A Modified δ-Generalized Labeled Multi-Bernoulli Filtering for Multi-Source DOA Tracking with Coprime Array

Xudong Dong, Jun Zhao, Meng Sun, Xiaofei Zhang, Yide Wang

A Modified δ-Generalized Labeled Multi-Bernoulli Filtering for Multi-Source DOA Tracking with Coprime Array

Xudong Dong, Jun Zhao, Meng Sun, Xiaofei Zhang, Yide Wang

***Abstract*—For the target tracking problem where the number of targets fluctuates with time and the measurement is a point measurement, the random finite set (RFS) class filtering is an available solution. However, in direction of arrival (DOA) tracking, the array observation is a super-positional value, and the tracking performance can be severely impaired if the RFS-based filter approach is applied. As a result, a novel measurement association mapping (NMAP) approach has been presented to cope with the mapping problem between the array observations and sources. Nevertheless, the tracking performance is poor when the number of particles is small. In this paper, a modified delta-Generalized Labeled Multi-Bernoulli (δ-GLMB) DOA tracking particle filter is proposed in combination with the NMAP strategy, which can achieve the same tracking performance with a smaller number of particles by modifying the particles in the δ-GLMB prediction step. Furthermore, the approach is extended to a coprime array and can achieve better DOA tracking performance than a uniform linear array. Simulation experiments validate the effectiveness of the proposed algorithm.**

***Index Terms*—Direction-of-arrival (DOA) tracking, coprime array, δ-generalized labeled multi-Bernoulli filter (δ-GLMB), super-positional measurement.**

I. Introduction

MULTI-target filtering concerns the estimation of the number of unknown time-varying targets and their individual states (e.g., $x-y$ position, direction of arrival (DOA), etc.) from a series of observations [1]. Although the words multi-target filtering and multi-target tracking are often used interchangeably, there is a distinction. In multi-target tracking, we are additionally interested in the target's trajectory (in fact, a real multi-target tracking system requires tracking labels). There are a lot of recent works on multi-target tracking applications that can track the position of targets [2], or their DOAs [3]–[7]. The work presented in this paper is focused on a Bayesian multi-target filtering that also provides multi-source DOA tracking (called as multi-source tracking for convenience).

The most difficult aspect of multi-source tracking is dealing with super-positional measurement, which is the superposition of information from multi-source. Approaches based on subspace update [3]–[7] have been proposed to address this concern, but these methods all presume a known and fixed number of sources, which is not applicable in practical circumstances. The random finite set (RFS, [2]) approach is a Bayesian version of the multi-target filtering/tracking problem in which the number of targets is random and the set of target states is considered as a finite set. The RFS theory-supported multi-target Bayesian filtering can detect dynamically and simultaneously the number and state of sources, where new sources appear (newborn) and old ones disappear (death), and has a wide range of application areas, such as sonar [8], computer vision [9], [10], traffic monitoring [11], sensor network and distributed estimation [12]–[14]. Due to the numerical complexity of Bayesian multi-target filter, the probability hypothesis density (PHD) [15], Cardinalized PHD (CPHD) [16] and multi-Bernoulli filters [17], have been developed as approximations. In principle, these approaches are not multi-target trackers because they are based on the indistinguishability of the targets (i.e., they cannot track the trajectory labels). Recently, Vo et al. propose the generalized labeled multi-Bernoulli (GLMB, [18]) algorithm based on RFS theory. Compared with the GLMB filtering, δ-GLMB [19] filtering owns stronger results that are immediately applied for multi-target tracking. However, the authors of [20] consider that in traditional sensor array observations, each sensor element's measurement is formed by a mixture of all the sources in the surveillance area, and this measurement model is referred to as the super-positional measurement model, which causes a mismatch in the source-measurement association mapping and thus reduces the tracking accuracy [21].

Traditional methods use, firstly, detection algorithms to convert the super-positional measurement model into a standard measurement model, and then PHD/CPHD filters to extract separable measurements. However, critical information will be lost during the conversion process, resulting in inaccurate estimations. RFS-based approaches are also widely employed in the field of DOA tracking [20]–[25]. The new measurement association mapping (NMAP) strategy proposed in [22] redefines the matching mechanism between the source and measurement, thereby resolving the problem of tracking performance degradation due to the incorrect association mapping [21]. However, the performance is poor at low signal-to-noise ratio (SNR). Using the NMAP strategy, the PHD DOA tracking method [23] based on coprime array is proposed, which improves the tracking accuracy and increases the number of detectable sources compared to the classical particle filter (PF, [6]) DOA tracking algorithm. In [20], [25], the multi-source DOA tracking problem in the RFS framework is solved by a CPHD filter. However, the aforementioned approaches will suffer from particle deterioration and will be unable to track the trajectory labels.

With respect to the previously discussed DOA tracking tech-

(Corresponding author: Meng Sun.)

X. Dong, X. Zhang and M. Sun are with the College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210000, China, and also with the Key Laboratory of Dynamic Cognitive System of Electromagnetic Spectrum Space (e-mail: nanhangdxd@nuaa.edu.cn; zhangxiaofei@nuaa.edu.cn; mengsun@nuaa.edu.cn).

J. Zhao is with the College of Electronic and Information Engineering, Tongji University, Shanghai 201800, China (e-mail: jun325709@163.com).

Yide Wang is with the IETR, University of Nantes, 44306 Nantes, France.

niques based on the RFS framework, the tracking performance of approach using sparse array [6], [23] is noticeably superior to that of method based on uniform linear array (ULA) [7], [21], [22]. This is due to the fact that, for a given number of sensors, the sparse array approach has a larger virtual array aperture than the ULA, which results in improved estimate performance. In this paper, combining the existing NMAP strategy [22], we extend the δ-GLMB filtering to the coprime array [26] scenario and propose a modified δ-GLMB (Mδ-GLMB) filtering. The main contributions are as follows:

- We extend the GLMB filtering to the coprime array DOA tracking scenario to increase the degrees of freedom, consequently the tracking perfromance and the number of sources.
- We present a modified δ-GLMB filtering and provide a sequential Monte Carlo implementation (also called particle filtering) scheme.
- In the prediction step of the proposed Mδ-GLMB method, a regularization method for the predicted particles is introduced to rectify the application of the particles, thus making them easier to approach the central area of the posterior probability density function.
- The proposed method can also be extended to the other sparse array geometries, like nested arrays, super nested arrays, etc.

The paper is organized as follows. Section II introduces the relevant backgrounds, including the source state model, measurement model and δ-GLMB filtering. In section III, we outline the recursive implementation of the δ-GLMB DOA tracking filtering. Section IV presents the proposed modified δ-GLMB filtering. Simulation and conclusion are given in sections V and VI, respectively.

Notations 1: Upper-case (lower-case) bold characters stand for matrices (vectors). $[\![a,b]\!]$ denotes the set $\{x\in\mathbb{Z}\,|a\leqslant x\leqslant b\}$, $(\cdot)^T$, $(\cdot)^H$ and $(\cdot)^*$ denote the transpose, conjugate transpose and conjugate of matrix, respectively. $diag(\cdot)$ and $vec(\cdot)$ indicate the diagonal matrices and vectorization operation, respectively. $\otimes$ and $\odot$ denote Kronecker and Khatri-Rao products, respectively. $\mathbf{I}_N$ denotes a $N\times N$ identity matrix and $\mathbf{0}_N$ is the $N\times N$ zero matrix, $\mathbb{E}$ is the expectation operator.

Notations 2: A single target state is denoted by lowercase letters (e.g., x, $\mathbf{x}$), while multi-target states are expressed as uppercase letters (e.g., X, $\mathbf{X}$), symbols for labeled states and their distributions are bolded to distinguish them from unlabeled ones (e.g., $\mathbf{x}$, $\mathbf{X}$, $\boldsymbol{\pi}$, etc.), spaces are represented by blackboard bold (e.g., $\mathbb{X}$, $\mathbb{R}$, $\mathbb{L}$, $\mathbb{C}$, etc.), state space is defined by Euclid math one font (e.g., $\mathcal{X}$, $\mathcal{Y}$, etc.), and the class of finite subsets of a space $\mathcal{X}$ or $\mathbb{X}$ is represented by $\mathcal{F}(\mathcal{X})$ or $\mathcal{F}(\mathbb{X})$, respectively. Both $:=$ and $\triangleq$ stand for definition or equivalence.

II. Background

This section presents the background of multi-source DOA tracking based on δ-GLMB filtering, including the source state model, measurement model, and basic theory of the δ-GLMB filtering recursion.

A. Source state model

Assume that there are N_k sources with state $\mathbf{x}_k^n = [\theta_{n,k}, \dot{\theta}_{n,k}]^T, n=1,\cdots,N_k$ move with a velocity $\dot{\theta}_{n,k}$ $(^\circ/s)$, where $\theta_{n,k}$ denotes the DOA of the n-th source at time k. The constant velocity (CV) model is given as follows

$$\mathbf{x}_k^n = \mathbf{F}_k\mathbf{x}_{k-1}^n + \mathbf{G}_k v_k, \tag{1}$$

with

$$\mathbf{F}_k = \begin{bmatrix} 1 & \Delta T \\ 0 & 1 \end{bmatrix}; \mathbf{G_k} = \begin{bmatrix} \Delta T^2/2 \\ \Delta T \end{bmatrix}, \tag{2}$$

where $\mathbf{F}_k$ and $\mathbf{G}_k$ are coefficient matrices, ΔT represents the time period and v_k is a Gaussian white noise term with distribution $\mathcal{N}\left(0,\sigma_k^2\right)$, where σ_k^2 is the noise variance.

Remark 1: In this paper, we add label to each source, and a unique label l of the α-th source $\mathbf{x}_k^\alpha$ consisting of an ordered pairs (k,α), where α is the index of targets born at time k (e.g. there are 4 targets born at time 2, then their labels are (2, 1), (2, 2), (2, 3), (2, 4), respectively.). The label space of all targets up to time k can be expressed as a disjoint union $\mathbb{L}_k = \bigcup_{h=1}^k \mathbb{B}_h$, where $\mathbb{B}_h$ denotes the label space of targets born at time h (noting that $\mathbb{L}_k = \mathbb{L}_{k-1}\cup\mathbb{B}_k$). Formally, the states in Eq. (1) can be represented as a labeled vector $\mathbf{x}_k^n := (\mathbf{x}_k^n, l_k^n)$ with label $l_k^n = (k,n)$, and the trajectory of a target is composed of a sequence of consecutive labeled states with the same label [18].

B. Measurement model

Fig. 1 shows the extended coprime array (ECA) configuration, the array locations are denoted as:

$$\mathbb{P} = \{Nmd\,|m\in[\![0,2M-1]\!]\}\cup\{Mnd\,|n\in[\![0,N-1]\!]\}, \tag{3}$$

where M and N $(M<N)$ are coprime numbers, let $r_1<r_2<\cdots<r_P, r_i\in\mathbb{P}, i=1,\cdots,P, r_1=0$, and $P=2M+N-1$. $d=\lambda/2$ is the minimum array spacing with λ denoting the wavelength.

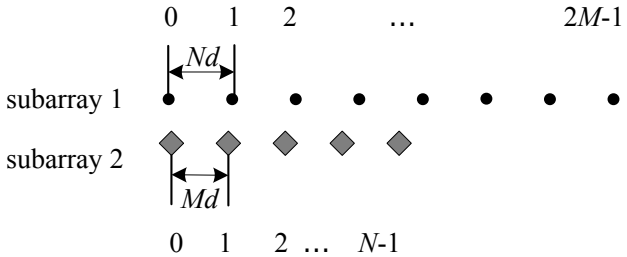


Fig. 1: Extended coprime array (ECA).

Considering N_k narrowband far field incoherent sources $s_n(k), n=1,2,\cdots,N_k$ with DOA $\theta_{n,k}$, impinging on an extended coprime array (shown in Fig. 1) with P sensors. The measurement model is as follows

$$\mathbf{y}_k = \mathbf{A}(\theta)\,\mathbf{s}_k + \mathbf{n}_k, \tag{4}$$

where

- $\mathbf{y}_k = [y_1(k), y_2(k), \cdots, y_P(k)]^T$ is the superpositional measurement;

- $\mathbf{s}_k = [s_1(k), s_2(k), \cdots, s_{N_k}(k)]^T$ is the source vector and $\mathbf{n}_k$ represents the additive Gaussian white noise (AGWN) vector with covariance matrix $\sigma^2\mathbf{I}_P$, where σ^2 is the noise power;
- $\mathbf{A}(\theta) = [\mathbf{a}(\theta_{1,k}), \mathbf{a}(\theta_{2,k}), \cdots, \mathbf{a}(\theta_{N_k,k})] \in \mathbb{C}^{P\times N_k}$ is the directional matrix with

$$\mathbf{a}(\theta_{n,k}) = \left[1, \cdots, e^{-j\frac{2\pi}{\lambda}r_P \sin\theta_{n,k}}\right]^T \tag{5}$$

the steering vector.

