PARTIAL ARITHMETIC CONSENSUS BASED DISTRIBUTED INTENSITY PARTICLE FLOW SMC-PHD FILTER FOR MULTI-TARGET TRACKING

Peipei Wu, Jinzheng Zhao, Shidrokh Goudarzi, Wenwu Wang

Centre for Vision, Speech and Signal Processing (CVSSP), University of Surrey, UK

ABSTRACT

Intensity Particle Flow (IPF) SMC-PHD has been proposed recently for multi-target tracking. In this paper, we extend IPF-SMC-PHD filter to distributed setting, and develop a novel consensus method for fusing the estimates from individual sensors, based on Arithmetic Average (AA) fusion. Different from conventional AA method which may be degraded when unreliable estimates are presented, we develop a novel arithmetic consensus method to fuse estimates from each individual IPF-SMC-PHD filter with partial consensus. The proposed method contains a scheme for evaluating the reliability of the sensor nodes and preventing unreliable sensor information to be used in fusion and communication in sensor network, which help improve fusion accuracy and reduce sensor communication costs. Numerical simulations are performed to demonstrate the advantages of the proposed algorithm over the uncooperative IPF-SMC-PHD and distributed particle-PHD with AA fusion.

***Index Terms*—** Distributed tracking, consensus filter, multi-target tracking, particle flow

1. INTRODUCTION

Multi-target tracking is a problem of estimating targets in many engineering applications such as surveillance [1] and audio-visual tracking [2]. In practice, the number of targets and their states are often unknown and time-varying, with the potential presence of clutter and mis-detection in the measurements. A number of methods have been developed to address the tracking problem in such a scenario, such as probability hypothesis density (PHD) filter and its extension Cardinality PHD (CPHD) [3]. There are two main types of PHD filtering algorithms, namely, Gaussian Mixture PHD (GM-PHD) [4] and Sequential Monte Carlo PHD (SMC-PHD) [5].

The research is funded in part by the US Army Research Laboratory and the UK MOD University Defence Research Collaboration (UDRC) in Signal Processing under the SIGNetS project. It is accomplished under Cooperative Agreement Number W911NF-20-2-0225. The views and conclusions contained in this document are of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory, the MOD, the U.S. Government or the U.K. Government. The U.S. Government and U.K. Government are authorised to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

In SMC-PHD, the target states are obtained by the posterior density, which is estimated by the weighted particles. However, this method can be limited by the weight degeneracy problem where most weights become close to zero except only few significant weights [6]. In a recent work, particle flow [6] is used to address this problem, by migrating particles from the prior distribution to the posterior distribution based on a homotopy function. Several particle flow based SMC-PHD filtering algorithms have been developed, using either the zero diffusion particle flow (ZPF) [7], or non-zero diffusion particle flow (NPF) [8], such as, ZPF-SMC-PHD [2, 9], NPF-SMC-PHD [9, 10], and IPF-SMC-PHD [11].

Developing tracking systems with distributed sensors has attracted increasing interest in recent years which could achieve more accurate and reliable tracking [12]. In this paper, we consider extending the IPF-SMC-PHD algorithm [11] into a distributed setting. To integrate the estimates of states from individual PHD filters, a popular choice is to use "geometric average" (GA) fusion, such as the classical algorithm based on covariance intersection [13, 14], where the mean value and covariance of the target states are obtained from the weighted average of the state estimates obtained by individual filters. This method performs well even when correlation between the estimates is unknown. However, this method could be degraded when different targets are close to each other [15], and its performance is also sensitive to possible clutter or mis-detection [16].

An alternative method is to use "arithmetic average" (AA) fusion, where the average of shared estimates from individual filters is used to correct potential errors in local estimates. As compared with the GA method, this method is shown to improve the robustness by compensating a local posterior in distributed PHD filtering, as demonstrated in [15, 17]. However, its accuracy might be degraded due to excessive differences among local estimates in the number of targets and their states. In a recent method for multi-target distributed tracking [18], a distributed set-theoretic information flooding (DSIF) algorithm is used to spread each local estimate across the network, and the global estimates of the model parameters are obtained by using an AA method. This method allows tracking of multiple targets under partial consensus without having to associate the estimates with the targets, however, its tracking accuracy can be degraded by outliers in the local estimates.

