1}^N \omega_j \prod_{i=1, i \neq j}^N \mathcal{N}(\hat{\mathcal{X}}_{k|k}^i, P_{k|k}^i)} \\ &= \frac{\prod_{j=1}^N \mathcal{N}(\hat{\mathcal{X}}_{k|k}^j, P_{k|k}^j)}{\sum_{j=1}^N \omega_j \mathcal{N}(\hat{\mathcal{X}}_{k|k}^{c,j}, P_{k|k}^{c,j})}, \end{aligned} \quad (14)$$

where each term in the denominator consists of the weight  $\omega_j$  multiplied by the product of  $N - 1$  Gaussian densities. If we assume that each tracker approximates the estimated density function with a Gaussian distribution, then

their multiplication will also remain Gaussian. So, we write  $\prod_{i=1, i \neq j}^N \mathcal{N}(\hat{\mathcal{X}}_{k|k}^i, P_{k|k}^i) = \mathcal{N}(\hat{\mathcal{X}}_{k|k}^{c,j}, P_{k|k}^{c,j})$ , where  $c$  is used to denote the combined Gaussian parameters. The mean and covariance of the combined Gaussian are given by [27]

$$P_{k|k}^{c,j} = \left[ \sum_{i=1, i \neq j}^N P_{k|k}^i \right]^{-1}, \quad (15)$$

and

$$\hat{\mathcal{X}}_{k|k}^{c,j} = P_{k|k}^{c,j} \left[ \sum_{i=1, i \neq j}^N P_{k|k}^i \hat{\mathcal{X}}_{k|k}^i \right]. \quad (16)$$

The denominator of (13) is a weighted sum of Gaussian densities, which is non-Gaussian in general. It can, however, be approximated by a single Gaussian using moment matching, similar to the weighted AMD fusion [19]. Specifically,

$$\sum_{j=1}^N \omega_j \mathcal{N}(\hat{\mathcal{X}}_{k|k}^{c,j}, P_{k|k}^{c,j}) \approx \mathcal{N}(\hat{\mathcal{X}}_{k|k}^{eq}, P_{k|k}^{eq}), \quad (17)$$

where  $\hat{\mathcal{X}}_{k|k}^{eq}$  and  $P_{k|k}^{eq}$  denote the mean and covariance of the approximated Gaussian, and are given by

$$\hat{\mathcal{X}}_{k|k}^{eq} = \sum_{j=1}^N \omega_j \hat{\mathcal{X}}_{k|k}^{c,j}, \quad (18)$$

and

$$P_{k|k}^{eq} = \sum_{j=1}^N \omega_j [P_{k|k}^{c,j} + (\hat{\mathcal{X}}_{k|k}^{eq} - \hat{\mathcal{X}}_{k|k}^{c,j})(\hat{\mathcal{X}}_{k|k}^{eq} - \hat{\mathcal{X}}_{k|k}^{c,j})^\top]. \quad (19)$$

Now, from (13) using the Gaussian approximation, the fused density becomes

$$p_\omega^h(\mathcal{X}_k) \approx \frac{\prod_{j=1}^N \mathcal{N}(\hat{\mathcal{X}}_{k|k}^j, P_{k|k}^j)}{\mathcal{N}(\hat{\mathcal{X}}_{k|k}^{eq}, P_{k|k}^{eq})} = \mathcal{N}(\hat{\mathcal{X}}_{k|k}^f, P_{k|k}^f), \quad (20)$$

where the fused mean and covariance are obtained as [27]

$$P_{k|k}^f = \left[ \sum_{j=1}^N P_{k|k}^j \right]^{-1}, \quad (21)$$

$$\hat{\mathcal{X}}_{k|k}^f = P_{k|k}^f \left[ \sum_{j=1}^N P_{k|k}^j \hat{\mathcal{X}}_{k|k}^j - P_{k|k}^{eq} \hat{\mathcal{X}}_{k|k}^{eq} \right]. \quad (22)$$

### C. Weight optimization

In density fusion, it is crucial to assign suitable weights to the individual distributions to obtain a probability density function of states. These weights are typically derived by formulating a cost function, where a symmetrical divergence measure serves as the performance criterion for evaluating distribution [22]. An objective function for determining the weights of individual densities is constructed to minimize the dispersion of the divergences between the fused density  $p_\omega(\mathcal{X}_k)$  and the individual densities  $\{p_i(\mathcal{X}_k)\}_{i=1}^N$ . We formu-

late an optimization problem as described below.

$$\omega_k^* = \arg \min_{\omega} \sum_{i=1}^N \left( \mathcal{D}(p_{\omega}(\mathcal{X}_k) : p_i(\mathcal{X}_k)) - \overline{\mathcal{D}} \right)^2, \quad (23)$$

subject to

$$0 \leq \omega_i \leq 1, \quad (24)$$

and

$$\sum_{i=1}^N \omega_i = 1, \quad (25)$$

where  $\mathcal{D}$  denotes a symmetric divergence measure and

$$\overline{\mathcal{D}} = \frac{1}{N} \sum_{i=1}^N \mathcal{D}(p_{\omega}(\mathcal{X}_k) : p_i(\mathcal{X}_k)). \quad (26)$$

In this work we have consider Kullback–Leibler (KL) divergence as a measure of divergence. Since the KL divergence is inherently asymmetric, it must be symmetrized for use in this context. The symmetrized KL divergence between two probability densities  $p_i(\mathcal{X}_k)$  and  $p_j(\mathcal{X}_k)$  is given by

$$\mathcal{D}(p_i(\mathcal{X}_k) : p_j(\mathcal{X}_k)) = \frac{1}{2} [\mathcal{D}_{\text{KL}}(p_i(\mathcal{X}_k) : p_j(\mathcal{X}_k)) + \mathcal{D}_{\text{KL}}(p_j(\mathcal{X}_k) : p_i(\mathcal{X}_k))], \quad (27)$$

where the KL divergence between them is defined as [28]:

$$\mathcal{D}_{\text{KL}}(p_i(\mathcal{X}_k) : p_j(\mathcal{X}_k)) = \int p_i(\mathcal{X}_k) \ln \frac{p_i(\mathcal{X}_k)}{p_j(\mathcal{X}_k)} d\mathcal{X}_k. \quad (28)$$

In the case of  $p_i(\mathcal{X}_k) \sim \mathcal{N}(\hat{\mathcal{X}}_{k|k}^i, P_{k|k}^i)$  and  $p_j(\mathcal{X}_k) \sim \mathcal{N}(\hat{\mathcal{X}}_{k|k}^j, P_{k|k}^j)$ , the KL divergence is expressed by

$$\begin{aligned} \mathcal{D}_{\text{KL}}(\cdot) = & \frac{1}{2} \left[ \text{tr}(P_{k|k}^j)^{-1} P_{k|k}^i - n_{\mathcal{X}} + \ln \frac{|P_{k|k}^j|}{|P_{k|k}^i|} \right] \\ & + \frac{1}{2} (\hat{\mathcal{X}}_{k|k}^i - \hat{\mathcal{X}}_{k|k}^j)^{\top} P_{k|k}^j^{-1} (\hat{\mathcal{X}}_{k|k}^i - \hat{\mathcal{X}}_{k|k}^j), \end{aligned} \quad (29)$$

where  $n_{\mathcal{X}}$  is the dimensionality of the density function,  $|\cdot|$  denotes the matrix determinant,  $\text{tr}(\cdot)$  denotes the trace of a matrix. A derivation of the above expression is provided in Appendix A.