The covariance matrix $\mathbf{R}_k$ of $\mathbf{y}_k$ (4) is given by

$$\mathbf{R}_k = \mathbb{E}\left\{\mathbf{y}_k\mathbf{y}_k^H\right\} = \mathbf{A}(\theta)\mathbf{R}_s\mathbf{A}^H(\theta) + \sigma^2\mathbf{I}_P, \tag{6}$$

where $\mathbf{R}_s = \mathbb{E}\left\{\mathbf{s}_k\mathbf{s}_k^H\right\} \in \mathbb{C}^{N_k\times N_k}$ is the signal covariance matrix, and $\mathbf{R}_k$ can be estimated as

$$\tilde{\mathbf{R}}_k \approx \frac{1}{T_k}\sum\nolimits_{t=1}^{T_k}\mathbf{y}_k(t)\mathbf{y}_k^H(t), \tag{7}$$

where T_k denotes the number of snapshots at time k. By adopting the coprime array technique [26] (including vectorization, elimination of redundant terms and other operations), we can obtain a virtual uniform linear array single snapshot vector $\mathbf{z}$, as follows

$$\mathbf{z} = \tilde{\mathbf{A}}\mathbf{b} + \sigma^2\mathbf{1}_{MN+M}, \tag{8}$$

where $\tilde{\mathbf{A}}$ is the virtual directional matrix of the ULA with $2M(N+1)-1$ virtual sensors located from $(-M(N+1)+1)d$ to $(M(N+1)-1)d$; $\mathbf{b} = [\sigma_1^2, \cdots, \sigma_{N_k}^2]^T$ denotes the signal vector with σ_n^2 the power of the n-th source; $\mathbf{1}_{MN+M} \in \mathbb{R}^{(2M(N+1)-1)\times 1}$ is a zero vector except that the $MN+M$-th element equals to 1.

Notice that the vector $\mathbf{z}$ is a single snapshot signal vector. By employing spatial smoothing method, a rank restored data covariance matrix [27] can be reconstructed by

$$\bar{\mathbf{R}}_{ss} = \frac{1}{G}\sum\nolimits_{i=1}^{G}\mathbf{z}_i\mathbf{z}_i^H, \tag{9}$$

where $\mathbf{z}_i = \mathbf{z}(i : G+i-1, :)$ denotes a vector consisting of the elements ranging from ith to $(G+i-1)$th of $\mathbf{z}$, and $G = M(N+1)$ denotes the total number of spatial smoothing subarray elements. Then $\bar{\mathbf{R}}_{ss}$ is a full-rank matrix so that the MUSIC [28] method can be performed for DOA estimation.

C. δ-GLMB FILTER

1) Labeled RFS: According to [18], a generalization of the Kronecker delta function can be denoted as

$$\delta_Y(X) \triangleq \begin{cases} 1, \text{if } X = Y \\ 0, \text{otherwise} \end{cases}, \tag{10}$$

and the generalized indicator function $1_Y(X)$

$$1_Y(X) \triangleq \begin{cases} 1, \text{ if } X \subseteq Y \\ 0, \text{ otherwise} \end{cases}, \tag{11}$$

To distinguish the target identity, each target is assigned a unique label $l \in \mathbb{L} = \{\alpha_i | i \in \mathbb{N}\}$. In addition, Vo et al. provide the label RFS [18] where can be written as

$$\mathbf{X}_k^l = \left\{\left(\mathbf{x}_k^1, l_1\right), \cdots, \left(\mathbf{x}_k^{N_k}, l_{N_k}\right)\right\} \in \mathcal{F}(\mathcal{X}) \times \mathbb{L}, \tag{12}$$

where $\mathbf{x}_k^i \in \mathcal{X}, i = 1, \cdots, N_k$ is the single target state, $l_i \in \mathbb{L}$ is a label independent of the target state, $\mathbb{L}$ is the discrete label space. From Eq. (4), $\mathbf{y}_k$ is a super-positional measurement consisting of N_k source informations, then the measurement RFS (without labels) is

$$\mathbf{Y}_k = \{\mathbf{y}_k\} \in \mathcal{F}(\mathcal{Y}), \tag{13}$$

where $\mathcal{F}(\mathcal{Y})$ is a set of all finite subsets of $\mathcal{Y}$ and $\mathcal{Y}$ denotes the measurement space.

Let $\mathcal{L} : \mathcal{F}(\mathcal{X}) \times \mathbb{L} \to \mathbb{L}$ be the projection $\mathcal{L}((\mathbf{x}, l)) = l$, then a finite subset $\mathbf{X}_k^l$ of $\mathcal{L} : \mathcal{F}(\mathcal{X}) \times \mathbb{L}$ owns distinct labels if and only if $\mathbf{X}_k^l$ and its labels $\mathcal{L}\left(\mathbf{X}_k^l\right) = \{\mathcal{L}(\mathbf{x}) : \mathbf{x} \in \mathbf{X}_k^l\}$ have the same cardinality, i.e., $\delta_{|\mathbf{X}|}(|\mathcal{L}(\mathbf{X})|) = 1$, where $|\mathbf{X}|$ represents the cardinality of the set $\mathbf{X}$. The distinct label indicator can be defined as $\Delta(\mathbf{X}) \triangleq \delta_{|\mathbf{X}|}(|\mathcal{L}(\mathbf{X})|)$ [19]. For convenience, we omit the complex representation of the time index k and label index l, and define $\mathbb{L} \triangleq \mathbb{L}_{0:k}$, $\mathbb{B} \triangleq \mathbb{L}_{k+1}$, $\mathbb{L}_+ \triangleq \mathbb{L} \cup \mathbb{B}$, $\boldsymbol{\pi} \triangleq \boldsymbol{\pi}_k$, $\boldsymbol{\pi}_+ \triangleq \boldsymbol{\pi}_{k+1|k}$, $g \triangleq g_k$, $f \triangleq f_{k+1|k}$, $\mathbf{X} \triangleq \mathbf{X}_k$, $\mathbf{Y} \triangleq \mathbf{Y}_k$.

2) δ-GLMB filtering: We use the standard inner product notation $\langle f, g\rangle \triangleq \int f(x)g^H(x)dx$ (for array signal processing) and the multi-sources exponential nation $h^{\mathbf{X}} \triangleq \prod_{\mathbf{x}\in\mathbf{X}} h(\mathbf{x})$, where $h(\mathbf{x})$ is a real-valued function with $h^{\emptyset} = 1$ by convention.

Since the GLMB RFS is closed under Bayesian recursion, the numerical implementation is not easy. According to [18], [19], the δ-GLMB RFS is a particular case of the GLMB RFS with the easy numerical implementation of expressions that are well suited for multi-target tracking. The δ-GLMB RFS posterior probability density can be expressed as

$$\boldsymbol{\pi}(\mathbf{X}) = \Delta(\mathbf{X}) \sum_{(I,\xi)\in\mathcal{F}(\mathbb{L})\times\Xi} \omega^{(I,\xi)}\delta_I(\mathcal{L}(\mathbf{X}))\left[p^{(\xi)}\right]^{\mathbf{X}}, \tag{14}$$

$$\sum_{(I,\xi)\in\mathcal{F}(\mathbb{L})\times\Xi} \omega^{(I,\xi)} = 1, \tag{15}$$

where Ξ is a discrete space. Each pair (I, ξ) represents a history measurement association mapping (also can called hypotheses), $\omega^{(I,\xi)}$ is the hypotheses weight with I the set of labels.

If the multi-target filter density at the current time is δ-GLMB given by Eq. (14), then the multi-target prediction density at the next time is also δ-GLMB, i.e.

$$\boldsymbol{\pi}_+(\mathbf{X}_+) = \Delta(\mathbf{X}_+) \sum_{(I_+,\xi)\in\mathcal{F}(\mathbb{L}_+)\times\Xi} \omega_+^{(I_+,\xi)}\delta_{I_+}[\mathcal{L}(\mathbf{X}_+)]\left[p_+^{(\xi)}\right]^{\mathbf{X}_+}, \tag{16}$$

$$\omega_+^{(I_+,\xi)} = \omega_S^{\xi}(I_+ \cap \mathbb{L})\,\omega_B(I_+ \cap \mathbb{B}), \tag{17}$$

$$\omega_S^{\xi}(L) = \left[\boldsymbol{\eta}_S^{(\xi)}\right]^L \sum_{L\subseteq I} 1_I(L)\left[1-\eta_S^{(\xi)}\right]^{I-L}\omega^{(I,\xi)}, \tag{18}$$

$$\eta_S^{(\xi)}(l) = \int\left\langle P_S(\cdot, l), f(\mathbf{x}|\cdot, l), p^{(\xi)}(\cdot, l)\right\rangle d\mathbf{x}, \tag{19}$$

$$p_+^{(\xi)}(\mathbf{x}, l) = 1_{\mathbb{L}}(l)\,p_{+,S}^{(\xi)}(\mathbf{x}, l) + 1_{\mathbb{B}}(l)\,p_B(\mathbf{x}, l), \tag{20}$$

$$p_{+,S}^{(\xi)}(\mathbf{x}, l) = \frac{\left\langle P_S(\cdot, l), f(\mathbf{x}|\cdot, l), p^{(\xi)}(\cdot, l)\right\rangle}{\eta_S^{(\xi)}(l)}. \tag{21}$$

- $I_+ \in \mathcal{F}(\mathbb{L}_+)$ is a set of the predicted track labels,

$\xi \triangleq (\theta_1, \cdots, \theta_k) \in \Xi \triangleq \Theta_1 \times \Theta_2 \times \cdots \times \Theta_k$ denotes the association mapping history up to time k.

- Each pair $(I_+, \xi) \in \mathcal{F}(\mathbb{L}_+) \times \Xi$ denotes a prediction hypotheses, with probability ${\omega_+}^{(I_+,\xi)}$; $p_+^{(\xi)}(\mathbf{x}, l)$ and $p(\cdot, l)^{\xi}$ are the prediction and update probability densities of the state with label l for history ξ, respectively;
- $\omega_B(I_+ \cap \mathbb{B})$ is the weight of the newborn track labels set and $I_+ \cap \mathbb{B} \neq \emptyset$, $\omega_S^{\xi}(L)$ denotes the weight of the surviving labels set;
- $\mathbb{B}$ is the newborn label space and $p_B(\mathbf{x}, l)$ is the probability density function of newborn source $\mathbf{x}$ with label l. $P_S(\cdot, l)$ is the survival probability and $f(\mathbf{x}|\cdot, l)$ is the transition kinematic density.

If the multi-target prediction density at the current time is δ-GLMB given by Eq. (16), then the multi-target filtering (update) density is also δ-GLMB, i.e.

$$\pi(\mathbf{X}|\mathbf{Y}) \propto \Delta(\mathbf{X}) \sum_{(I,\xi)\in\mathcal{F}(\mathbb{L})\times\Xi} \sum_{\theta\in\Theta(I)} \omega^{(I,\xi,\theta)}(\mathbf{Y}) \times \delta_I(\mathcal{L}(\mathbf{X}))\left[p^{(\xi,\theta)}(\cdot|\mathbf{Y})\right]^{\mathbf{X}}, \tag{22}$$

$$\omega^{(I,\xi,\theta)}(\mathbf{Y}) \propto \left[\mu_{\mathbf{Y}}^{(\xi,\theta)}\right]^{I} \omega^{(I,\xi)}, \tag{23}$$

$$\mu_{\mathbf{Y}}^{(\xi,\theta)}(l) = \left\langle p_+^{(\xi)}(\cdot, l), \psi_{\mathbf{Y}}(\cdot, l; \theta) \right\rangle, \tag{24}$$

$$p^{(\xi,\theta)}(\mathbf{x}, l|\mathbf{Y}) = \frac{p_+^{(\xi)}(\mathbf{x}, l)\, \psi_{\mathbf{Y}}(\mathbf{x}, l; \theta)}{\mu_{\mathbf{Y}}^{(\xi,\theta)}(l)}, \tag{25}$$

$$\psi_{\mathbf{Y}}(\mathbf{x}, l; \theta) = \begin{cases} P_D(\mathbf{x}, l)\, g\left(\mathbf{y}_{\theta(l)}|\mathbf{x}, l\right), & \theta(l) > 0 \\ 1 - P_D(\mathbf{x}, l), & \theta(l) = 0 \end{cases}. \tag{26}$$

- Θ is the measurement association mapping: $\theta : \mathbb{L} \to \{0, 1, \cdots, |\mathbf{Y}|\}$, where $\Theta(I)$ denotes the subset of association mapping with domain I. and θ satisfies $\theta(i) = \theta(j) > 0 \Rightarrow i = j$.
- (I, ξ, θ) is a hypotheses when the track label set I has an association mapping history $\xi \triangleq (\theta_1, \cdots, \theta_{k+1}) \in \Xi \triangleq \Theta_1 \times \Theta_2 \times \cdots \times \Theta_{k+1}$, and $\omega^{(I,\xi,\theta)}$ is the corresponding hypotheses weight. An associative mapping θ describes which trajectory generates which measurement, i.e., trajectory l generates measurement $\mathbf{y}_{\theta(l)} \in \mathbf{Y}$, and assigns the integer 0 to the missed detection trajectory.
- $P_D(\mathbf{x}, l)$ represents the detection probability of state $\mathbf{x}$ and $g\left(\mathbf{y}_{\theta(l)}|\mathbf{x}, l\right)$ is a likelihood function.