This paper presents a distributed IPF-SMC-PHD filter for multi-target tracking, where a new AA fusion method is proposed to reduce the adverse impact of outliers on the tracking performance. This is achieved by evaluating the quality of the estimations with a confidence measure, via calculating the distance between global and local estimates. This new scheme is incorporated into DSIF, leading to an improved fusion algorithm, called C-DSIF-AA, which can prevent unreliable estimates from being shared among the sensor nodes.

2. PROBLEM DEFINITION AND BACKGROUND

2.1. System Model

In this section, we introduce a system model for multi-target distributed tracking. There are S sensors in the network which are indexed by $s \in \mathcal{S} = \{1, 2, \dots, S\}$ and each of them can observe the whole tracking area. The network topology can be represented by a time-varying graph $G(\mathcal{S}, \mathcal{E})$ where edge sets $\mathcal{E} \subseteq \mathcal{S} \times \mathcal{S}$, each edge in $\mathcal{E}$ denotes the connection state between sensors [19]. In a single target tracking system, the target state at time k is defined as $\mathbf{x}_k \in \mathbb{R}^d$ where d is dimension of the target state. The motion model f of target is defined as:

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k) + \mathbf{u}_k, \tag{1}$$

where $\mathbf{u}_k$ is process noise at time k, often assumed as Gaussian distributed. For multi-target tracking, the number of targets is unknown and time-varying. The random finite set (RFS) theory [20] can be used to model multi-target tracking, where the collection of the target state is defined as $\mathcal{X}_k = \{\mathbf{x}_k^1, \mathbf{x}_k^2, \dots, \mathbf{x}_k^{N_k}\}$ where $N_k = |\mathcal{X}_k|$ is the number of targets and $|\cdot|$ is the cardinality of the set. Assume that each target only has one corresponding measurement in each sensor at each time. In addition, the measurement in each sensor may be different due to the use of different measurement methods and measurement noise. Measurements collected by sensor s can be described as RFS: $\mathcal{Z}_{s,k} = \{\mathbf{z}_{s,k}^1, \mathbf{z}_{s,k}^2, \dots, \mathbf{z}_{s,k}^{M_{s,k}}\}$, where $\mathbf{z}_{s,k}^m$ is the m-th measurement and $M_{s,k} = |\mathcal{Z}_{s,k}|$ is the number of measurements from sensor s at time k.

2.2. Intensity Particle Flow

In this section, we discuss an algorithm for multi-target tracking for a single sensor node. Given $\mathcal{Z}_{s,k}$, the SMC-PHD filter can be used to estimate target number $\hat{N}_{s,k+1}$ and target states $\hat{\mathcal{X}}_{s,k+1}$ for the sensor node s. To simplify notations, sensor index s will be ignored in the following content of this sub-section. The filtering result of SMC-PHD could be degraded by the weight degeneracy problem. NPF can mitigate this problem [10], however, it is calculated for each particle with only a single measurement, a data association algorithm is required. In addition, the particles might be associated to other targets or clutter when the target disappears. IPF is proposed to address those problems [11]. Similar to other particle flow algorithms [7, 8], IPF in each sensor is also defined by a homotopy function [21] as follows

$$\begin{aligned}\boldsymbol{f}_k^i = &-[\sum_{m=1}^{M_k} \frac{\lambda_{p_{D,k}} \nabla(\nabla h_k^{i,m})}{G_k^m} + \nabla(\nabla ln(w_{k|k-1}^i))]^{-1} \\ &\times \sum_{m=1}^{M_k} \frac{\lambda_{p_{D,k}} \nabla h_k^{i,m}}{G_k^m},\end{aligned} \tag{2}$$

where $\lambda_{p_{D,k}}$ is the detection probability at each pseudo time step, ∇ is the spatial vector differentiation operation $\frac{\partial}{\partial \widetilde{\mathbf{x}}_{k-1}^i}$ and $\widetilde{\mathbf{x}}_{k-1}^i$ is the i-th particle state at time $k-1$, $w_{k|k-1}^i$ is the weight, $h_k^{i,m}$ is the likelihood of the i-th particle based on the m-th measurement, and G_k^m is defined as:

$$G_k^m = \boldsymbol{k}_k + \sum_{i=N_{k-1}+1}^{N_{k-1}+N_B} S_k^{i,m} + \sum_{i=1}^{N_{k-1}} h_k^{i,m} w_{k|k-1}^i, \tag{3}$$

where $\boldsymbol{k}_k$ is the clutter intensity, N_B is new born particle numbers, $S_k^{i,m}$ is the birth intensity function given as:

$$S_k^{i,m} = \gamma_k(\widetilde{\mathbf{x}}_{k|k-1}^{i,m}) * max(0, 1 - \sum_{i=1}^{N_{k-1}} h_k^{i,m} w_{k|k-1}^{i,m}), \tag{4}$$

in which $\widetilde{x}^m$ is the particle state generated by the m-th measurement, and γ_k is the born probability. These above equations are used to update the particles states and weights, and more details can be found from Algorithm 2 in [11].

2.3. DSIF

To extend a single sensor system into a distributed scenario, local estimates from the distributed sensors need to be shared across sensors and then the estimates on each sensor will be updated following a concensus process. To achieve this, algorithms such as DSIF [18] could be used. This is an iterative algorithm for sharing the local estimates $\{\hat{N}_{s,k+1}, \hat{\mathcal{X}}_{s,k+1}\}$ at sensor s with other sensors. The DSIF algorithm starts from iteration $t = 0$, with the initial set $\mathcal{I}_s(0) = \{\hat{N}_{s,k+1}, \hat{X}_{s,k+1}\}$, and the set $\mathcal{U}_s(0) = \cup_{j \in \mathcal{A}_s} \mathcal{I}_j(0)$, where $\mathcal{A}_s$ is a set containing the indices of the sensors from which the s-th sensor can receive information directly. In each iteration t, $\mathcal{I}_s(t)$ is updated as:

$$\mathcal{I}_s(t) = \mathcal{I}_s(t-1) \cup \mathcal{U}_s(t-1), \tag{5}$$

and $\mathcal{U}_s(t)$ is updated as:

$$\mathcal{U}_s(t) = \cup_{j \in \mathcal{A}_s} \{\mathcal{I}_j(t) \backslash \mathcal{I}_j(t-1)\}, \tag{6}$$

where $\mathcal{A} \backslash \mathcal{B}$ is a set that contains the elements from $\mathcal{A}$ but not from $\mathcal{B}$. If consensus has been achieved at iteration t, $\mathcal{U}_s(t)$ should be $\emptyset$. The update process for $\mathcal{I}_s(t)$ is repeated, until a so-called Degree of Consensus (DoC) C^o (as defined in equation (13) in [18]) reaches a pre-defined threshold T_c.

3. DISTRIBUTED IPF-SMC-PHD

This section presents a distributed IPF-SMC-PHD filter with a novel AA fusion method for a multi-target distributed tracking system. We denote the measurements at time k collected by the s-th sensor as $\mathcal{I} = \mathcal{Z}_{s,k}$. We assume that the local estimates $\{\hat{N}_{s,k+1}, \hat{\mathcal{X}}_{s,k+1}\}$ by each sensor at time k are obtained independently. In practice, local estimates from unreliable sensors may adversely impact on the tracking accuracy and potentially increase communication cost if such estimates are propagated among the sensors. To address this limitation, we propose a new scheme to evaluate the reliability of the local estimates with a confidence measure. This scheme allows those reliable sensors to be selected and communicated for reaching consensus $\{\hat{N}_{k+1}, \hat{\mathcal{X}}_{k+1}\}$ in order to obtain global estimates.

3.1. AA Fusion with Confidence Measure

Assume that the local estimates by sensor s at time k have been shared using the DSIF algorithm at the t-th iteration, the classical AA fusion gives a global estimate for the cardinality and target state at time $k+1$ as follows:

$$\hat{N}_{k+1}(t) = \frac{1}{|\mathcal{I}_s(t)|} \sum_{\hat{N}_{s,k+1} \in \mathcal{I}_s(t)} \hat{N}_{s,k+1}, \tag{7}$$

$$\hat{\mathbf{x}}_{k+1}(t) = \frac{1}{|\mathcal{I}_s(t)|} \sum_{\hat{\mathbf{x}}_{s,k+1} \in \mathcal{I}_s(t)} \hat{\mathbf{x}}_{s,k+1}, \tag{8}$$

where $\hat{\mathbf{x}}_{k+1}(t)$ is an element in $\hat{\mathcal{X}}_{k+1}(t)$. From these two equations, it is clear that $\{\hat{N}_{s,k+1}, \hat{\mathcal{X}}_{s,k+1}\}$ from unreliable sensors may degrade the performance of the AA fusion. Next, we discuss how to obtain more accurate $\{\hat{N}_{s,k+1}(t), \hat{\mathcal{X}}_{s,k+1}(t)\}$.