#### IV. SIMULATION RESULTS

To simulate the designed problem, we considered a total of 100 sonobuoys distributed over a  $(10 \times 10) \text{ km}^2$  surveillance region. A uniform distribution is adopted to realize the sensor placement. The entire area was divided into four  $(5 \times 5) \text{ km}^2$  square sub-regions, and four trackers ( $N = 4$ ) were deployed, one assigned to each sub-region. All trackers were positioned near the center of their respective sub-regions, ensuring that all sensors could transmit their measurements while minimizing energy expenditure throughout the tracking interval.

Initially, all sensors were assumed to be in passive mode. Whenever a target was suspected within the surveillance region, each tracker activates two sensors from its sub-region, and allowing them to transmit their measurements. These sensors are strategically chosen by minimizing a cost function derived from their Fisher information matrix, as outlined in [29], [15]. Using the bearing angle of these chosen sensors, the

designed tracking system becomes observable, and the tracker could triangulate the target position within its estimation algorithm.

Two tracking scenarios are considered: a non-maneuvering target modeled as nearly constant velocity (CV) and a maneuvering target modeled as nearly constant turn (CT). At any given time, only a single target is assumed to be present for tracking. Both the scenarios as described in the section II-A are simulated for 36 minutes with a sampling interval of  $T = 0.25 \text{ min}$ , and the remaining simulation parameters are listed in Table I.

The tracking scenarios are illustrated in Fig. 2, where black dots represent sensors' location, and red triangles denote tracker positions. The surveillance region is divided into sub-regions (outlined by dotted lines), each labeled accordingly. The true target and its estimated initial position are marked with red stars, while their respective trajectories are shown by dotted red and blue lines. Furthermore, all trackers were assumed to mutually interconnected, sharing their tracking information with one another. In this case, fusion was performed at each node, and a single-step fusion was sufficient to achieve consensus, resulting in identical track densities across all trackers after fusion.

TABLE I: Tracking parameters

| Parameters                               | Scenario 1                     | Scenario 2                        |
|------------------------------------------|--------------------------------|-----------------------------------|
| No. of operational sensors ( $n_s$ )     | 100                            | 100                               |
| No. of selected sensors ( $n$ )          | 2                              | 2                                 |
| Sampling interval ( $T$ )                | 0.25 min                       | 0.25 min                          |
| Sensor selection interval                | 2 min                          | 2 min                             |
| Process noise intensities ( $q_1, q_2$ ) | $1.944 \text{ m}^2/\text{min}$ | $0.01 \text{ rad}^2/\text{min}^3$ |
| Measurement noise ( $R_{i,k}$ )          | $(2^\circ)^2$                  | $(2^\circ)^2$                     |
| Initial target location                  | (9, 9) km                      | (8.5, 8) km                       |
| Target speed ( $v^t$ )                   | 10 knots                       | 10 knots                          |
| Target initial course angle              | $-130^\circ$                   | $-165^\circ$                      |
| Target turn rate ( $\Omega$ )            | –                              | $-1.84^\circ/\text{min}$          |

##### A. Initialization

Prior to estimation, all tracks are initialized using the first two measurements from two selected sensors, following the method described in [29]. The target location and corresponding uncertainty can be obtained by triangulating the bearing measurements from any two sensors. Let us consider the two sensors, positioned at  $(x_1^s, y_1^s)$  and  $(x_2^s, y_2^s)$ , and their respective bearing measurements at time  $k$  are  $Z_{1,k}$  and  $Z_{2,k}$ . The intersection of bearing lines gives the target position and is obtained by

$$\begin{aligned} x_k &= x_1^s + \lambda \sin(Z_{1,k}), \\ y_k &= y_1^s + \lambda \cos(Z_{1,k}), \end{aligned} \quad (30)$$

where  $\lambda = \frac{(y_2^s - y_1^s) \sin(Z_{2,k}) - (x_2^s - x_1^s) \cos(Z_{2,k})}{\sin(Z_{2,k} - Z_{1,k})}$ . Here,  $Z_{2,k} \neq Z_{1,k}$ , and the difference between the two measurements is sufficiently large to enable reliable and more accurate triangulation. Using two consecutive triangulated position estimates, the target state is initialized by

$$\hat{\mathcal{X}}_{0|0} = \left[ x_0, y_0, \frac{x_1 - x_0}{T}, \frac{y_1 - y_0}{T} \right]^\top, \quad (31)$$

where  $(x_0, y_0)$  and  $(x_1, y_1)$  are successive triangulated positions. The corresponding initial covariance is

$$P_{0|0} = \text{diag}([P_{xy}, \frac{v_m^2}{3} I_2]), \quad (32)$$

with  $P_{xy}$  denoting the position uncertainty obtained by first-order propagation of bearing measurement variance and determined by

$$P_{xy} = J R J^\top, \quad (33)$$

where  $J = \nabla_{\mathcal{Y}_0}(H \hat{\mathcal{X}}_{0|0})$  denotes the Jacobian of  $H \hat{\mathcal{X}}_{0|0}$  with respect to the measurement vector  $\mathcal{Y}_0$ ,  $H = [I_2 \ 0_2]$  is the transformation matrix,  $\mathcal{Y}_0 = [Z_{1,0}, Z_{2,0}]^\top$  is the combined sensor measurement, and  $R = \text{diag}([R_{1,0}, R_{2,0}])$  is the corresponding covariance. Assuming the target speed along each axis is uniformly distributed in  $[-v_m, v_m]$ , the variance of each velocity component is  $\frac{v_m^2}{3}$ , defining the velocity block of  $P_{0|0}$ .