III. Recursive Implementation of the δ-GLMB DOA Tracking Filtering

The δ-GLMB can be characterized by the parameter set $\left\{\left(\omega^{(I,\xi)}, p^{(\xi)}\right), (I, \xi) \in \mathcal{F}(\mathbb{L} \times \Xi)\right\}$. From the implementation point of view, it is convenient to consider the δ-GLMB parameter set as an enumeration of all hypotheses and their corresponding (positive) weights and trajectories density $\left\{\left(I^{(h)}, \xi^{(h)}, \omega^{(h)}, p^{(h)}\right)\right\}_{h=1}^{H}$, where $H \in \mathbb{R}^+$ is the number of all hypotheses, $\omega^{(h)} \triangleq \omega^{\left(I^{(h)},\xi^{(h)}\right)}$ and $p^{(h)} \triangleq p^{\left(\xi^{(h)}\right)}$. The hypothesis of the h-th component is denoted as $\left(I^{(h)}, \xi^{(h)}\right)$, while the corresponding weight and trajectory density are $\omega^{(h)}$ and $p^{(h)}(\cdot, l), l \in I^{(h)}$, respectively. Thus, implementing the δ-GLMB filter is equivalent to passing the δ-GLMB parameter set forward with time recursively.

A. δ-GLMB prediction

The prediction density given in Eq. (16) has a compact form, but it is more difficult to be calculated since in Eq. (18) it is necessary to sum over all hypersets of L. [18] gives its equivalent form

$$\pi_+(\mathbf{X}_+) = \Delta(\mathbf{X}_+) \sum_{(I,\xi)\in\mathcal{F}(\mathbb{L})\times\Xi} \omega^{(I,\xi)} \sum_{J\in\mathcal{F}(I)} \left[\boldsymbol{\eta}_S^{(\xi)}\right]^{J} \times \left[1 - \boldsymbol{\eta}_S^{(\xi)}\right]^{I-J} \sum_{L\in\mathcal{F}(\mathbb{B})} \omega_B(L)\delta_{J\cup L}(\mathcal{L}(\mathbf{X}_+)) \left[p_+{}^{(\xi)}\right]^{\mathbf{X}_+}, \tag{27}$$

where $J \subseteq I, L \subseteq \mathbb{B}$. Next, this subsection gives a detailed implementation of δ-GLMB prediction, which uses the K-shortest-path algorithm [29] to prune the predicted δ-GLMB parameter set components without computing all the prediction hypotheses and their weights.

1) Compute the predicted parameter sets (Sequential Monte Carlo (SMC) implementation): For the SMC approximation, assuming that each single target density $p^{(\xi)}(\cdot, l)$ can be represented by a weighted sample set $\left\{\omega_i^{(\xi)}(l), \mathbf{x}_i^{(\xi)}(l)\right\}_{i=1}^{N^{(\xi)}(l)}$ and that the newborn density $p_B^{(l)}(\cdot)$ can be represented by $\left\{\omega_i^{(\xi)}(l), \mathbf{x}_i^{(\xi)}(l)\right\}_{i=1}^{N_B^{(\xi)}(l)}$, we have

$$\boldsymbol{\eta}_S^{(\xi)}(l) = \sum\nolimits_{i=1}^{N^{(\xi)}(l)} \omega_i^{(\xi)}(l)\, P_S\left(\mathbf{x}_i^{(\xi)}(l), l\right), \tag{28}$$

and $p_+^{(\xi)}(\mathbf{x}, l)$ can be denoted as

$$\left\{1_{\mathbb{L}}(l)\, \hat{\omega}_{S,i}^{(\xi)}(l), \mathbf{x}_{S,i}^{(\xi)}(l)\right\}_{i=1}^{N^{(\xi)}(l)} \cup \left\{1_{\mathbb{L}}(l)\, \omega_{B,i}^{(\xi)}(l), \mathbf{x}_{B,i}^{(\xi)}(l)\right\}_{i=1}^{N_B^{(\xi)}(l)} \tag{29}$$

$$\mathbf{x}_{S,i}^{(\xi)}(l) \sim q\left(\cdot \left| \mathbf{x}_i^{(\xi)}(l), l, \mathbf{y}\right.\right), i = 1, \cdots, N^{(\xi)}(l), \tag{30}$$

$$\omega_{S,i}^{(\xi)}(l) = \frac{\omega_i^{(\xi)}(l)\, f\left(\mathbf{x}_{S,i}^{(\xi)}(l) \left| \mathbf{x}_i^{(\xi)}(l)\right.\right) P_S\left(\mathbf{x}_i^{(\xi)}(l), l\right)}{q\left(\mathbf{x}_{S,i}^{(\xi)}(l) \left| \mathbf{x}_i^{(\xi)}(l), l, \mathbf{y}\right.\right)}, \tag{31}$$

$$\hat{\omega}_{S,i}^{(\xi)}(l) = \omega_{S,i}^{(\xi)}(l) \Big/ \sum\nolimits_{i=1}^{N^{(\xi)}(l)} \omega_{S,i}^{(\xi)}(l), \tag{32}$$

where $q\left(\cdot \left| \mathbf{x}_i^{(\xi)}(l), l, \mathbf{y}\right.\right)$ is the suggested density.

2) Pruning the prediction density: Given the enumerated parameter set $\left\{\left(I^{(h)}, \xi^{(h)}, \omega^{(h)}, p^{(h)}\right)\right\}_{h=1}^{H}$ with δ-GLMB filter density, (16) can be rewritten as

$$\boldsymbol{\pi}_+(\mathbf{X}_+) = \sum\nolimits_{h=1}^{H} \boldsymbol{\pi}_+^{(h)}(\mathbf{X}_+), \tag{33}$$

$$\boldsymbol{\pi}_+^{(h)}(\mathbf{X}_+) = \Delta(\mathbf{X}_+) \sum_{J\in I^{(h)}} \sum_{L\in\mathbb{B}} \omega_S^{\left(I^{(h)},\xi^{(h)}\right)}(J) \times \omega_B(L)\, \delta_{J\cup L}(\mathcal{L}(\mathbf{X}_+)) \left[p_+{}^{\left(\xi^{(h)}\right)}\right]^{\mathbf{X}_+}. \tag{34}$$

From [19], for δ-GLMB prediction, the h-th component generates $2^{|I^{(h)}|+|\mathbb{B}|}$ components. The K-shortest path prob-

TABLE I: Pseudo-code of δ-GLMB prediction

1: Input: $\left\{\left(I^{(h)},\xi^{(h)},\omega^{(h)},p^{(h)},K^{(h)}\right)\right\}_{h=1}^{H}$, K_B, $\{(\varsigma_B^{(l)},p_B^{(l)})\}_{l\in\mathbb{B}}$

2: Compute the cost function vector: $\mathbf{c}_B=[c_b(l_1),\cdots,c_b(l_{|\mathbb{B}|})]$, where $c_b(l_i)=-ln[\varsigma_B^{l_i}/(1-\varsigma_B^{l_i})]$, $i=1,\cdots,|\mathbb{B}|$.

3: $\{L^{(b)}\}_{b=1}^{K_B}=K$-shortest-path($\mathbb{B},\mathbf{c}_B,K_B$).

4: Calculate $\omega_B^{(b)}=\prod_{l\in\mathbb{B}-L^{(b)}}\left(1-\varsigma_B^{(l)}\right)\prod_{l\in L^{(b)}}\varsigma_B^{(l)}$, for $b=1,\cdots,K_B$.

5: For $h=1:H$

Calculate $\boldsymbol{\eta}_S^{(h)}:=\boldsymbol{\eta}_S^{\xi^{(h)}}$ according to Eq. (28).

Calculate cost vector: $\mathbf{c}_S^{(h)}:=\mathbf{c}_S^{(I^{(h)},\xi^{(h)})}=[c_S(l_1),\cdots,c_S(l_{|I^{(h)}|})]$, where $c_S(l_j)=-ln[\boldsymbol{\eta}_S^{(h)}(l_j)/(1-\boldsymbol{\eta}_S^{(h)}(l_j))]$, $j=1,\cdots,|I^{(h)}|$.

$\{J^{(h,j)}\}_{j=1}^{K^{(h)}}=K$-shortest-path($I^{(h)},\mathbf{c}_S^{(h)},K^{(h)}$).

6: For $(j,b)=(1,1):(K^{(h)},K_B)$

$\omega_+^{(h,j,b)}:=\omega_S^{\left(I^{(h)},\xi^{(h)}\right)}\left(J^{(h,j)}\right)\omega_B^{(b)}$

$\omega_S^{\left(I^{(h)},\xi^{(h)}\right)}\left(J^{(h,j)}\right)=\omega^{(h)}\left[\boldsymbol{\eta}_S^{(h)}\right]^{J^{(h,j)}}\left[1-\boldsymbol{\eta}_S^{(h)}\right]^{I^{(h)}-J^{(h,j)}}$

$I_+^{(h,j,b)}:=J^{(h,j)}\cup L^{(b)}$

7: End

Calculate $p_+^{(h)}:=p_+^{(\xi^{(h)})}$ according to Eq. (29).

8: End

9: Normalize weight $\left\{\omega_+^{(h,j,b)}\right\}_{(h,j,b)=(1,1,1)}^{\left(H,K^{(h)},K_B\right)}$.

8: Output: $\left\{I_+^{(h,j,b)},\omega_+^{(h,j,b)},p_+^{(h)}\right\}_{(h,j,b)=(1,1,1)}^{\left(H,K^{(h)},K_B\right)}$.

lem attempts to find a subset of the K shortest distances of I in non-decreasing order. Given the enumerated parameter set $\left\{\left(I^{(h)},\xi^{(h)},\omega^{(h)},p^{(h)}\right)\right\}_{h=1}^{H}$ with δ-GLMB filter density, the K-shortest path algorithm is used to determine $K^{(h)}$ subsets with maximum weights $\omega_S^{(h,1)}\geqslant\omega_S^{(h,2)}\geqslant\cdots\geqslant\omega_S^{\left(h,K^{(h)}\right)}$.

For the newborn targets, the labeled multi-Bernoulli newborn model is utilized, i.e.

$$\omega_B(L)=\prod_{l\in\mathbb{B}}\left(1-\varsigma_B^{(l)}\right)\prod_{l\in L}\frac{1_{\mathbb{B}}(l)\,\varsigma_B^{(l)}}{\left(1-\varsigma_B^{(l)}\right)}, \tag{35}$$

$$p_B(\mathbf{x},l):=p_B^{(l)}(\mathbf{x}), \tag{36}$$

where $\varsigma_B^{(l)}$ is the existence probability with newborn label l and $p_B^{(l)}(\mathbf{x})$ denotes the newborn probability density function. Similarly, K_B newborn subsets with the highest newborn weights can be obtained, and the specific K-shortest path algorithm operation can be found in [19]. Then, for each h, the pruning version of $\boldsymbol{\pi}_+^{(h)}$ can be expressed as

$$\tilde{\boldsymbol{\pi}}_+^{(h)}(\mathbf{X}_+)=\Delta(\mathbf{X}_+)\sum_{j=1}^{K^{(h)}}\sum_{b=1}^{K_B}\sum_{J^{(h,j)}\in I^{(h)}}\sum_{L^{(b)}\in\mathbb{B}}\omega_B\left(L^{(b)}\right)$$
$$\times\omega_S^{\left(I^{(h)},\xi^{(h)}\right)}\left(J^{(h,j)}\right)\delta_{J^{(h,j)}\cup L^{(b)}}\left(\mathcal{L}(\mathbf{X}_+)\right)\left[p_+{}^{(\xi^{(h)})}\right]^{\mathbf{X}_+}. \tag{37}$$

The specific values of the number of required components $K^{(h)}$ and K_B are specified by the user or related to the specific application. A general strategy is to choose $K^{(h)}=\lceil\omega^{(h)}J_{\max}\rceil$, where J_{max} is the total number of expected hypotheses. Moreover, K_B can be chosen such that the resulting pruning captures the probabilistic quality of the desired proportion of the newborn density. The pseudo-code of δ-GLMB prediction algorithm is given in TABLE I.

B. δ-GLMB update

The component pruning operation is also essential in the δ-GLMB update process. This section presents a detailed implementation of the δ-GLMB update, which prunes the multi-objective filter density by a ranked assignment algorithm [19], and also without exhaustively computing all hypotheses and their weights. First, the ranked assignment problem in the context of pruning the δ-GLMB filter density is introduced.

1) Ranked assignment problem: From Eq. (23), each hypothesis (I,ξ) with weight $\omega^{(I,\xi)}$ generates a new set of hypotheses $(I,(\xi,\theta)),\theta\in\Theta(I)$ with weight $\omega^{(I,\xi,\theta)}(\mathbf{Y})\propto\left[\boldsymbol{\mu}_{\mathbf{Y}}^{(\xi,\theta)}\right]^{I}\omega^{(I,\xi)}$. For a given hypothesis (I,ξ), the component with the maximum weight can be selected without exhaustively computing all hypotheses and their weights if an associative mapping $\theta\in\Theta(I)$ in descending order of $\left[\boldsymbol{\mu}_{\mathbf{Y}}^{(\xi,\theta)}\right]^{I}$ can be generated. The solution of the following ranked assignment problem proposed in [19] can achieve such a requirement.