3.1.1. *Consensus on Cardinality Estimation*

Assume all the sensors can observe the entire scene. We achieve consensus on cardinality at time k, by finding the mode $\hat{\hat{N}}_{k+1}(t)$ of the cardinality estimates in $\mathcal{I}_s(t)$. If the number of appearances of the mode is higher than a pre-defined threshold, with respect to the $|\mathcal{I}_s(t)|$, e.g. 0.6, the mode can be considered as the global cardinality estimate: $\hat{N}_{k+1}(t) = \hat{\hat{N}}_{k+1}(t)$; Otherwise; $\hat{N}_{k+1}(t) = \frac{1}{|\mathcal{I}_s(t)|} \sum_{\hat{N}_{s,k+1} \in \mathcal{I}_s(t)} \hat{N}_{s,k+1}(t)$. Note that $\hat{N}_{k+1}(t)$ should be rounded to its nearest integer.

3.1.2. *Consensus on Target State Estimation*

In the set $\mathcal{I}_s$, some local estimates may not be reliable and could degrade the quality of the global estimate. To address this issue, we propose a method to identify the potential outliers in the local estimates collected by sensor s. In an ideal scenario, the state estimation corresponding to the same target provided by the sensors should be close in space. Therefore, the distance between the estimate will be relatively small. With this in mind, we can form a matrix D_s, whose ab-th element $d_{a,b}$ is calculated as a distance between the estimates from sensors a and b: $d_{a,b} = ||\hat{\mathbf{x}}_a - \hat{\mathbf{x}}_b||^2$. Note the time k and iteration t have been dropped from $\mathcal{I}_s(t)$ for simplicity. We then compare $d_{a,b}$ against a pre-defined threshold ϵ, e.g. the radius of the particle birth region in SMC-PHD, and if $d_{a,b} < \epsilon$, these two estimates are considered close in space. The s-th column of the matrix D_s shows the distance of estimations between sensor s and other sensors in $\mathcal{I}_s$, if most elements (e.g. 80% elements) in this column are considered close, the estimation given by sensor s is considered as reliable and noted as $\hat{\hat{\mathbf{x}}}_s$; otherwise, $\hat{\mathbf{x}}_s$ will be considered as outliers. Moreover, only reliable estimation $\hat{\hat{\mathbf{x}}}_s$ will be used for obtaining consensus on the state $\hat{\mathbf{x}}$ by weighted average and the indices of reliable sensors are placed in set $\mathcal{L}_s$. The weight corresponding to each reliable estimation $\hat{\hat{\mathbf{x}}}$ is calculated as $w_s = o_s / \sum_{b \in \mathcal{L}} o_b$, where $o_s = 1/\sum d_{s,a}^{\hat{\mathbf{x}}}$, and $a, b \in \mathcal{L}, a \neq s$. As a result, the final fusion is obtained as $\hat{\mathbf{x}} = \sum w_s \hat{\hat{\mathbf{x}}}_s$, which forms $\hat{\mathcal{X}}_{k+1}(t) = \{\hat{\mathbf{x}}^1, \hat{\mathbf{x}}^2, \ldots, \hat{\mathbf{x}}^{\hat{N}_{k+1}}\}$ at time k.