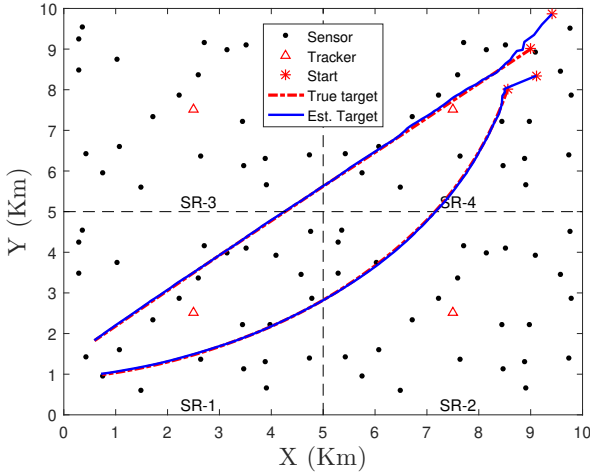


Fig. 2: Illustration of the tracking scenarios with sensor locations, true and estimated target trajectory, when the target moves with nearly constant velocity and with a constant turn rate.

### B. Scenario 1: Target moving with nearly constant velocity

In this scenario, we have considered a straight line motion of target, where it is modeled as nearly constant-velocity motion. At each instant of time all the trackers produces its own track by processing the the available measurements. These generates tracks shared to all of their neighboring trackers for fusion. During the fusion, a cost function is minimized using nonlinear programming and the corresponding optimal weights for each tracker's pdf are shown in Fig. 3. From the figure, it could be

seen the consensus fusion assigns higher weights to trackers that are closer to the target, initially favoring tracker-4 and eventually tracker-1, with the weights updating dynamically as the target moves.

The RMSE results for both position and velocity are compared for all four trackers without fusion, alongside their wighted harmonic mean density fusion using two selected sensors and the SRF, as shown in Figs. 5 and 6. The results indicate that track fusion consistently follows the most accurate trajectory among the individual trackers throughout the entire tracking interval and achieves faster convergence.

The RMSE results of both position and velocity are compared for consensus tracking by assigning equal weights and optimized weights for each tracker, where the optimized weights are obtained from the cost function using SRF, as shown in Figs. 7 and 8. The results demonstrate that tracking performance with optimized weights is superior to that with equal weights, indicating that weight optimization is essential for achieving accurate consensus estimates. Furthermore, among the fusion methods with optimized weights, HMD and GMD exhibit similar performance.

The RMSE results of both position and velocity are compared for weighted harmonic density fusion based consensus tracking using different DSPFs [30] and SRF, as shown in Figs. 9 and 10. From the figures, it can be seen that all the Gaussian filters designed for tracking perform well in consensus tracking, exhibiting almost identical performance.

The consistency of the proposed fusion is verified by the averaged normalized estimation error squared (ANEES), calculated using the relation:

$$\text{ANEES}_k = \frac{1}{M_c} \sum_{m=1}^{M_c} (\mathcal{X}_k^m - \hat{\mathcal{X}}_{k|k}^m)^\top P_{k|k}^{m-1} (\mathcal{X}_k^m - \hat{\mathcal{X}}_{k|k}^m),$$

where  $M_c$  is the number of independent Monte Carlo simulations. The designed filter is considered consistent if, over time, the ANEES value lies within the 95% confidence bounds  $[l_b, u_b]$ . These bounds for  $n_{\mathcal{X}}$  states are defined as:

$$l_b = n_{\mathcal{X}} \left[ \left( 1 - \frac{2}{9n_{\mathcal{X}}M_c} \right) - 1.96 \sqrt{\frac{2}{9n_{\mathcal{X}}M_c}} \right]^3, \quad (34)$$

$$u_b = n_{\mathcal{X}} \left[ \left( 1 - \frac{2}{9n_{\mathcal{X}}M_c} \right) + 1.96 \sqrt{\frac{2}{9n_{\mathcal{X}}M_c}} \right]^3. \quad (35)$$

The ANEES values of all four tracks without fusion are compared against those obtained using harmonic fusion tracking, as illustrated in Fig. 11. The figure demonstrates that each individual sub-region tracker maintains consistent performance, with Tracker-4 reaching the 95% confidence bounds earlier than the others. On the other hand, the consensus tracking achieved through harmonic fusion also converges to the confidence bounds and remains within the 95% confidence region throughout the entire tracking period, indicating robust and reliable fusion behavior.

We also evaluated the ANEES performance for the other two fusion methods. Their comparison results are not reported here, but we found that the ANEES of the AMD-based

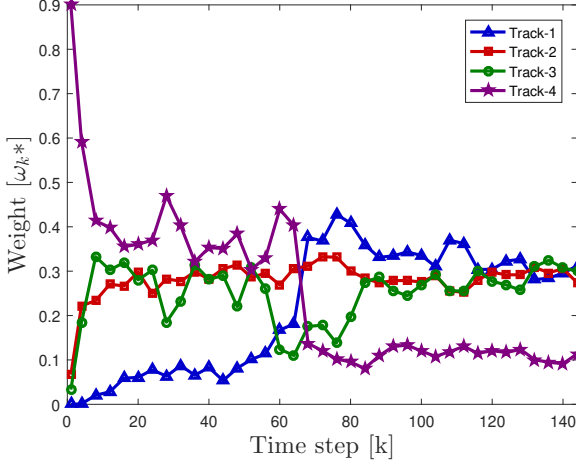


Fig. 3: Weight variation of each track density over the time when target moves with a constant velocity (scenario 1).

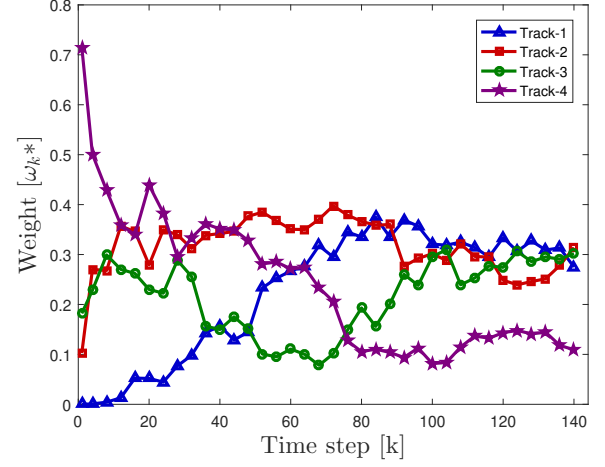


Fig. 4: Weight variation of track densities over the time when target moves with a constant turn rate (scenario-2).

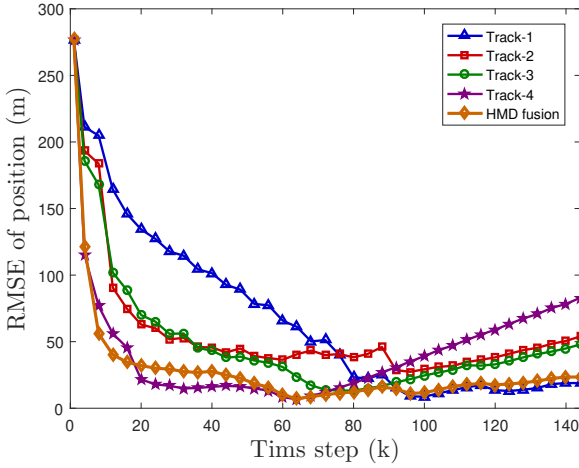


Fig. 5: RMSE of position using SRF for each tracker and the weighted HMD fusion when the target moves with constant velocity (scenario 1).