Construct the $|I|\times|\mathbf{Y}|$-dimensional cost matrix $\mathbf{C}_{\mathbf{Y}}^{(I,\xi)}$ as

$$\mathbf{C}_{\mathbf{Y}}^{(I,\xi)}=\begin{bmatrix}c_{1,1}&\cdots&c_{1,|\mathbf{Y}|}\\ \vdots&&\vdots\\ c_{|I|,1}&\cdots&c_{|I|,|\mathbf{Y}|}\end{bmatrix}, \tag{38}$$

with

$$c_{i,j}=-\ln\left[\frac{\left\langle p_+^{(\xi)}(\cdot,l_i),g(\mathbf{y}_j|\cdot,l_i)\right\rangle}{\left\langle p_+^{(\xi)}(\cdot,l_i),1-p_D(\cdot,l_i)\right\rangle}\right] \tag{39}$$

the cost of assigning the $j\in\{1,2,\cdots,|\mathbf{Y}|\}$-th measurement to track $l_i,i\in\{1,2,\cdots,|I|\}$.

Note from the array model (4) that the measurement is super-positional information (i.e., $|\mathbf{Y}|=1$), therefore the cost matrix $\mathbf{C}$ becomes a vector with dimension $|I|\times1$, which reduces the algorithm tracking performance. This allows a single measurement to be assigned to multiple trajectories at the same time, which leads to low pruning efficiency. An alternative solution is the NMAP strategy proposed in [22], denoted as follows

$$\mathbf{\Gamma}_n=\mathbf{U}_{S,n}\tau_n\mathbf{U}_{S,n}^H,n=1,2,\cdots,\tilde{N}_k, \tag{40}$$

where $\tilde{N}_k$ is the estimated number of sources by MDL method [30], $\mathbf{U}_{S,n}$ denotes the n-th column of matrix $\mathbf{U}_S$, and $\mathbf{U}_S$ can be obtained by performing eigenvalue decomposition (EVD) of the spatial smoothing covariance matrix (9)

$$\bar{\mathbf{R}}_{ss}=\mathbf{U}\Lambda\mathbf{U}^H=\sum_{i=1}^{G}\tau_i\mathbf{u}_i\mathbf{u}_i^H=\mathbf{U}_s\mathbf{\Sigma}_s\mathbf{U}_s^H+\mathbf{U}_N\mathbf{\Sigma}_N\mathbf{U}_N^H, \tag{41}$$

where $\mathbf{\Lambda}=diag\{\tau_1,\tau_2,\cdots,\tau_G\}$ is the eigenvalues matrix, and $\tau_1\geqslant\tau_2\geqslant\cdots\geqslant\tau_{\tilde{N}_k}\geqslant\tau_{\tilde{N}_k+1}\geqslant\cdots\geqslant\tau_G$. $\mathbf{\Sigma}_s$ is a diagonal matrix consisting of $\tilde{N}_k$ largest eigenvalues, whereas $\mathbf{\Sigma}_N$ is the diagonal matrix composed of the remaining eigenvalues. $\mathbf{U}_s=\left[\mathbf{u}_1,\mathbf{u}_2,\cdots,\mathbf{u}_{\tilde{N}_k}\right]$ is the signal subspace, which is the eigenvectors matrix corresponding to the $\tilde{N}_k$ largest eigenvalues. $\mathbf{U}_N=\left[\mathbf{u}_{\tilde{N}_k+1},\cdots,\mathbf{u}_G\right]$ is the noise subspace, which is the eigenvectors matrix corresponding to the remaining $G-\tilde{N}_k$ smallest eigenvalues.

Remark 2: According to (40)-(41), we can get $\tilde{N}_k$ reconstructed matrices $\mathbf{\Gamma}_n, n=1,\cdots,\tilde{N}_k$. Therefore the measurement RFS (13) can be rewritten as

$$\mathbf{Y}_k=\left\{\mathbf{\Gamma}_1,\mathbf{\Gamma}_2,\cdots,\mathbf{\Gamma}_{\tilde{N}_k}\right\}\in\mathcal{F}\left(\mathcal{Y}\right), \tag{42}$$

then, the dimension of the cost matrix (38) becomes $|I|\times\tilde{N}_k$, and its computation is detailed in the next section.

2) Calculate the updated parameter sets: Assuming that each single target density $p^{(\xi)}(\cdot,l)$ can be represented by a weighted sample set $\left\{\omega_n^{(\xi)}\left(l\right),\mathbf{x}_n^{(\xi)}\left(l\right)\right\}_{n=1}^{N^{(\xi)}(l)}$, then we have

$$c_{ij}=-\ln\left[\frac{\sum_{n=1}^{J^{(\xi)}(l_i)}\omega_n^{(\xi)}\left(l_i\right)P_D\left(\mathbf{x}_n^{(\xi)}\left(l_i\right),l_i\right)g\left(\mathbf{y}_j|\mathbf{x}_n^{(\xi)}\left(l_i\right),l_i\right)}{\sum_{n=1}^{J^{(\xi)}(l_i)}\omega_n^{(\xi)}\left(l_i\right)\left(1-P_D\left(\mathbf{x}_n^{(\xi)}\left(l_i\right),l_i\right)\right)}\right], i=1,\cdots,|I|,j=1,\cdots,\tilde{N}_k, \tag{43}$$

and for each give association history (ξ,θ),

$$\mu_{\mathbf{Y}}^{(\xi,\theta)}\left(l\right)=\sum\nolimits_{n=1}^{J^{(\xi)}(l)}\omega_n^{(\xi)}\left(l_i\right)\psi_{\mathbf{Y}}\left(\mathbf{x}_n^{(\xi)}\left(l\right),l;\theta\right), \tag{44}$$

$$\omega_n^{(\xi,\theta)}\left(l\right)=\frac{\omega_n^{(\xi)}\left(l\right)\psi_{\mathbf{Y}}\left(\mathbf{x}_n^{(\xi)}\left(l\right),l;\theta\right)}{\mu_{\mathbf{Y}}^{(\xi,\theta)}\left(l\right)}, \tag{45}$$

and $p^{(\xi,\theta)}(\cdot,l|\mathbf{Y})$ can be denoted as the following weighted sample sets

$$\left\{\omega_n^{(\xi,\theta)}\left(l\right),x_n^{(\xi)}\left(l\right)\right\}_{n=1}^{J^{(\xi)}(l)}. \tag{46}$$

3) Calculation scheme for the likelihood function: By performing EVD again to the measurement component $\mathbf{\Gamma}_n, n=1.\cdots,\tilde{N}_k$, we have

$$\mathbf{\Gamma}_n=\mathbf{U}_n\mathbf{\Sigma}_n\mathbf{U}_n^H=\tilde{\mathbf{U}}_{n,1}\mathbf{\Phi}_1\tilde{\mathbf{U}}_{n,1}^H+\tilde{\mathbf{U}}_{n,G-1}\mathbf{\Phi}_{G-1}\tilde{\mathbf{U}}_{n,G-1}^H, \tag{47}$$

where $\mathbf{\Sigma}_n=diag\left\{\varepsilon_1,\varepsilon_2,\cdots,\varepsilon_G\right\}$ is the diagonal matrix and $\mathbf{\Phi}_1=\varepsilon_1$ denotes the largest eigenvalue. $\mathbf{\Phi}_{G-1}$ is a matrix consisting of the remaining eigenvalues. $\widetilde{\mathbf{U}}_{n,1}\in\mathbb{C}^{G\times1}$ and $\widetilde{\mathbf{U}}_{n,G-1}\in\mathbb{C}^{G\times(G-1)}$ stand for the n-th signal eigenvector and remaining eigenvectors, respectively.

Then, the likelihood function can be replaced by the following MUSIC pseudo-spectrum

$$g\left(\mathbf{y}_j|\mathbf{x}_n^{(\xi)}\left(l_i\right),l_i\right)=P_{\text{MUSIC}}\left(\mathbf{x}_n^{(\xi)}\left(l_i\right)\right)=\left|\frac{1}{\mathbf{a}\left(\mathbf{cx}_n^{(\xi)}\left(l_i\right)\right)^H\tilde{\mathbf{U}}_{j,G-1}\tilde{\mathbf{U}}_{j,G-1}^H\mathbf{a}\left(\mathbf{cx}_n^{(\xi)}\left(l_i\right)\right)}\right|^{\zeta}, \tag{48}$$
$$i=1,\cdots,|I|,j=1,\cdots,\tilde{N}_k,n=1,\cdots,J^{(\xi)(l_i)},$$

where $\mathbf{c}=[1,0]$ such that $\mathbf{cx}_n^{(\xi)}\left(l_i\right)$ denotes the DOA information, and $\zeta\in\mathbf{R}^+$ is the exponential weighting factor.

Remark 3: Fig. 2 shows the exponential weighting of the MUSIC spatial spectral function (also called likelihood function. For convenience, we do not adopt the NMAP strategy here, and the noise subspace of the MSUIC spectral function can be obtained by Eq. (41)) for 2 sources (DOAs are $-40°$ and $50°$, respectively.), where SNR = 10 dB, $T_k=200$, and the number of sensors $M=4$, $N=5$. It can be seen from Fig. 2 that the likelihood function can be increased by increasing exponential weighting factor ζ, and the beam width of the MUSIC spectrum becomes narrower. However, if ζ is very large, the diversity of particles will be lost, which leads to the degradation of the algorithm tracking performance, i.e., a suitable ζ should be chosen to make the algorithm performance optimal. The specific value of ζ will be discussed in the simulation.

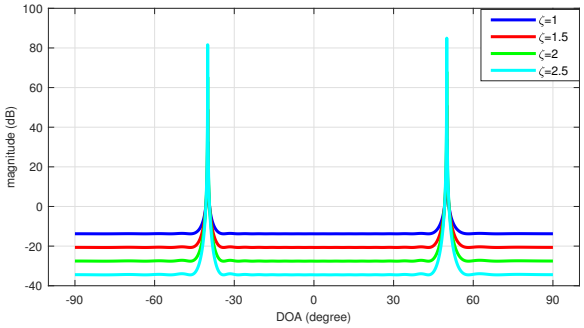


Fig. 2: The likelihood function with different ζ.

4) Pruning update density: Given the enumerated parameter set $\left\{\left(I^{(h)},\xi^{(h)},\omega^{(h)},p^{(h)}\right)\right\}_{h=1}^{H}$ with δ-GLMB prediction density, (22) can be rewritten as

$$\boldsymbol{\pi}\left(\mathbf{X}|\mathbf{Y}\right)=\sum\nolimits_{h=1}^{H}\boldsymbol{\pi}^{(h)}\left(\mathbf{X}|\mathbf{Y}\right), \tag{49}$$

$$\boldsymbol{\pi}^{(h)}\left(\mathbf{X}|\mathbf{Y}\right)=\Delta\left(\mathbf{X}\right)\sum_{j=1}^{\left|\Theta\left(I^{(h)}\right)\right|}\omega^{(h,j)}\delta_{I^{(h)}}\left(\mathcal{L}\left(\mathbf{X}\right)\right)\left[p^{(h,j)}\right]^{\mathbf{X}}, \tag{50}$$

$$\omega^{(h,j)}\triangleq\omega^{\left(I^{(h)},\xi^{(h)},\theta^{(h,j)}\right)}\left(\mathbf{Y}\right), \tag{51}$$

$$p^{(h,j)}\triangleq p^{\left(I^{(h)},\xi^{(h)},\theta^{(h,j)}\right)}\left(\cdot\,|\mathbf{Y}\right). \tag{52}$$

Each δ-GLMB prediction component with index h yields $\left|\Theta\left(I^{(h)}\right)\right|$ δ-GLMB update density components. According to [19], a simple and efficient method for pruning the δ-GLMB update density Eq. (49) is to prune $\boldsymbol{\pi}^{(h)}\left(\cdot|\mathbf{Y}\right)$. For each prediction component with index h, solving the ranked optimal assignment problem for the cost matrix $\mathbf{C}_{\mathbf{Y}}^{\left(I^{(h)},\xi^{(h)}\right)}$ will produce $T^{(h)}$ hypotheses $\theta_{(h,j)},j=1,2,\cdots,T^{(h)}$ with the highest weight in non-increasing order. The prediction component h produces a large number of δ-GLMB update density components, and the ranked assignment algorithm determines $T^{(h)}$ components with the maximum weights $\omega^{(h,1)}\geqslant\omega^{(h,2)}\geqslant\cdots\geqslant\omega^{\left(h,T^{(h)}\right)}$. Thus, the pruned version of $\boldsymbol{\pi}^{(h)}\left(\cdot|\mathbf{Y}\right)$ can be expressed as

$$\tilde{\boldsymbol{\pi}}^{(h)}\left(\mathbf{X}|\mathbf{Y}\right)=\Delta\left(\mathbf{X}\right)\sum_{j=1}^{T^{(h)}}\omega^{(h,j)}\delta_{I^{(h)}}\left(\mathcal{L}\left(\mathbf{X}\right)\right)\left[p^{(h,j)}\right]^{\mathbf{X}}. \tag{53}$$

The value of the required components $T^{(h)}$ is specified by the user or related to the specific application. A general strategy is to choose $T^{(h)}=\left\lceil\omega^{(h)}J_{\max}\right\rceil$, where J_{max} is the total