3.2. Confidence-Informed DSIF Algorithm: C-DSIF

After obtaining the global estimate $\hat{N}_{k+1}$ and $\hat{\mathcal{X}}_{k+1}$, we can determine whether the local estimates $\{\hat{N}_{s,k+1}, \hat{\mathcal{X}}_{s,k+1}\}$ at each sensor s are reliable or not. This can be achieved by a confidence measure in terms of the difference (or distance) between global estimate and local estimates, i.e. $d_s^{\hat{N}} = |\hat{N}_s - \hat{N}|$, and $d_s^{\hat{\mathbf{x}}} = ||\hat{\mathbf{x}}_s - \hat{\mathbf{x}}||^2$. If $d_s^{\hat{N}} > d_T$, where d_T is a predefined threshold, or if $d_s^{\hat{\mathbf{x}}} > \epsilon$, the sensor s is considered as unreliable, and the local estimates $\hat{N}_s$ and $\hat{\mathbf{x}}_s$ from sensor s will not be shared with other sensors at the time $k+1$. A flag R_{k+1} will be assigned as $TRUE$. A re-initialization is then required for the local IPF-SMC-PHD filter in sensor s. Algorithm 1 shows the overall process of the D-IPF-SMC-PHD algorithm which includes the new consensus algorithm C-DSIF-AA.

4. NUMERICAL RESULTS

We simulated four targets moving in a squared region of 40 m $\times$ 40 m, with 20 time frames. Each target follows the constant velocity model as in (1). There are 20 sensors in the network, and the maximum connection hop is 5. Each sensor follows a measurement model: $\mathbf{z}_k = f_z(\mathbf{x}_k) + \mathbf{v}_k$ with different measurement noise $\mathbf{v}_k$. T_c is set as 0.5, which means at most half of the sensors will be utilized for consensus. Other parameters are given as: $\epsilon = 6$, $d_T = 2$. The proposed algorithm (D-IPF-SMC-PHD) is compared with an uncooperative method (IPF-SMC-PHD in a single sensor), IPF-SMC-PHD with complete average consensus using all the sensors (CA-

Algorithm 1 D-IPF-SMC-PHD

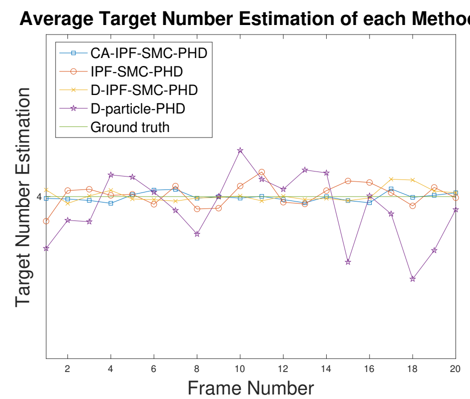
Input: $\{\mathcal{Z}_{1,k}, \cdots, \mathcal{Z}_{S,k}\}$, $\{\mathcal{A}_1, \cdots \mathcal{A}_S\}$, k, S, T_c, ϵ, d_T, R_k
Output: $\hat{N}_{k+1}$, $\hat{X}_{k+1}$, R_{k+1}

Use IPF-SMC-PHD to estimate $\{\hat{N}_{s,k+1}, \hat{\mathcal{X}}_{s,k+1}\}$ from $\{\mathcal{Z}_{s,k}\}$, for all $s = 1, ..., S$.
$t \leftarrow 0$; $C^o(0) \leftarrow 0$; $I_s(t) \leftarrow \{\hat{N}_{s,k+1}, \hat{\mathcal{X}}_{s,k+1}\}, \forall s$.
if $R_k == TRUE$ **then**
    $\mathcal{I}_s(t) \leftarrow \emptyset$.
end if
while $C^o(t) < T_c$ **do**
    $t \leftarrow t + 1$.
    if $t == 1$ **then**
        $\mathcal{I}_s(t) \leftarrow \mathcal{I}_s(t-1) \cup \{\cup_{\forall j \in A_s} \mathcal{I}_j(t-1)\}$.
    else
        Calculate $\mathcal{U}_s(t)$ using (6).
        Update $\mathcal{I}_s(t)$ using (5).
    end if
    Calculate $C^o(t)$.
end while
Calculate $\hat{N}_{k+1}$ and $\hat{\mathcal{X}}_{k+1}$ as discussed in Section 3.1.
if $R_k == FALSE$ **then**
    Calculate R_{k+1} as discussed in Section 3.2.
else
    Set $R_{k+1} = FALSE$.
end if
Return $\hat{N}_{k+1}$, $\hat{\mathcal{X}}_{k+1}$, R_{k+1}.