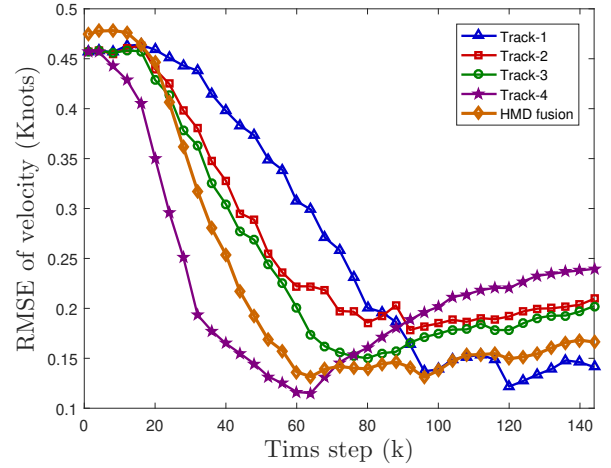


Fig. 6: RMSE of velocity using SRF for each tracker and the weighted HMD fusion when the target moves with constant velocity (scenario 1).

fusion drops below the 95% lower bound, indicating that the designed consensus filter with this method overestimates the uncertainty, even though its mean estimate remains closer to the true target. Instead, the other two methods perform well, with the advantage of HMD fusion being that it avoids non-integer powers of the track density, thereby simplifying implementation.

Table II presents the percentage of track divergence for each sub-region's track under different configurations in both scenarios. It compares the performance without fusion, with HMD fusion, and across different numbers of selected sensors, considering a track bound of 1.0 km. The results show that as the number of sensors increases, the percentage of track divergence decreases. Moreover, applying HMD fusion among the distributed filters reduces the track divergence to zero. These findings demonstrate that the proposed HMD fusion technique effectively eliminates track divergence and enhances the robustness of the tracking system. Furthermore, increasing

the number of sensors contributes to improved tracking accuracy and more consistent performance.

Performances of the various trackers are compared in terms of root mean square error (RMSE) averaged over the time span. It is termed as average root mean square error (ARMSE). ARMSE values for position (in meters) and velocity (in knots) across individual tracks without fusion, are compared to the results obtained using HMD fusion-based tracking in Table III. The comparison clearly demonstrates that the proposed fusion substantially improves tracking accuracy in both position and velocity.

The ARMSE variation was computed by dividing the entire tracking interval into three 12-minute segments, and these segment-wise variations, along with the overall tracking period, are compared using the bar diagrams in Figs. 13 and 14. From these figures, it is clear that our fusion method consistently performs close to the best individual tracker, achieving the lowest overall ARMSE in both scenarios. Another

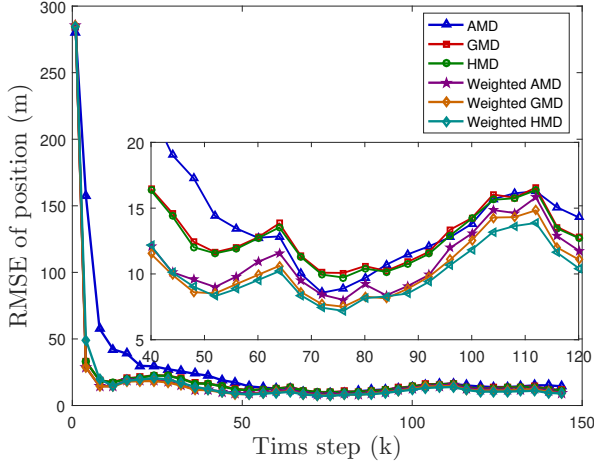


Fig. 7: RMSE of position using SRF for different fusion methods using equal weights and optimized weights when the target moves with constant velocity (scenario 1).

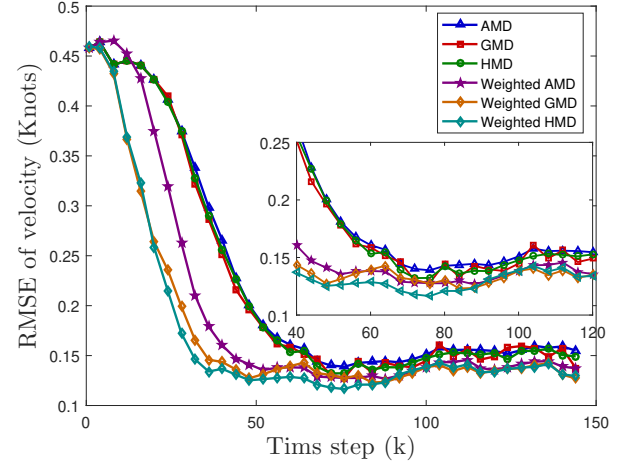


Fig. 8: RMSE of velocity using SRF for different fusion methods using equal weights and optimized weights when the target moves with constant velocity (scenario 1).

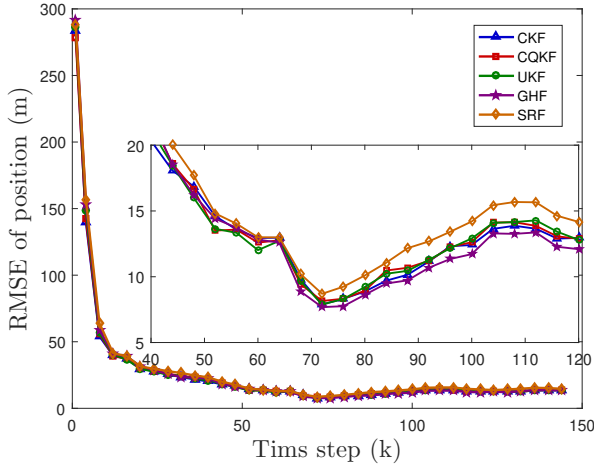


Fig. 9: RMSE of position for HMD fusion using different filtering techniques when the target moves with constant velocity (scenario 1).

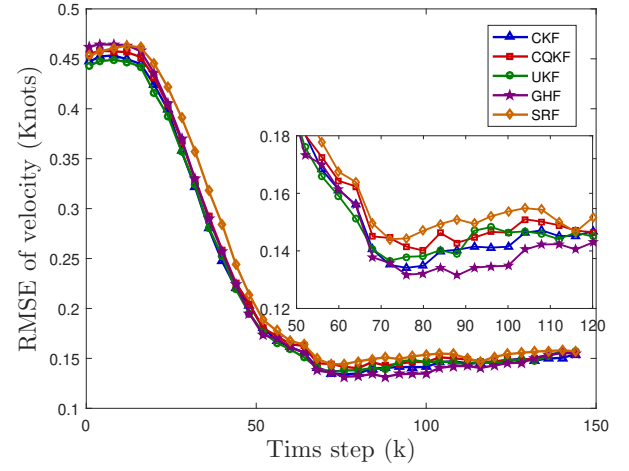


Fig. 10: RMSE of velocity for HMD fusion using different filtering techniques when the target moves with constant velocity (scenario 1).