TABLE II: Pseudo-code of δ-GLMB update

Input: $\left\{\left(I^{(h)}, \xi^{(h)}, \omega^{(h)}, p^{(h)}, T^{(h)}\right)\right\}_{h=1}^{H}$, $\mathbf{Y}$
For $h = 1 : H$
  Calculate cost matrix $\mathbf{C}_{\mathbf{Y}}^{(h)} := \mathbf{C}_{\mathbf{Y}}^{\left(I^{(h)}, \xi^{(h)}\right)}$ according to Eq. (38) and (43).
  $\{\theta^{(h,j)}\}_{j=1}^{T^{(h)}}$ = ranked-assignment $(\mathbf{Y}, I^{(h)}, \mathbf{C}_{\mathbf{Y}}^{(h)}, T^{(h)})$.
  For $j = 1 : T^{(h)}$
    Calculate $\boldsymbol{\mu}_{\mathbf{Y}}^{(h,j)} := \boldsymbol{\mu}_{\mathbf{Y}}^{\left(\xi^{(h)}, \theta^{(h,j)}\right)}$ according to Eq. (43).
    Calculate $p^{(h,j)} := p^{\left(\xi^{(h)}, \theta^{(h,j)}\right)}(\cdot\,|\mathbf{Y})$ based on Eq. (44) and (46).
    $\omega^{(h,j)} := \left[\mu_{\mathbf{Y}}^{(h,j)}\right]^{I^{(h)}} \omega^{(h)}$ according to Eq. (23).
    $I^{(h,j)} := I^{(h)}$.
    $\xi^{(h,j)} := \left(\xi^{(h)}, \theta^{(h,j)}\right)$.
  End
End
Normalize weights $\left\{\omega^{(h,j)}\right\}_{(h,j)=(1,1)}^{\left(H, T^{(h)}\right)}$.
Output: $\left\{I^{(h,j)}, \xi^{(h,j)}, \omega^{(h,j,)}, p^{(h,j)}\right\}_{(h,j)=(1,1)}^{\left(H, T^{(h)}\right)}$.

number of expected hypotheses. The pruned update density has a total number of components of $T = \sum_{h=1}^{H} T^{(h)}$, and the pruned δ-GLMB update density is obtained by normalizing the sum of the weights. TABLE II gives the pseudo-code of δ-GLMB update algorithm.

IV. Proposed Modified δ-GLMB Filtering

In SMC implementation, the resampling is considered as a way to reduce the particle degeneracy problem. However, it has been pointed out that resampling causes additional issues, particularly the loss of particle diversity. The regularization-based step (a kind of jittering of particles) proposed in [31] is called regularized particle filtering (RPF). The tracking performance of the δ-GLMB algorithm is further improved by regularizing to increase the diversity of predicted particles. In this section, we first introduce the regularized PF, and then apply it to the field of array signal processing, and finally give a concrete implementation of the regularized δ-GLMB filtering (called modified δ-GLMB).

A. Regularized PF

In fact, the regularization step is a jittering operation on the particles that yields particles that are closer to the posterior probability density. Suppose there is a weighted particle set $\left\{\mathbf{x}_k^i, \omega_k^i\right\}_{i=1}^{N}$, where N denotes the total number of particles. Then the corrected set of particles is as follows

$$\mathbf{x}_k^{i^*} = \mathbf{x}_k^i + h_{opt}\mathbf{D}_k\varsigma^i, i = 1, \cdots, N \tag{54}$$

where

- $h_{opt} = A \cdot N^{-1/(n_x+4)}$ is the optimal kernel bandwidth [31] and $A = [4/(n_x+2)]^{1/(n_x+4)}$ is a constant.
- n_x is the dimension of state vector $\mathbf{x}_k^i$.
- $\mathbf{D}_k$ such that $\mathbf{D}_k\mathbf{D}_k^T = \mathbf{S}_k$, where $\mathbf{S}_k$ is the empirical covariance matrix of particle set $\left\{\mathbf{x}_k^i, \omega_k^i\right\}_{i=1}^{N}$.

The description of the regularized PF algorithm for one cycle is given in TABLE III. Notice that the array signal processing method is based on the complex domain, so the computational scheme of the likelihood function $g\left(\mathbf{y}_k|\mathbf{x}_k^i\right)$ no longer follows Eq. (48). We utilize the least squares method

TABLE III: Regularized particle filtering

$\left\{\mathbf{x}_k^{i^*}, \omega_k^i\right\}_{i=1}^{N} = \text{RPF}\left\{\left\{\mathbf{x}_{k-1}^i, \omega_{k-1}^i\right\}_{i=1}^{N}, \mathbf{y}_k\right\}$
For $i = 1 : N$
  Draw $\mathbf{x}_k^i \sim f\left(\mathbf{x}_k|\mathbf{x}_{k-1}^i\right)$ according to Eq. (1) and Calculate $\tilde{\omega}_k^i = g\left(\mathbf{y}_k|\mathbf{x}_k^i\right)$.
End
Normalize the weights: $\omega_k^i = \tilde{\omega}_k^i \Big/ \sum_{i=1}^{N} \tilde{\omega}_k^i$.
Calculate $\mathbf{D}_k$ such that $\mathbf{D}_k\mathbf{D}_k^T = \mathbf{S}_k$.
Resample the particle set: $\left\{\mathbf{x}_k^i, \omega_k^i, -\right\}_{i=1}^{N} = \text{resample}\left\{\left\{\mathbf{x}_k^i, \omega_k^i\right\}_{i=1}^{N}\right\}$.
For: $i = 1 : N$
  Draw $\varsigma^i \sim \mathcal{N}(0, 1)$ from the Gaussian kernel.
  $\mathbf{x}_k^{i^*} = \mathbf{x}_k^i + h_{opt}\mathbf{D}_k\varsigma^i$.
End

to obtain space domain filtering $\mathbf{S}$, and a Gaussian function is applied for likelihood function. The regularized PF algorithm based on array signal processing (called RPF-SP) is given in TABLE IV.

TABLE IV: Regularized particle filtering in the presence of array signal processing

$\left\{\mathbf{x}_k^{i^*}, \omega_k^i\right\}_{i=1}^{N} = \text{RPF-SP}\left\{\left\{\mathbf{x}_{k-1}^i, \omega_{k-1}^i\right\}_{i=1}^{N}\right\}$
For $i = 1 : N$
  Draw $\mathbf{x}_k^i \sim f\left(\mathbf{x}_k|\mathbf{x}_{k-1}^i\right)$ according to Eq. (1).
  Calculate $\tilde{\omega}_k^i = g\left(\mathbf{y}_k|\mathbf{x}_k^i\right)$:
    Generate the observation $\mathbf{y}_k^i$ for $\mathbf{x}_k^i$ according to Eq. (4).
    Calculate the direction vector $\mathbf{a}(\mathbf{cx}_k^i)$ based on Eq. (5).
    Calculate the space domain filter $\mathbf{S} = \mathbf{a}^+\mathbf{y}_k^i$, where $\mathbf{a}^+ = \left(\mathbf{a}^H\mathbf{a}\right)^{-1}\mathbf{a}^H$.
    Calculate weight $\tilde{\omega}_k^i = \exp\left[-\left\|\mathbf{y}_k^i - \mathbf{aS}\right\|^2\right]$.
End
Normalize the weights: $\omega_k^i = \tilde{\omega}_k^i \Big/ \sum_{i=1}^{N} \tilde{\omega}_k^i$.
Calculate the mean value: $r_k^{mean} = \sum_{i=1}^{N} \mathbf{cx}_k^i\omega_k^i$.
Calculate the empirical covariance matrix: $\mathbf{S}_k = \sum_{i=1}^{N} \omega_k^i\left(\mathbf{cx}_k^i - r_k^{mean}\right)^2$.
Calculate $\mathbf{D}_k$ such that $\mathbf{D}_k\mathbf{D}_k^T = \mathbf{S}_k$:
Resample the particle set:
  $\left\{\mathbf{x}_k^i, -\right\}_{i=1}^{N} = \text{resample}\left\{\left\{\mathbf{x}_k^i, \omega_k^i\right\}_{i=1}^{N}\right\}$.
For: $i = 1 : N$
  Draw $\varsigma^i \sim \mathcal{N}(0, 1)$ from the Gaussian kernel.
  $A = [4/(n_x+2)]^{1/(n_x+4)}$ with $n_x = 1$.
  $h_{opt} = A \cdot N^{-1/(n_x+4)}$.
  $\mathbf{x}_k^{i^*} = \mathbf{x}_k^i + h_{opt}\mathbf{D}_k\varsigma^i$.
End

B. Modified δ-GLMB filtering

In this section, the RPF-SP algorithm is used to manipulate the predicted particles of the δ-GLMB SMC implementation, in fact, (30) and (31) can be rewritten as follows

$$\mathbf{x}_{S,i}^{(\xi)}(l) \sim f\left(\cdot\,\middle|\mathbf{x}_i^{(\xi)}(l), l\right), i = 1, \cdots, N^{(\xi)}(l), \tag{55}$$

$$\omega_{S,i}^{(\xi)}(l) = \omega_{i^*}^{(\xi)}(l) P_S\left(\mathbf{x}_i^{(\xi)}(l), l\right) \tag{56}$$

with

$$\begin{aligned}&\left\{\mathbf{x}_{S^*,i}^{(\xi)}(l), \omega_{i^*}^{(\xi)}(l)\right\}_{i=1}^{N^{(\xi)}(l)} \\ &= \text{RPF-SP}\left\{\left\{\omega_i^{(\xi)}(l), \mathbf{x}_{S,i}^{(\xi)}(l)\right\}_{i=1}^{N^{(\xi)}(l)}\right\},\end{aligned} \tag{57}$$

then, (29) can be expressed as

$$\left\{1_{\mathbb{L}}(l)\omega_{S,i}^{(\xi)}(l), \mathbf{x}_{S^*,i}^{(\xi)}(l)\right\}_{i=1}^{N^{(\xi)}(l)} \cup \left\{1_{\mathbb{L}}(l)\omega_{B,i}^{(\xi)}(l), \mathbf{x}_{B,i}^{(\xi)}(l)\right\}_{i=1}^{N_B^{(\xi)}(l)}. \tag{58}$$

The detailed SMC implementation of the modified δ-GLMB algorithm is given in TABLE V, integrating the contents of

TABLE I, II and IV. In TABLE V, the multi-target state estimation is given in step 4. For the GLMB update density, a simple and intuitive multi-target estimator is the multi-Bernoulli estimator [19], which selects the set of trajectories or labels whose probability of existence (the probability of existence of track l is the sum of the weights of all hypotheses containing track l, i.e., $\sum_{(I,\xi)\in\mathcal{F}(\mathbb{L})\times\Xi}\omega^{(I,\xi)}1_I(l)$) is above some threshold, and then estimates the track state based on the maximum a posteriori or expectation posterior of the probability density $p^{(\xi)}(\cdot,l), l\in L$.

TABLE V: Pseudo-code of δ-GLMB update

1: For $k=1:K$
2: δ-GLMB prediction see TABLE I and IV.
3: δ-GLMB update see TABLE II.
4: Multi-target state estimation
Input: $N_{max}, \left\{I^{(h,j)}, \xi^{(h,j)}, \omega^{(h,j,)}, p^{(h,j)}\right\}_{(h,j)=(1,1)}^{\left(H,T^{(h)}\right)}$.
Calculate the cardinality distribution:
$\rho(n)=\sum_{h=1}^{H}\sum_{j=1}^{T^{(h)}}\omega^{(h,j)}\delta_n\left(\left|I^{(h,j)}\right|\right), n=1,2,\cdots,N_{\max}$.
Find the largest cardinality:
$\tilde{N}=\text{argmax}(\rho)$.
For $n=1:\tilde{N}$
$\left(\tilde{h},\tilde{j}\right)=\arg\max_{(h,j)}\omega^{(h,j)}\delta_n\left(\left|I^{(h,j)}\right|\right)$.
$\mathbf{X}_k=\left\{(\tilde{\mathbf{x}},l): l\in I^{(\tilde{h},\tilde{j})}, |\tilde{\mathbf{x}}=\int\mathbf{x}p^{(\tilde{h},\tilde{j})}(\mathbf{x},l)\,d\mathbf{x}\right\}$.
End
5: End

V. SIMULATION

The tracking performance of the proposed modified δ-GLMB filtering is verified by three simulation scenarios (a single source tracking scenario, multi-source with fixed number of sources and multi-source with time-varying number of sources). We also give some simulation examples to illustrate the performance of various methods-the proposed modified δ-GLMB filtering, the GLMB filtering based on ULA (GLMB-ULA, [22]), the PHD filtering with extended coprime array (PHD-ECA, [23]), the spatially smoothed PAST (SS-PAST, [5]) tracking method with ECA, particle filtering with ECA (PF-ECA, [6]). However, particle filtering can only handle a single source scenario, and the SS-PAST method is prone to singular values in a single source scenario. Therefore, the SS-PAST method is no longer considered for comparison in the single source scenario, and similarly, the PF-ECA method is not considered for comparison in the multi-source scenario.