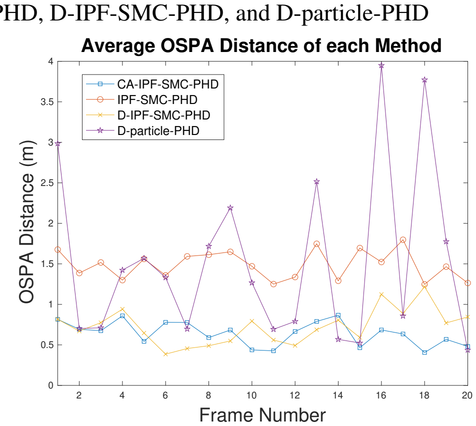
IPF-SMC-PHD), and distributed particle-PHD with DSIF on GM parameters (D-particle-PHD) [22]. Each algorithm was run 100 times independently in the above simulation scenario with the same ground truth. In each run, the measurement might be slightly different, but the measurements given to each algorithm compared are the same.

Figure 1a shows the average of the estimated number of targets by each algorithm over 100 runs at each time frame. The numbers are not in integers after averaging over the 100 tests. It can be observed that the estimate given by our proposed algorithm resembles more closely with ground truth than IPF-SMC-PHD and D-particle-PHD. The difference between CA-IPF-SMC-PHD and D-IPF-SMC-PHD is not significant. Considering the fact that our proposed algorithm reaches consensus under partial consensus, not every sensor will be used for consensus. Hence, our proposed algorithm provides improved robustness over the complete average consensus, as it can be used for the scenario when some sensors in the network fail.

Figure 1b shows the tracking performance of the proposed method, in terms of OSPA [23], as compared with the baselines. Clearly, our proposed algorithm provides better accuracy than IPF-SMC-PHD and D-particle-PHD; the average OSPA of D-IPF-SMC-PHD fluctuates in the range of $[0.5, 1.2]$. The average OSPA of D-particle-PHD fluctuates more drastically over the time frames, than the proposed method. D-IPF-SMC-PHD performs similarly to CA-IPF-SMC-PHD, but uses less information. Regarding the efficiency, our proposed method runs under partial consensus condition, thus we can reduce half communication cost in complete consensus. The average number of communication iterations of D-IPF-SMC-PHD, D-particle-PHD and CA-IPF-SMC-PHD is 4.4, 5.1, and 6.3, respectively, in each sensor. The average time cost of D-IPF-SMC-PHD, D-particle-PHD, and CA-IPF-SMC-PHD is 41.414s, 41.99s, and 42.21s, respectively. Noted that the local filters were not run in parallel, which consumed the biggest portion of time in simulations.



(a) The number of targets estimated by CA-IPF-SMC-PHD, IPF-SMC-PHD, D-IPF-SMC-PHD, and D-particle-PHD



(b) The OSPA of targets estimated by CA-IPF-SMC-PHD, IPF-SMC-PHD, D-IPF-SMC-PHD, and D-particle-PHD

Fig. 1: Simulation results for all the compared algorithms.

5. CONCLUSION

We have presented a novel AA fusion method to extend IPF-SMC-PHD into a distributed version under partial consensus condition, with improved tracking accuracy and reduced computational/communication cost. Simulation results show that the proposed method outperforms several recent baseline methods. In the future, we will study the performance of D-IPF-SMC-PHD for target tracking when sensor nodes are limited in sensing range.

6. REFERENCES

[1] B. Benfold and I. Reid, "Stable multi-target tracking in real-time surveillance video," in *CVPR*, 2011, pp. 3457–3464.

[2] Y. Liu, W. Wang, J. Chambers, V. Kilic, and A. Hilton, "Particle flow smc-phd filter for audio-visual multi-speaker tracking," in *International Conference on Latent Variable Analysis and Signal Separation*. Springer, 2017, pp. 344–353.

[3] F. Yang, Y. Wang, Y. Liang, and Q. Pan, "A survey of phd filter based multi-target tracking," *Acta automatica sinica*, vol. 39, no. 11, pp. 1944–1956, 2013.

[4] D. E. Clark, K. Panta, and B.-N. Vo, "The gm-phd filter multiple target tracker," in *9th International Conference on Information Fusion*. IEEE, 2006, pp. 1–8.