TABLE II: Percentage of track divergence for track bound = 1.0 Km

|            | Sensors | Track-1 | Track-2 | Track-3 | Track-4 | HMD |
|------------|---------|---------|---------|---------|---------|-----|
| Scenario 1 | $n = 2$ | 0       | 0.65    | 0.70    | 1.90    | 0   |
|            | $n = 3$ | 0       | 0.60    | 0.64    | 1.15    | 0   |
|            | $n = 4$ | 0       | 0.59    | 0.60    | 1.10    | 0   |
| Scenario 2 | $n = 2$ | 0.05    | 0.25    | 1.15    | 2.30    | 0   |
|            | $n = 3$ | 0       | 0.20    | 1.05    | 1.95    | 0   |
|            | $n = 4$ | 0       | 0.20    | 1.00    | 1.90    | 0   |

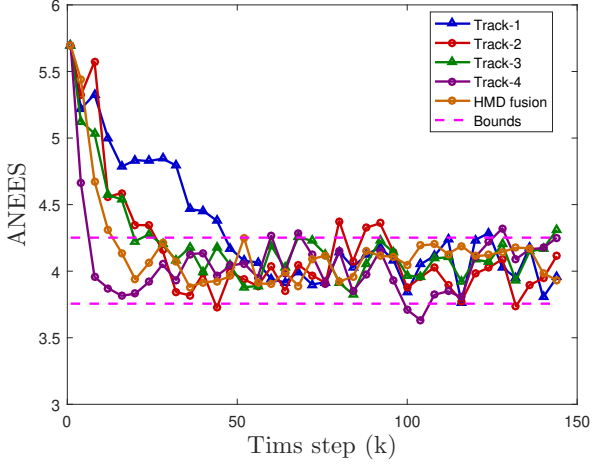


Fig. 11: ANEES using SRF for each tracker and the weighted HMD fusion when the target moves with constant velocity (scenario 1).

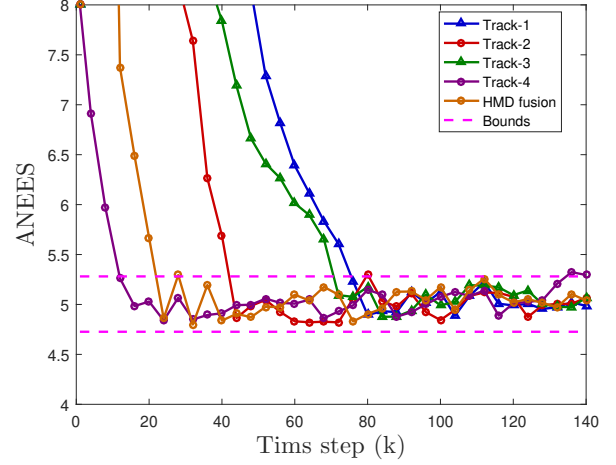


Fig. 12: ANEES using SRF for each tracker and the weighted HMD fusion when the target moves with constant turn rate (scenario 2).

TABLE III: Average root mean square error (ARMSE) for position and velocity

|            | Sensors | Track-1 |        | Track-2 |        | Track-3 |        | Track-4 |        | HMD   |        |
|------------|---------|---------|--------|---------|--------|---------|--------|---------|--------|-------|--------|
|            |         | Pos     | Vel    | Pos     | Vel    | Pos     | Vel    | Pos     | Vel    | Pos   | Vel    |
| Scenario 1 | $n = 2$ | 67.81   | 0.2786 | 55.23   | 0.2660 | 47.19   | 0.2506 | 39.71   | 0.2228 | 22.29 | 0.2191 |
|            | $n = 3$ | 63.42   | 0.2684 | 53.94   | 0.2516 | 55.13   | 0.2408 | 37.14   | 0.2178 | 20.64 | 0.2118 |
|            | $n = 4$ | 61.98   | 0.2648 | 52.37   | 0.2438 | 44.56   | 0.2364 | 36.54   | 0.2142 | 19.36 | 0.2084 |
| Scenario 2 | $n = 2$ | 58.60   | 0.2786 | 39.80   | 0.2527 | 76.10   | 0.3402 | 43.40   | 0.2495 | 22.70 | 0.2333 |
|            | $n = 3$ | 55.40   | 0.2696 | 37.20   | 0.2484 | 71.50   | 0.3312 | 40.20   | 0.2378 | 21.10 | 0.2218 |
|            | $n = 4$ | 53.80   | 0.2658 | 36.40   | 0.2446 | 68.40   | 0.3272 | 38.70   | 0.2342 | 20.80 | 0.2164 |

observation is that the tracker closest to the target provides more accurate estimates, as measurements from nearby sensors exhibit larger variations, improving tracking performance. This relative performance changes over time as the target moves from one subregion to another.

### C. Scenario 2: Target moving with constant turn rate

In this scenario, the target is modeled with a nearly coordinated turn motion, with a process noise intensity of  $q_2 = 0.01 \text{ rad}^2/\text{min}^3$ . All other parameters remain the same as in Scenario 1. The scenario is illustrated in Fig. 2, where the true target and its estimated trajectories are shown by dotted red and blue lines, respectively, with the coordinated turn clearly visible in the true path. The optimal weights for each tracker's posterior densities are shown in Fig. 4. From the weight plot, it can be observed that the consensus fusion initially assigns higher weights to tracker-4, followed by tracker-2, and then tracker-1. This indicates that the fusion process gives higher weight to trackers that are closer to the target, relying more on their pdfs, with the weights dynamically changing over time as the target moves.

The RMSE results of both position and velocity are compared for all four trackers along with their HMD-fusion for

two selected sensors and using SRF as shown in Figs. 15 and 16. The figures show that track fusion-based estimation in consensus tracking always has follow the best performance of individuals in the entire tracking interval and provides faster convergence to all the trackers.

The RMSE results for both position and velocity are compared across different track density fusion methods in consensus tracking using SRF, considering both equal and optimized weights for each tracker. The optimized weights are obtained by minimizing the corresponding cost function, as illustrated in Figs. 17 and 18. The results indicate that consensus tracking with optimized weights outperforms that with equal weights, highlighting the importance of weight optimization for accurate state estimation. Moreover, among the fusion methods employing optimized weights, the harmonic mean density (HMD) and geometric mean density (GMD) approaches exhibit nearly similar performance.