A. CRB of DOA tracking

The performance evaluation of the proposed DOA tracking algorithm can be assessed with the stochastic CRB [32], which is the inverse of the Fisher information matrix (FIM). However, the conventional FIM is singular when the number of sources exceeds the number of physical sensors. To solve this problem, the Fisher information matrix is converted to a virtual array-based form in [33]–[35], and can be expressed as

$$\mathbf{FIM}=T_k\left[vec\left(\frac{\partial\mathbf{R}_x}{\partial\boldsymbol{\xi}}\right)\right]^H\left(\mathbf{R}_x^T\otimes\mathbf{R}_x\right)^{-1}\left[vec\left(\frac{\partial\mathbf{R}_x}{\partial\boldsymbol{\xi}}\right)\right], \tag{59}$$

which remains non-singular in a wider range of conditions, where T_k denotes the number of snapshots at time k. Thus, it overcomes the model mismatch problem of the stochastic CRB and gives the lower bound on the estimation error when the number of sources is larger than the number of physical sensors.

In our research, the deterministic parameter vector $\boldsymbol{\xi}$ can be expressed as

$$\boldsymbol{\xi}=\left[\theta^T,\mathbf{b}^T,\sigma^2\right]^T. \tag{60}$$

Then the Fisher information matrix can be written as

$$\mathbf{FIM}=T_k\left[\frac{\partial\bar{\mathbf{z}}}{\partial\boldsymbol{\xi}}\right]^H\left(\mathbf{R}_x^T\otimes\mathbf{R}_x\right)^{-1}\left[\frac{\partial\bar{\mathbf{z}}}{\partial\boldsymbol{\xi}}\right], \tag{61}$$

where

$$\bar{\mathbf{z}}=vec\left(\mathbf{R}_k\right)=\left(\mathbf{A}^*\odot\mathbf{A}\right)\mathbf{b}+\sigma^2vec\left(\mathbf{I}_P\right) \tag{62}$$

$$\frac{\partial\bar{\mathbf{z}}}{\partial\boldsymbol{\xi}}=\left[\frac{\partial\bar{\mathbf{z}}}{\partial\theta_1},\cdots,\frac{\partial\bar{\mathbf{z}}}{\partial\theta_{N_k}},\frac{\partial\bar{\mathbf{z}}}{\partial b_1},\cdots,\frac{\partial\bar{\mathbf{z}}}{\partial b_{N_k}},\frac{\partial\bar{\mathbf{z}}}{\partial\sigma^2}\right], \tag{63}$$

$$\frac{\partial\bar{\mathbf{z}}}{\partial\theta_n}=b_n\left[\frac{\partial\mathbf{a}^*(\theta_n)}{\partial\theta_n}\otimes\mathbf{a}(\theta_n)+\mathbf{a}^*(\theta_n)\otimes\frac{\partial\mathbf{a}(\theta_n)}{\partial\theta_n}\right], \tag{64}$$

$$\frac{\partial\bar{\mathbf{z}}}{\partial b_n}=\mathbf{a}^*(\theta_n)\otimes\mathbf{a}(\theta_n), n=1,\cdots,N_k, \tag{65}$$

$$\frac{\partial\bar{\mathbf{z}}}{\partial\sigma^2}=vec\left(\mathbf{I}_P\right). \tag{66}$$

Therefore, the CRB for the n-th source can be obtained by

$$\text{CRB}(\theta_n)=\left[\mathbf{FIM}^{-1}\right]_{n,n}, \tag{67}$$

for $1\leqslant n\leqslant N_k$. And CRB versus SNR or snapshots can be defined as:

$$\text{CRB}=\frac{1}{KQN_k}\sum_{k=1}^{K}\sum_{j=1}^{Q}\sum_{n=1}^{N_k}\text{CRB}_{k,j}(\theta_n), \tag{68}$$

where $\text{CRB}_{k,j}$ is the j-th Monte Carlo's CRB at time k, N_k is the true number of sources.

B. Evaluation and Measurement

In the following simulations, SNR is defined as:

$$\text{SNR}=10\log\sigma_s^2/\sigma^2, \tag{69}$$

where σ_s^2 is the signal power and σ^2 denotes the noise power. And root mean square error (RMSE) can be expressed as

$$\text{RMSE}=\sqrt{\frac{1}{QK\tilde{N}_k}\sum_{j=1}^{Q}\sum_{k=1}^{K}\sum_{n=1}^{\tilde{N}_k}\left(\tilde{\mathbf{x}}_{nj}(k)-\mathbf{x}_n(k)\right)^2}, \tag{70}$$

where $\tilde{\mathbf{x}}_{nj}(k)$ is the estimated value of n-th true source $\mathbf{x}_n(k)$ at time k for the j-th Monte Carlo. Q is the total number of Monte Carlo experiments, K is the total observation time and $\tilde{N}_k$ stands for the estimated number of sources at time k.

An other evaluation method is the probability of convergence (PROC), which can be defined as

$$\text{PROC}=\frac{1}{QK}\sum_{k=1}^{K}\sum_{j=1}^{Q}1_{kj} \tag{71}$$

with

$$1_{kj} = \begin{cases} 1, if \sqrt{\frac{1}{\tilde{N}_k}\sum_{n=1}^{\tilde{N}_k}\left(\tilde{\mathbf{x}}_{nj}(k) - \mathbf{x}_n(k)\right)^2} < \sigma \\ 0, otherwise \end{cases}, \quad (72)$$

where σ denotes the error threshold.

C. Exponential weighting factor ζ

The exponential weighting factor ζ is determined based on experimental simulations. Fig. 3(a)-(b) run 100 MC to select the appropriate exponential weighting factor ζ, where $M = 4$, $N = 5$ and snapshots $T_k = 100$. Fig. 3(a) shows the performance of the proposed algorithm in terms of ζ for two given values of SNR, and Fig. 3(b) shows the performance of the proposed algorithm versus SNR for 4 given values of ζ. In the single-source scenario described in case 1, the larger the value of ζ, the worse the performance of the proposed algorithm, which indicates that the single source tracking can achieve good results without exponential weighting. In both cases 2 and 3, ζ has optimal values of 4 and 2, respectively. Therefore, in case 1, $\zeta = 1$. In cases 2 and 3, $\zeta = 4$ and $\zeta = 2$ are chosen, respectively. Note that using a slightly different value of ζ does not significantly change the tracking performance. For example, in Case 2, using $\zeta = 4.1$ or $\zeta = 3.9$ leads to similar tracking performance.

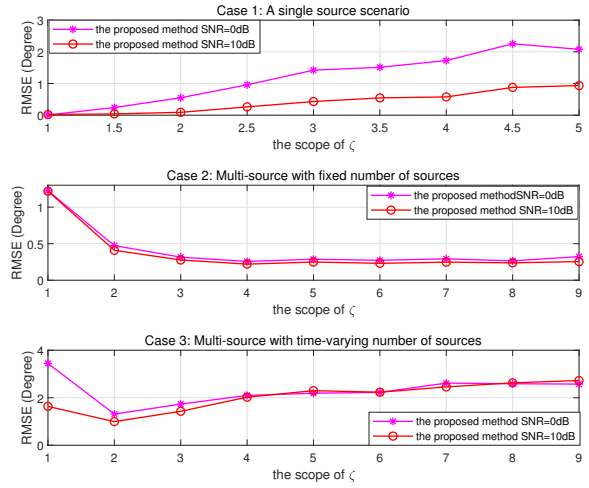
D. Simulation scenarios

In the following three simulations, $M = 4$, $N = 5$ and the number of array elements $P = 2M + N - 1 = 12$ (i.e. the number of ULA sensors), $K = 50$s , $\Delta T = 1$s and $T_k = 200$, the survival and detection probabilities of the sources are assumed to be constants $P_{S,k}(\mathbf{x}_k) = 0.99$ and $P_{D,k}(\mathbf{x}_k) = 0.98$, respectively. The total number of expected hypotheses $J_{max} = 2000$, and $K_B = 5$.

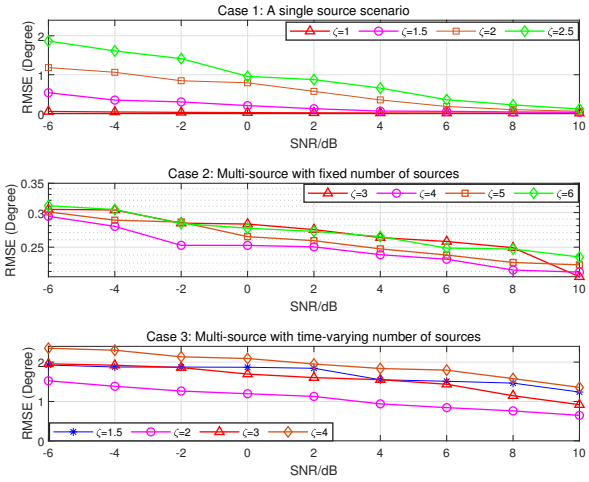
Case 1: A single source scenario

Consider a single source scenario with 1 source, surviving at time $1-40$s. The initial source state is $\mathbf{x}_0^1 = [-31.2°, 1.2]$, and the newborn model is a GLMB RFS with parameters $\boldsymbol{\pi}_B = \left\{\zeta_B^{(l,i)}, p_B^{(l,i)}(\mathbf{x}, l)\right\}_{i=1}^{1}$ where $\zeta_B^{(l,1)} = 0.1$ and the probability density function $p_B^{(l,i)}(\mathbf{x}) = \left(\mathbf{x}_{B,k}^{(l,j)}, \omega_{B,k}^{(l,j)}\right)_{j=1}^{N_{B,k}^l} \sim \mathcal{N}(\mathbf{x}; \mathbf{m}_1, \mathbf{P})$, where $\mathbf{m}_1 = [-30°, 0]$, $\mathbf{P} = diag\left\{4^2, 2^2\right\}$, $\omega_{B,k}^{(l,j)} = 1/N_{B,k}^l$. Each newborn source produces 1000 particles, i.e., $N_B^{\xi}(l) = N_{B,k}^l = 1000$, and $N^{\xi}(l) = 1000$. (The number of particles of GLMB-ULA and PHD-ECA methods is also 1000.)

Fig. 4(a) shows the single source trajectory tracking for one MC and the tracking error for 100 MC is illustrated in Fig. 4(b), where SNR = 10 dB and $T_k = 200$. It can be seen from Fig. 4(a) that the four methods all have a great tracking trajectory capacity. However, in Fig. 4(b), the proposed algorithm has better tracking performance at each time. Compared with the GLMB-ULA method, although the proposed algorithm has a performance improvement of 0.02 degrees, the calculation cost is affordable. Table VI gives the



(a)



(b)

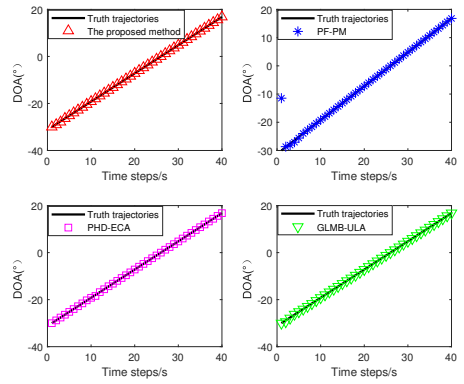
Fig. 3: Selection of exponential weighting factor (a) RMSE versus ζ, $Q = 100$, (b) RMSE versus SNR, $Q = 100$.

average running time of the proposed method and the GLMB-ULA method by averaging the running time of 100 MC trials. The operating environment is Intel(R) Core(TM) i7-10700F CPU @ 2.90GHz 2.90 GHz processor with a 64-bit operating system MATLAB 2020b. TABLE VI shows that the running time of the proposed method is $1.1275s$, which is only 9.95% longer than that of GLMB-ULA method with running time $1.0255s$.

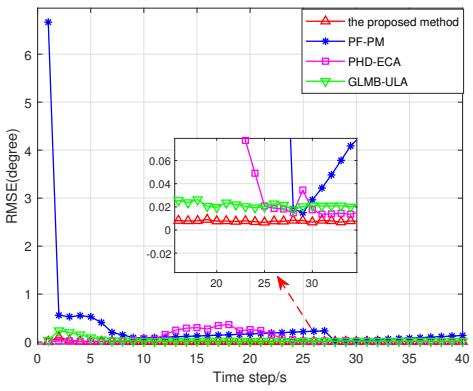
TABLE VI: Average running time, $Q = 100$

Algorithm	Survival particles	Birth particles	Running time/s
GLMB-ULA	1000	1000	1.0255
The proposed Mδ-GLMB	1000	1000	1.1275

Fig. 5 compares the RMSE performance and PROC versus SNRs with $T_k = 200$. Different SNRs from -6 dB to 10 dB with an increment of 2 dB are utilized to generate noisy

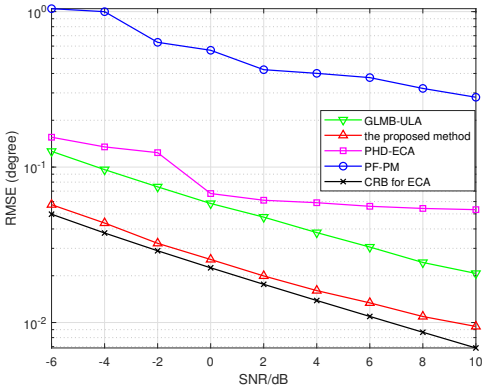


(a)

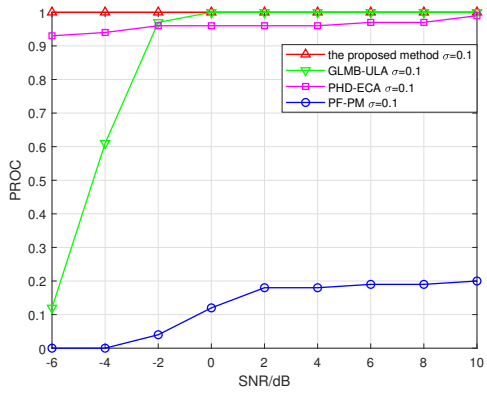


(b)

Fig. 4: (a) A single source trajectory tracking, SNR = 10 dB, $Q = 1$. (b) Tracking error versus time step, SNR = 10 dB, $Q = 100$.