[5] V. Kılıç, M. Barnard, W. Wang, A. Hilton, and J. Kittler, "Mean-shift and sparse sampling-based smc-phd filtering for audio informed visual speaker tracking," *IEEE Transactions on Multimedia*, vol. 18, no. 12, pp. 2417–2431, 2016.

[6] F. Daum and J. Huang, "Nonlinear filters with log-homotopy," in *Signal and Data Processing of Small Targets*, vol. 6699. International Society for Optics and Photonics, 2007, p. 669918.

[7] P. Bunch and S. Godsill, "Approximations of the optimal importance density using gaussian particle flow importance sampling," *Journal of the American Statistical Association*, vol. 111, no. 514, pp. 748–762, 2016.

[8] F. Daum and J. Huang, "Particle flow with non-zero diffusion for nonlinear filters," in *Signal Processing, Sensor Fusion, and Target Recognition XXII*, vol. 8745. International Society for Optics and Photonics, 2013, p. 87450P.

[9] Y. Liu, V. Kilic, J. Guan, and W. Wang, "Audio-visual particle flow smc-phd filtering for multi-speaker tracking," *IEEE Transactions on Multimedia*, vol. 22, no. 4, pp. 934–948, 2020.

[10] Y. Liu, A. Hilton, J. Chambers, Y. Zhao, and W. Wang, "Non-zero diffusion particle flow smc-phd filter for audio-visual multi-speaker tracking," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4304–4308.

[11] Y. Liu, W. Wang, and V. Kilic, "Intensity particle flow smc-phd filter for audio speaker tracking," *arXiv preprint arXiv:1812.01570*, 2018.

[12] S. Bhatti and J. Xu, "Survey of target tracking protocols using wireless sensor network," in *Fifth International Conference on Wireless and Mobile Communications*. IEEE, 2009, pp. 110–115.

[13] S. Julier and J. K. Uhlmann, "General decentralized data fusion with covariance intersection," in *Handbook of Multisensor Data Fusion*. CRC Press, 2017, pp. 339–364.

[14] T. Bailey, S. Julier, and G. Agamennoni, "On conservative fusion of information with unknown non-gaussian dependence," in *2012 15th International Conference on Information Fusion*. IEEE, 2012, pp. 1876–1883.

[15] T. Li, J. M. Corchado, and S. Sun, "On generalized covariance intersection for distributed phd filtering and a simple but better alternative," in *20th International Conference on Information Fusion (Fusion)*. IEEE, 2017, pp. 1–8.

[16] L. Gao, G. Battistelli, and L. Chisci, "Multiobject fusion with minimum information loss," *IEEE Signal Processing Letters*, vol. 27, pp. 201–205, 2020.

[17] T. Li, V. Elvira, H. Fan, and J. M. Corchado, "Local-diffusion-based distributed smc-phd filtering using sensors with limited sensing range," *IEEE Sensors Journal*, vol. 19, no. 4, pp. 1580–1589, 2018.

[18] T. Li, J. M. Corchado, and J. Prieto, "Convergence of distributed flooding and its application for distributed bayesian filtering," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 3, no. 3, pp. 580–591, 2016.

[19] R. Olfati-Saber and J. Shamma, "Consensus filters for sensor networks and distributed sensor fusion," in *Proceedings of the 44th IEEE Conference on Decision and Control*, 2005, pp. 6698–6703.

[20] R. P. Mahler, *Statistical Multisource-Multitarget Information Fusion*. Artech House, Inc., 2007.

[21] B.-N. Vo, S. Singh, A. Doucet *et al.*, "Sequential monte carlo implementation of the phd filter for multi-target tracking," in *Proc. Int'l Conf. on Information Fusion*, 2003, pp. 792–799.

[22] T. Li and F. Hlawatsch, "A distributed particle-phd filter using arithmetic-average fusion of gaussian mixture parameters," *Information Fusion*, vol. 73, pp. 111–124, 2021.

[23] B. Ristic, B.-N. Vo, D. Clark, and B.-T. Vo, "A metric for performance evaluation of multi-target tracking algorithms," *IEEE Transactions on Signal Processing*, vol. 59, no. 7, pp. 3452–3457, 2011.