The RMSE results for both position and velocity are compared for weighted consensus tracking using various deterministic sample point filters (DSPFs) [30] and the SRF, as shown in Figs. 19 and 20. As illustrated, all the Gaussian filters designed for tracking perform effectively in the consensus framework, exhibiting nearly identical performance.

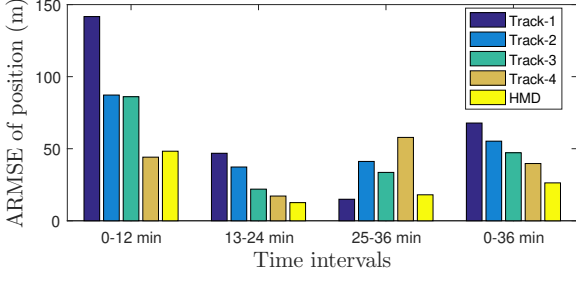


Fig. 13: ARMSE variation over time using SRF for each tracker and the weighted HMD fusion when the target moves with constant velocity (scenario 1).

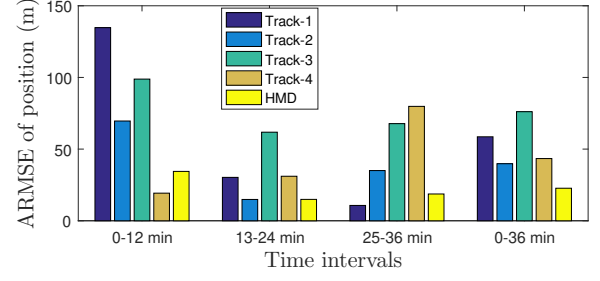


Fig. 14: ARMSE variation over time using SRF for each tracker and the weighted HMD fusion when the target moves with a constant turn rate (scenario 2).

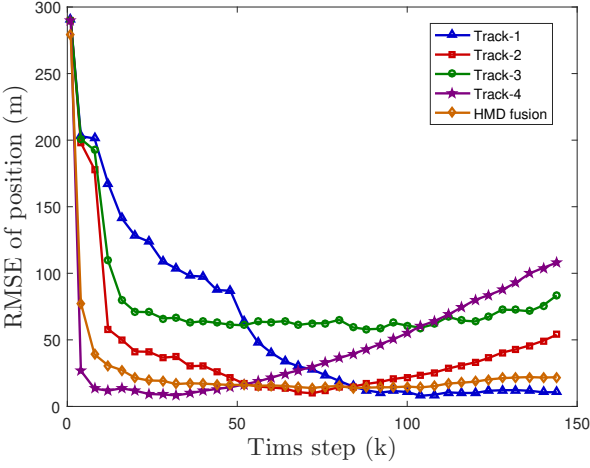


Fig. 15: RMSE of position using SRF for each tracker and the weighted HMD fusion when the target moves with constant turn rate (scenario 2).

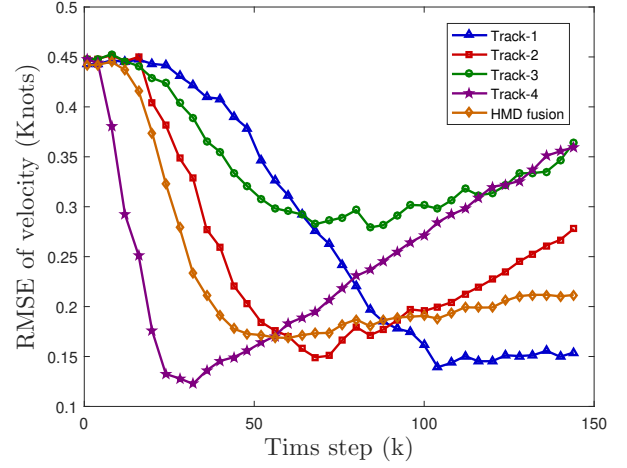


Fig. 16: RMSE of velocity using SRF for each tracker and the weighted HMD fusion when the target moves with constant turn rate (scenario 2).

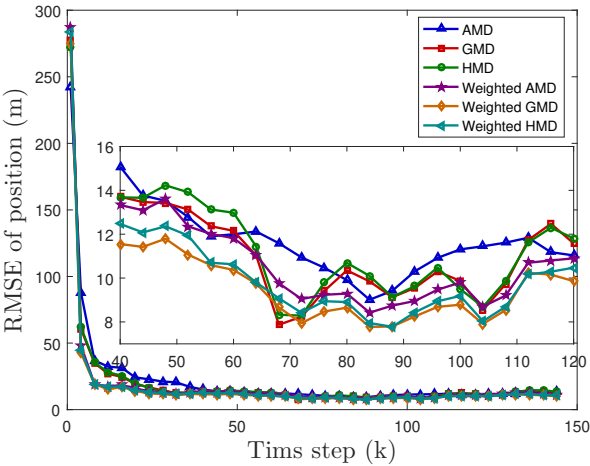


Fig. 17: RMSE of position using SRF for different fusion methods using equal weights and optimized weights when the target moves with constant turn rate (scenario 2).

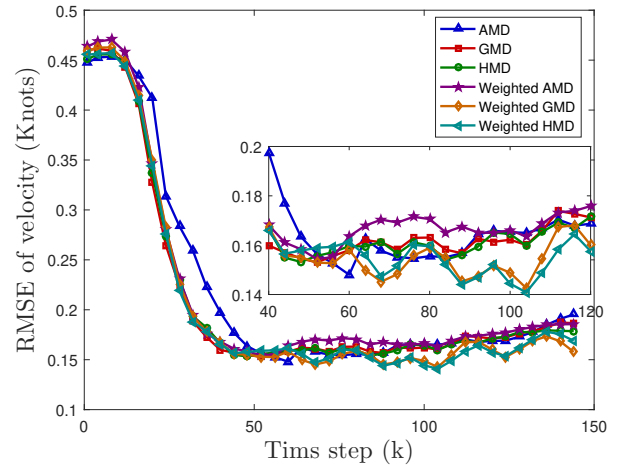


Fig. 18: RMSE of velocity using SRF for different fusion methods using equal weights and optimized weights when the target moves with constant turn rate (scenario 2).

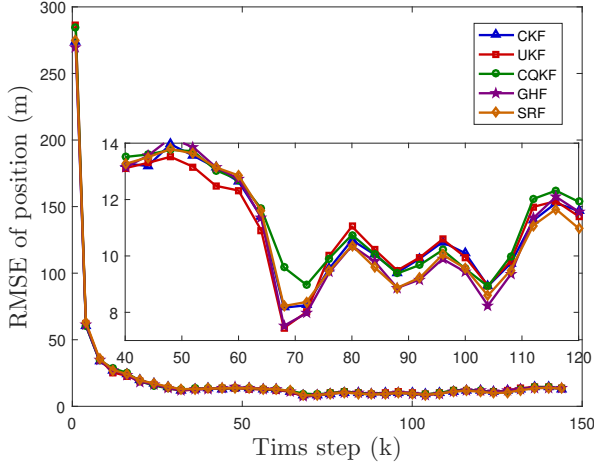


Fig. 19: RMSE of position for HMD fusion using different filtering techniques when the target moves with constant turn rate (scenario 2).