(a)



(b)

Fig. 5: (a) RMSE performance comparison versus SNR, $T_k = 200$, $Q = 100$. (b) PROC versus SNR, $T_k = 200$, $Q = 100$.

environments. For performance analysis, the error threshold $\sigma = 0.1$. As shown in Fig. 5(a), although the RMSE performance of the four methods improves with the increase of SNR, the RMSE of the proposed method is the smallest. The PF-PM method cannot accurately estimate the DOA, resulting in the low PROC in Fig. 5(b). Compared with the RFS based methods, such as GLMB-ULA and PHD-ECA methods, the proposed modified δ-GLMB filtering has better tracking performance. Fig. 5(b) provides the PROC performance comparison of the four algorithms. It can be seen from Fig. 5(b) that the proposed algorithm is superior to other compared methods.

Case 2: Multi-source with fixed number of sources

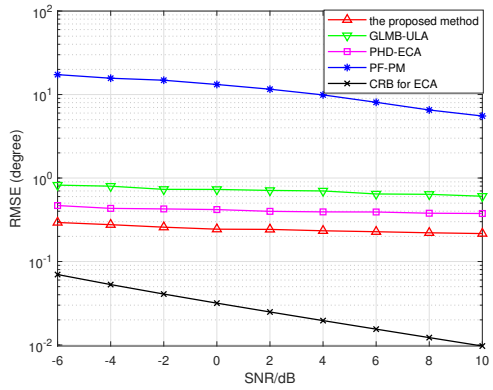
Consider a multi-source scenario with 2 sources, whose survival times are $1-50$s and $1-50$s, respectively. The initial source states are $\mathbf{x}_0^1 = [-41^\circ, 1], \mathbf{x}_0^2 = [51^\circ, -1]$, and the newborn model can be expressed as parameter sets $\boldsymbol{\pi}_B = \left\{\zeta_B^{(l,i)}, p_B^{(l,i)}(\mathbf{x}, l)\right\}_{i=1}^{2}$ where $\zeta_B^{(l,1)} = 0.04, \zeta_B^{(l,2)} = 0.03$ and the probability density function $p_B^{(l,i)}(\mathbf{x}) = \left(\mathbf{x}_{B,k}^{(l,j)}, \omega_{B,k}^{(l,j)}\right)_{j=1}^{N_{B,k}^l} \sim \mathcal{N}(\mathbf{x}; \mathbf{m}_i, \mathbf{P})$, where $\mathbf{m}_1 = [-40^\circ, 0], \mathbf{m}_2 = [50^\circ, 0]$, $\mathbf{P} = diag\left\{2^2, 1^2\right\}$, $\omega_{B,k}^{(l,j)} = 1/N_{B,k}^l$. Each newborn source produces 200 particles, i.e., $N_B^\xi(l) = N_{B,k}^l = 200$, and $N^\xi(l) = 200$. (The number of particles is 200 for the GLMB-ULA method and 1000 for the PHD-ECA algorithm.)

Fig. 6 depicts the comparison of RMSE and PROC versus SNRs, where $Q = 100$, $\sigma = 0.8$ and $T_k = 200$. The RMSE of the four compared methods shown in Fig. 6(a) decreases as SNR increases, and the proposed method has the best performance. Fig. 6(b) illustrates the comparison of PROC in terms of SNRs. It can be seen that the proposed method has better tracking performance than other methods, whose PROC is almost close to 1 as the SNR increases.

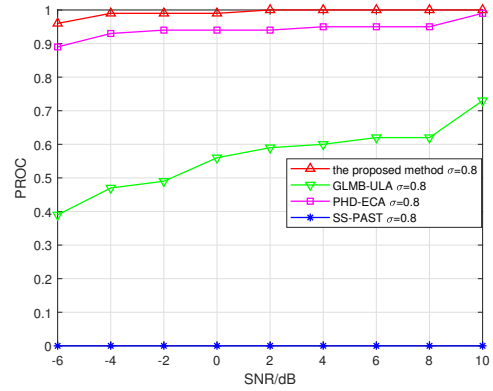
The RMSE comparison results in terms of different snapshots are shown in Fig. 7. The RMSE performance of these methods improves as the number of snapshots grows, and the performance of these algorithms eventually stabilizes as the number of snapshots grows higher, as shown in Fig. 7. However, compared to other approaches, the proposed method's performance is superior, as shown by its lower RMSE and larger PROC.

Case 3: Multi-source with time-varying number of sources

In order to show the viability of the proposed method in the time-varying environment of source number, the survival states of several random sources are given in TABLE VII. The initial source targets are $\mathbf{x}_0^1 = [-31.2^\circ; 1.2]$, $\mathbf{x}_0^2 = [1^\circ; -1.0]$,

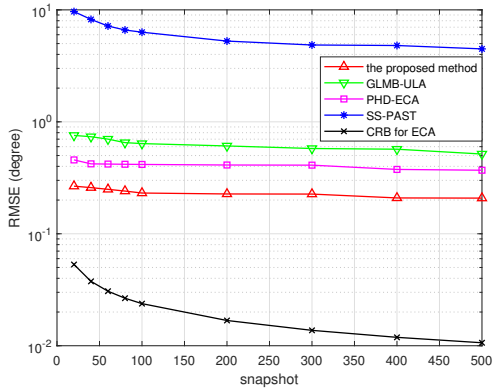


(a)

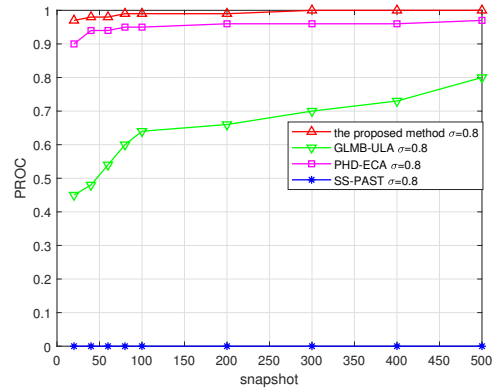


(b)

Fig. 6: Comparison of RMSE and PROC versus SNRs, $Q = 100$, $T_k = 200$. (a) RMSE. (b) PROC.



(a)



(b)

Fig. 7: Comparison of RMSE and PROC versus snapshots, $Q = 100$, SNR = 10 dB. (a) RMSE. (b) PROC.

TABLE VII: Source survival state

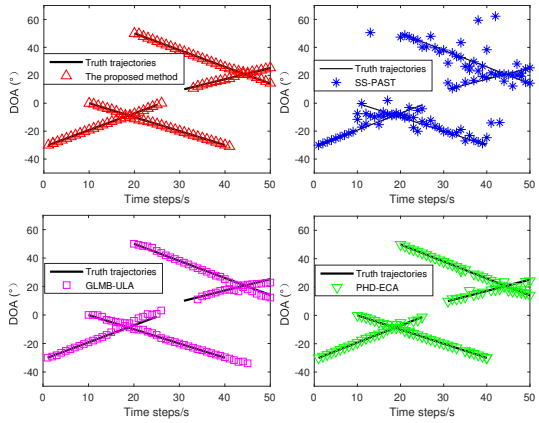
Source	Survival time	initial source state (degree)	Velocities(*rad/s*)
1	1-25s	- 31.2	1.2
2	10-40s	1	-1.0
3	20-50s	51.2	-1.2
4	31-50s	9.2	0.8

$\mathbf{x}_0^3 = [51.2°; -1.2]$, $\mathbf{x}_0^4 = [9.2°; 0.8]$, and the newborn model with parameters $\boldsymbol{\pi}_B = \left\{\zeta_B^{(l,i)}, p_B^{(l,i)}(\mathbf{x}, l)\right\}_{i=1}^{4}$, where $\zeta_B^{(l,1)} = 0.02, \zeta_B^{(l,2)} = 0.02, \zeta_B^{(l,3)} = 0.03, \zeta_B^{(l,4)} = 0.03$ and the probability density function $p_B^{(l,i)}(\mathbf{x}) = \left(\mathbf{x}_{B,k}^{(l,j)}, \omega_{B,k}^{(l,j)}\right)_{j=1}^{N_{B,k}^l} \sim \mathcal{N}(\mathbf{x}; \mathbf{m}_i, \mathbf{P})$, where $\mathbf{m}_1 = [-30°, 0], \mathbf{m}_2 = [0°, 0], \mathbf{m}_3 = [50°, 0], \mathbf{m}_4 = [10°, 0]$, $\mathbf{P} = diag\left\{2^2, 1^2\right\}$, other experiment parameters are the same as Case 2.

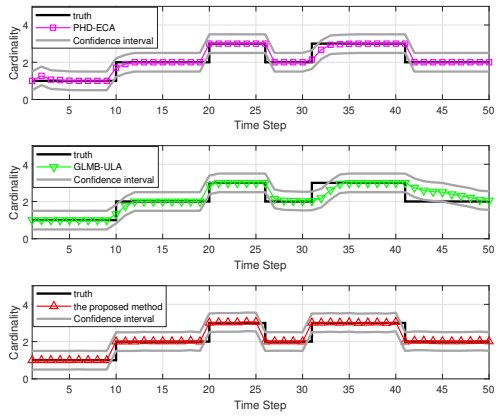
In this scenario, the received sources comprise one source between time steps 1 and 9, two sources between time step 10-19s, 26-30s, and 41-50s, and three sources between time steps 20-25s and 31-40s. The proposed method's tracking result is presented in Fig. 8 where SNR = 10 dB and $T_k = 200$, including the tracking trajectories and cardinality estimation. The results show that the proposed approach can efficiently detect and estimate the source number. Furthermore, the proposed approach can identify the appearance of new sources and the disappearance of old sources, and it can estimate the number of sources accurately. For the GLMB-ULA method, the DOA tracking values deviate from the true state during time steps 31–35s and 41–50s, and even the source number is overestimated (more estimated than the reality). Fig. 8(b) shows that the PHD-ECA approach outperforms the GLMB-ULA method in terms of estimated the number of sources, but it overestimates or underestimates the source number at several time steps. Because the SS-PAST approach is a subspace update-based method that requires a known number of sources, it cannot be used to estimate the number of sources. The experimental results in Fig. 8 show that the proposed method outperforms the GLMB-ULA and the PHD-ECA methods in estimating the source number.

The comparison of RMSE and PROC for various SNRs and snapshots is shown in Figs 9-10. Different SNRs from -6 dB to 10 dB with an increment of 2 dB are employed to generate noisy environments. The DOA tracking performance of all four approaches improves with the increase of SNR or snapshot, as shown in Fig. 9(a) and Fig. 10(a), but the method proposed in this paper has higher tracking performance.

Fig. 9(b) shows the comparison of PROC at different SNRs, where $T_k = 200$, $\sigma = 1$ or $\sigma = 3$. Fig. 10(b) exhibits the comparison of PROC under different snapshot conditions, where SNR = 10, $\sigma = 3$. Because the SS-PAST algorithm has many deviated estimates (as can be seen in Fig. 8(a)) and

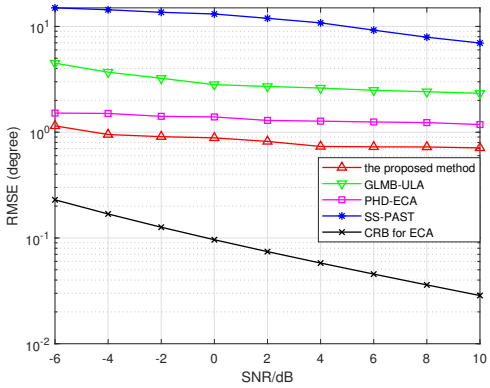


(a)

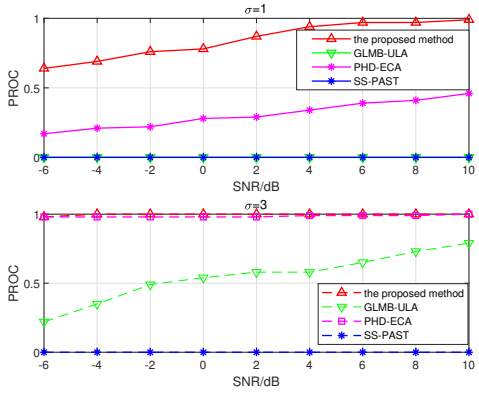


(b)

Fig. 8: (a) Tracking trajectories of time-varying sources, $Q = 1$, $T_k = 200$, SNR = 10 dB. (b) Cardinality with time-varying sources, $Q = 100$, $T_k = 200$, SNR = 10 dB.