#### APPENDIX A DERIVATION OF THE EXPRESSION (29)

Let us consider two Gaussian pdfs, and can be expressed as

$$p_\ell(\mathcal{X}_k) = \frac{1}{(2\pi)^{\frac{n_{\mathcal{X}}}{2}} |P_{k|k}^\ell|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}^\ell)^\top P_{k|k}^{\ell-1}(\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}^\ell)\right),$$

where  $\ell \in \{i, j\}$ . To compute the divergence, we compute

$$\begin{aligned} \ln \frac{p_i(\mathcal{X}_k)}{p_j(\mathcal{X}_k)} &= \ln \frac{|P_{k|k}^j|^{\frac{1}{2}}}{|P_{k|k}^i|^{\frac{1}{2}}} + \frac{1}{2}(\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}^j)^\top P_{k|k}^{j-1}(\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}^j) \\ &\quad - \frac{1}{2}(\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}^i)^\top P_{k|k}^{i-1}(\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}^i). \end{aligned} \quad (36)$$

By taking the expectation of the above expression over the first pdf  $\mathcal{X}_k \sim \mathcal{N}(\mathcal{X}_k; \hat{\mathcal{X}}_{k|k}^i, P_{k|k}^i)$  is our desired KLD, and it is computed by

$$\mathcal{D}_{\text{KL}}(p_i(\mathcal{X}_k) : p_j(\mathcal{X}_k)) = \mathbb{E}_{\mathcal{X} \sim p_i} \left[ \log \frac{p_i(\mathcal{X}_{k|k})}{p_j(\mathcal{X}_{k|k})} \right]. \quad (37)$$

The Eqn. (36) contains the three terms; to avoid the complexity, let us compute the expectation of each term separately. The first term is independent of  $\mathcal{X}$ , so expectation does not change its value, and it can be written as

$$\mathbb{E} \left[ \log \frac{|P_{k|k}^j|^{\frac{1}{2}}}{|P_{k|k}^i|^{\frac{1}{2}}} \right] = \frac{1}{2} \log \frac{|P_{k|k}^j|}{|P_{k|k}^i|}. \quad (38)$$

The second term of Eqn. (36) is in quadratic form and by substituting  $(\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}^j) = (\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}^i) + (\hat{\mathcal{X}}_{k|k}^i - \hat{\mathcal{X}}_{k|k}^j)$  it can be written as  $[(\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}^j)^\top P_{k|k}^{j-1}(\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}^j)] = (\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}^i)^\top P_{k|k}^{j-1}(\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}^i) + 2(\hat{\mathcal{X}}_{k|k}^i - \hat{\mathcal{X}}_{k|k}^j)^\top P_{k|k}^{j-1}(\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}^i) + (\hat{\mathcal{X}}_{k|k}^i - \hat{\mathcal{X}}_{k|k}^j)^\top P_{k|k}^{j-1}(\hat{\mathcal{X}}_{k|k}^i - \hat{\mathcal{X}}_{k|k}^j)$ , and its expectation is computed by

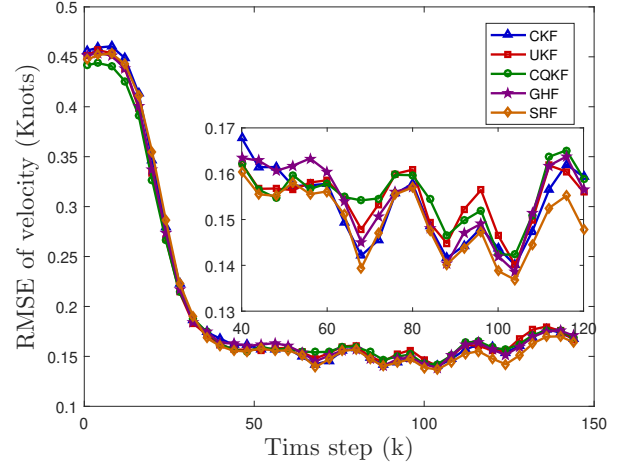


Fig. 20: RMSE of velocity for HMD fusion using different filtering techniques when the target moves with constant turn rate (scenario 2).

tation is computed by

$$\begin{aligned} \mathbb{E}[(\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}^i)^\top P_{k|k}^{j-1}(\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}^i)] &= \text{tr}(P_{k|k}^{j-1} P_{k|k}^i), \\ \mathbb{E}[2(\hat{\mathcal{X}}_{k|k}^i - \hat{\mathcal{X}}_{k|k}^j)^\top P_{k|k}^{j-1}(\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}^i)] &= 0, \\ \mathbb{E}[(\hat{\mathcal{X}}_{k|k}^i - \hat{\mathcal{X}}_{k|k}^j)^\top P_{k|k}^{j-1}(\hat{\mathcal{X}}_{k|k}^i - \hat{\mathcal{X}}_{k|k}^j)] &= (\hat{\mathcal{X}}_{k|k}^i - \hat{\mathcal{X}}_{k|k}^j)^\top P_{k|k}^{j-1}(\hat{\mathcal{X}}_{k|k}^i - \hat{\mathcal{X}}_{k|k}^j). \end{aligned}$$

Thus, the expectation of the entire second term can be expressed as

$$\begin{aligned} \mathbb{E}[\frac{1}{2}(\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}^j)^\top P_{k|k}^{j-1}(\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}^j)] &= \frac{1}{2}(\hat{\mathcal{X}}_{k|k}^i - \hat{\mathcal{X}}_{k|k}^j)^\top P_{k|k}^{j-1}(\hat{\mathcal{X}}_{k|k}^i - \hat{\mathcal{X}}_{k|k}^j) \\ &\quad + \frac{1}{2} \text{tr}(P_{k|k}^{j-1} P_{k|k}^i). \end{aligned} \quad (39)$$

The third term of Eqn. (36) is also in quadratic form, and its expectation can be calculated by

$$\mathbb{E}[\frac{1}{2}(\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}^i)^\top P_{k|k}^{i-1}(\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}^i)] = \text{tr}(P_{k|k}^{i-1} P_{k|k}^i) = \frac{n_{\mathcal{X}}}{2}. \quad (40)$$

By combining all three terms from Eqns. (38), (39), and (40), the KLD for two multivariate Gaussian pdfs can be computed by

$$\begin{aligned} \mathcal{D}_{\text{KL}}(p_i(\mathcal{X}_k) : p_j(\mathcal{X}_k)) &= \frac{1}{2} \text{tr}(P_{k|k}^{j-1} P_{k|k}^i) + \frac{1}{2} \ln \frac{|P_{k|k}^j|}{|P_{k|k}^i|} \\ &\quad - \frac{n_{\mathcal{X}}}{2} + \frac{1}{2}(\hat{\mathcal{X}}_{k|k}^i - \hat{\mathcal{X}}_{k|k}^j)^\top P_{k|k}^{j-1}(\hat{\mathcal{X}}_{k|k}^i - \hat{\mathcal{X}}_{k|k}^j), \end{aligned}$$

where  $n_{\mathcal{X}}$  is the dimension of the distributions and  $|\cdot|$  denotes the determinant of a matrix.

#### REFERENCES

- [1] G. Isbitiren and O. B. Akan, "Three-dimensional underwater target tracking with acoustic sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 8, pp. 3897–3906, 2011.
- [2] H. Ghafoor and Y. Noh, "An overview of next-generation underwater target detection and tracking: An integrated underwater architecture," *IEEE Access*, vol. 7, pp. 98 841–98 853, 2019.

- [3] X. Hou, L. Yang, Y. Yang, J. Zhou, and G. Qiao, "Bearing-only underwater uncooperative target tracking for non-Gaussian environment using fast particle filter," *IET Radar, Sonar & Navigation*, vol. 16, no. 3, pp. 501–514, 2022.
- [4] W. Zhao, X. Li, Z. Pang, and C. Hao, "A novel distributed bearing-only target tracking algorithm for underwater sensor networks with resource constraints," *IET Radar, Sonar & Navigation*, vol. 18, no. 7, pp. 1161–1177, 2024.
- [5] G. J. Macaulay, D. Chu, and E. Ona, "Field measurements of acoustic absorption in seawater from 38 to 360 KHz," *The Journal of the Acoustical Society of America*, vol. 148, no. 1, pp. 100–107, 2020.
- [6] S. Das and S. Bhaumik, "Constrained state estimation for underwater target tracking problem," *Digital Signal Processing*, p. 105394, 2025.
- [7] S. Majumder, S. Scheduling, and H. F. Durrant-Whyte, "Multisensor data fusion for underwater navigation," *Robotics and Autonomous Systems*, vol. 35, no. 2, pp. 97–108, 2001.
- [8] J. Zhang and H. Ji, "Distributed multi-sensor particle filter for bearings-only tracking," *International Journal of Electronics*, vol. 99, no. 2, pp. 239–254, 2012.
- [9] E. Taghavi, R. Tharmarasa, T. Kirubarajan, and M. McDonald, "Multisensor-multitarget bearing-only sensor registration," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, no. 4, pp. 1654–1666, 2016.
- [10] J. Kim, "Tracking multiple objects under bearings-only measurements of multiple heterogeneous sonar sensors in clutter environments," *IEEE Internet of Things Journal*, 2025.
- [11] H. Wei, C. Shen, and Y. Shi, "Distributed Lyapunov-based model predictive formation tracking control for autonomous underwater vehicles subject to disturbances," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 8, pp. 5198–5208, 2019.
- [12] W. Li, G. Wei, F. Han, and Y. Liu, "Weighted average consensus-based unscented Kalman filtering," *IEEE Transactions on Cybernetics*, vol. 46, no. 2, pp. 558–567, 2015.
- [13] L. Jiao, D. Wang, Y. Bai, P. Chen, and F. Liu, "Deep learning in visual tracking: A review," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 9, pp. 5497–5516, 2021.
- [14] Z. Deng and X. Li, "Multisensor data fusion: A review of the state-of-the-art," *Information Fusion*, vol. 14, no. 1, pp. 28–44, 2015.
- [15] R. Tharmarasa, T. Kirubarajan, and M. Hernandez, "Large-scale optimal sensor array management for multitarget tracking," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 37, no. 5, pp. 803–814, 2007.
- [16] X. R. Li and V. P. Jilkov, "Survey of maneuvering target tracking. Part I. Dynamic models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1333–1364, 2003.
- [17] X. Tian and Y. Bar-Shalom, "Track-to-track fusion configurations and association in a sliding window," *Journal of Advances in Information Fusion*, vol. 4, no. 2, p. 146, 2009.
- [18] F. Nielsen, "On the Jensen–Shannon symmetrization of distances relying on abstract means," *Entropy*, vol. 21, no. 5, p. 485, 2019.
- [19] C. Niculescu and L.-E. Persson, *Convex functions and their applications*. Springer, 2006, vol. 23.
- [20] M. B. Hurley, "An information theoretic justification for covariance intersection and its generalization," in *Proceedings of the fifth international conference on Information Fusion. FUSION 2002.(IEEE Cat. No. 02EX5997)*, vol. 1. IEEE, 2002, pp. 505–511.
- [21] R. P. Mahler, "Optimal/robust distributed data fusion: a unified approach," in *Signal Processing, Sensor Fusion, and Target Recognition IX*, vol. 4052. SPIE, 2000, pp. 128–138.
- [22] M. Uney, D. E. Clark, and S. J. Julier, "Information measures in distributed multitarget tracking," in *14th International Conference on Information Fusion*. IEEE, 2011, pp. 1–8.
- [23] N. Sharma, S. Bhaumik, R. Tharmarasa, and T. Kirubarajan, "On pooling-based track fusion strategies: Harmonic mean density," *IET Radar, Sonar & Navigation*, 2024.
- [24] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2004.
- [25] V. P. Dubey, J. Saha, S. Bhaumik, and A. Dey, "Tracking an underwater target in a large surveillance region with sensor location uncertainty," *IEEE Access*, vol. 11, pp. 140 007–140 021, 2023.
- [26] F. Nielsen, "Revisiting Chernoff information with likelihood ratio exponential families," *Entropy*, vol. 24, no. 10, p. 1400, 2022.
- [27] C. H. Kang, S. Y. Kim, and J. W. Song, "Data fusion with inverse covariance intersection for prior covariance estimation of the particle flow filter," *IEEE Access*, vol. 8, pp. 221 203–221 213, 2020.
- [28] Y. Zhang, J. Pan, L. K. Li, W. Liu, Z. Chen, X. Liu, and J. Wang, "On the properties of Kullback-Leibler divergence between multivariate Gaussian distributions," *Advances in Neural Information Processing Systems*, vol. 36, pp. 58 152–58 165, 2023.
- [29] V. P. Dubey, R. K. Tiwari, and S. Bhaumik, "Selection of sensors during tracking a submarine," in *OCEANS 2021: San Diego-Porto*. IEEE, 2021, pp. 1–6.
- [30] S. Bhaumik and P. Date, *Nonlinear estimation: methods and applications with deterministic sample points*. CRC Press, 2019.