(a)



(b)

Fig. 9: Comparison of RMSE and PROC versus SNRs, $Q = 100$, $T_k = 200$. (a) RMSE. (b) PROC.

cannot solve the time-varying source DOA tracking problem, it has the largest RMSE and a very small PROC. It can be seen that the PROC of the proposed method is larger than the other methods under the same threshold σ pre-condition. Similarly, the PROC performance can be improved by increasing the value of threshold σ. In summary, the proposed modified δ-GLMB DOA tracking algorithm outperforms the SS-PAST method, GLMB-ULA method and PHD-ECA method in estimating and tracking the DOA.

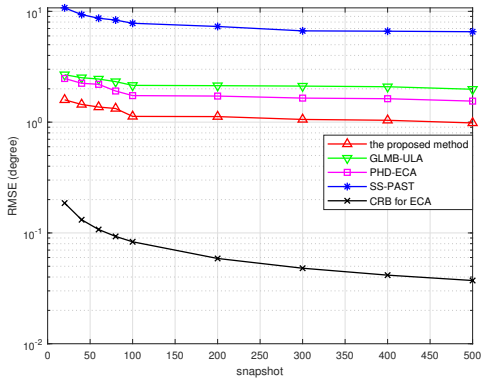
E. Performance of other sparse array

The proposed method can also be extended to other sparse arrays [36]–[40], like nested array (NA) [36], super nested array (SNA) [37] and augmented nested array (ANA) [38]. Therefore, in this subsection, a simulation example of the proposed method with sparse arrays (such as NA, SNA and ANA) is presented. The subarrays $N_1 = N_2 = 3$ are taken in NA-based arrays and $M = 2$, $N = 3$ are taken for ECA. Other simulation parameters are same with those of the Case 3.

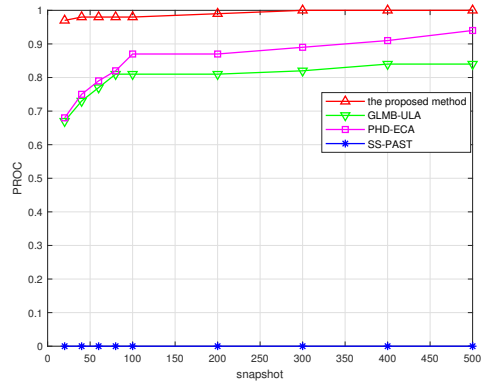
Fig. 11 compares the OSPA (the detailed definition can be found in [41]) distance performance of the proposed algorithm in the context of sparse arrays, where the OSPA distance is considered as the performance metric. This simulation example shows the applicability of the proposed Mδ-GLMB method for the NA, SNA and ANA. Further investigation on Mδ-GLMB with nested arrays or other sparse arrays will be performed in the future.

VI. CONCLUSION

We have addressed the multi-source time-varying DOA tracking problem in this paper. By combining the coprime array methodology with the new measurement association mapping method, a Mδ-GLMB filtering is proposed. To be more specific, the predicted particles are rectified in the Mδ-GLMB filtering prediction stage to improve particle validity. Furthermore, the MUSIC spatial spectral function is applied for the likelihood function of the particles and exponentially weighted, which fixes the mapping problem between the array observations and sources properly. Moreover, the method is extended to the coprime array, which can provide better DOA estimation and tracking performance than the traditional uniform linear array. Simulation comparisons with existing algorithms show the superiority of the proposed algorithm. In addition, the proposed method is not only applicable to the considered coprime array geometries, but also to arbitrary



(a)



(b)

Fig. 10: Comparison of RMSE and PROC versus snapshots, $Q = 100$, SNR = 10 dB. (a) RMSE. (b) PROC.

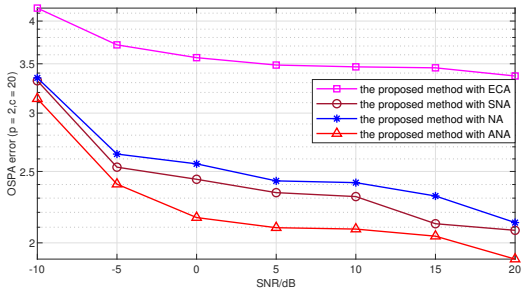


Fig. 11: OSPA distance versus SNR ($p = 2$, $c = 20$), SNR = 10 dB, $Q = 100$, $T_k = 200$.

types of sparse arrays, such as nested arrays, super-nested arrays, etc., which will be discussed in the future work.

REFERENCES

[1] D. Reid, "An algorithm for tracking multiple targets," *IEEE Transactions on Automatic Control*, vol. 24, no. 6, pp. 843–854, 1979.

[2] R. P. Mahler, *Statistical multisource-multitarget information fusion*, vol. 685. Artech House Norwood, MA, USA, 2007.

[3] B. Lin, J. Liu, M. Xie, and J. Zhu, "Direction-of-arrival tracking via low-rank plus sparse matrix decomposition," *IEEE Antennas and Wireless Propagation Letters*, vol. 14, pp. 1302–1305, 2015.

[4] A. Das, "A Bayesian sparse-plus-low-rank matrix decomposition method for direction-of-arrival tracking," *IEEE Sensors Journal*, vol. 17, no. 15, pp. 4894–4902, 2017.

[5] Z. Zheng, Y. Huang, W. Q. Wang, and H. C. So, "Spatial smoothing past algorithm for DOA tracking using difference coarray," *Signal Processing Letters, IEEE*, vol. 126, no. 11, pp. 1623–1627, 2019.

[6] F. Dong, L. Xu, and X. Li, "Particle filter algorithm for DOA tracking using co-prime array," *IEEE Communications Letters*, vol. 24, no. 11, pp. 2493–2497, 2020.

[7] S. Wu, J. Zhao, X. Dong, Q. Xue, and R. Cai, "DOA tracking based on unscented transform multi-Bernoulli filter in impulse noise environment," *Sensors (Basel, Switzerland)*, vol. 19, no. 18, 2019.

[8] T. T. Jeong, "Particle PHD filter multiple target tracking in sonar image," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 1, pp. 409–416, 2007.

[9] N. T. Pham, W. Huang, and S. H. Ong, "Tracking multiple objects using probability hypothesis density filter and color measurements," in *2007 IEEE International Conference on Multimedia and Expo*, pp. 1511–1514, IEEE, 2007.

[10] R. Hoseinnezhad, B.-N. Vo, and B.-T. Vo, "Visual tracking in background subtracted image sequences via multi-Bernoulli filtering," *IEEE Transactions on Signal Processing*, vol. 61, no. 2, pp. 392–397, 2012.

[11] M. Canaud, L. Mihaylova, J. Sau, and N.-E. El Faouzi, "Probabilty hypothesis density filtering for real-time traffic state estimation and prediction," *Networks and Heterogeneous Media (NHM)*, vol. 8, no. 3, pp. 825–842, 2013.

[12] X. Zhang, "Adaptive control and reconfiguration of mobile wireless sensor networks for dynamic multi-target tracking," *IEEE Transactions on Automatic Control*, vol. 56, no. 10, pp. 2429–2444, 2011.

[13] G. Battistelli, L. Chisci, C. Fantacci, A. Farina, and A. Graziano, "Consensus CPHD filter for distributed multitarget tracking," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 3, pp. 508–520, 2013.

[14] M. Ueney, D. E. Clark, and S. J. Julier, "Distributed fusion of PHD filters via exponential mixture densities," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 3, pp. 521–531, 2013.

[15] R. P. Mahler, "Multitarget Bayes filtering via first-order multitarget moments," *IEEE Transactions on Aerospace and Electronic systems*, vol. 39, no. 4, pp. 1152–1178, 2003.

[16] R. Mahler, "PHD filters of higher order in target number," *IEEE Transactions on Aerospace and Electronic systems*, vol. 43, no. 4, pp. 1523–1543, 2007.

[17] B. T. Vo, B. N. Vo, and A. Cantoni, "The cardinality balanced multi-target multi-Bernoulli filter and its implementations," *IEEE Transactions on Signal Processing*, vol. 57, no. 2, pp. 409–423, 2008.

[18] B. T. Vo and B. N. Vo, "Labeled random finite sets and multi-object conjugate priors," *IEEE Transactions on Signal Processing*, vol. 61, no. 13, pp. 3460–3475, 2013.

[19] B. N. Vo, B. T. Vo, and D. Phung, "Labeled random finite sets and the Bayes multi-target tracking filter," *IEEE Transactions on Signal Processing*, vol. 62, no. 24, pp. 6554–6567, 2014.

[20] A. Masnadi-Shirazi and B. D. Rao, "A covariance-based superpositional CPHD filter for multisource DOA tracking," *IEEE Transactions on Signal Processing*, vol. 66, no. 2, pp. 309–323, 2017.

[21] J. Zhao, R. Gui, X. Dong, and S. Wu, "Time-varying DOA tracking algorithm based on generalized labeled multi-Bernoulli," *IEEE Access*, vol. 9, pp. 5943–5950, 2021.

[22] J. Zhao, R. Gui, and X. Dong, "A new measurement association mapping strategy for DOA tracking," *Digital Signal Processing*, vol. 118, p. 103228, 2021.

[23] J. Zhao, R. Gui, and X. Dong, "PHD filtering for multi-source DOA tracking with extended co-prime array: An improved MUSIC pseudo-likelihood," *IEEE Communications Letters*, vol. 25, no. 10, pp. 3267–3271, 2021.

[24] X. Dong, X. Zhang, J. Zhao, M. Sun, and Q. Wu, "Multi-maneuvering sources DOA tracking with improved interactive multi-model multi-Bernoulli filter for acoustic vector sensor (AVS) array," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 8, pp. 7825–7838, 2021.

[25] A. A. Saucan, T. Chonavel, C. Sintes, and J. M. L. Caillec, "CPHD-DOA tracking of multiple extended sonar targets in impulsive environments," *IEEE Transactions on Signal Processing*, vol. 64, no. 5, pp. 1147–1160, 2016.

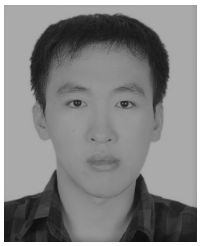
[26] S. Qin, Y. D. Zhang, and M. G. Amin, "Generalized coprime array configurations for direction-of-arrival estimation," *IEEE Transactions on Signal Processing*, vol. 63, no. 6, pp. 1377–1390, 2015.

[27] J. Pan, M. Sun, Y. Wang, and X. Zhang, "An enhanced spatial smoothing technique with ESPRIT algorithm for direction of arrival estimation in coherent scenarios," *IEEE Signal Processing Magazine*, vol. 36, no. 4, pp. 74–84, 2019.

[28] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, 1986.
[29] D. Eppstein, "Finding the k shortest paths," in *Proceedings 35th Annual Symposium on Foundations of Computer Science*, vol. 28, pp. 154–165, 1994.
[30] Q. Cheng, P. Pal, M. Tsuji, and Y. Hua, "An MDL algorithm for detecting more sources than sensors using outer-products of array output," *IEEE Transactions on Signal Processing*, vol. 62, no. 24, pp. 6438–6453, 2014.
[31] C. Musso, N. Oudjane, and F. Le Gland, "Improving regularised particle filters," *Sequential Monte Carlo methods in practice*, pp. 247–271, 2001.
[32] P. Stoica and A. Nehorai, "MUSIC, maximum likelihood and Cramer Rao bound: further results and comparisons," *IEEE Transactions on Acoustics Speech and Signal Processing*, vol. 38, no. 12, pp. 2140–2150, 2002.
[33] C. L. Liu and P. P. Vaidyanathan, "Cramer Rao bounds for coprime and other sparse arrays, which find more sources than sensors," *Digital Signal Processing*, vol. 61, pp. 43–61, 2017.
[34] C. Zhou, Y. Gu, X. Fan, Z. Shi, G. Mao, and Y. D. Zhang, "Direction-of-arrival estimation for coprime array via virtual array interpolation," *IEEE Transactions on Signal Processing*, vol. 66, no. 22, pp. 5956–5971, 2018.
[35] M. Wang and A. Nehorai, "Coarrays, MUSIC, and the Cramer Rao bound," *IEEE Transactions on Signal Processing*, vol. 65, no. 4, pp. 933–946, 2016.
[36] P. Pal and P. P. Vaidyanathan, "Nested arrays: A novel approach to array processing with enhanced degrees of freedom," *IEEE Transactions on Signal Processing*, vol. 58, no. 8, pp. 4167–4181, 2010.
[37] C. L. Liu and P. P. Vaidyanathan, "Super nested arrays: Linear sparse arrays with reduced mutual coupling?part i: Fundamentals," *IEEE Transactions on Signal Processing*, vol. 64, no. 15, pp. 3997–4012, 2016.
[38] J. Liu, Y. Zhang, Y. Lu, S. Ren, and S. Cao, "Augmented nested arrays with enhanced dof and reduced mutual coupling," *IEEE Transactions on Signal Processing*, vol. 65, no. 21, pp. 5549–5563, 2017.
[39] A. Raza, W. Liu, and Q. Shen, "Thinned coprime array for second-order difference co-array generation with reduced mutual coupling," *IEEE Transactions on Signal Processing*, vol. 67, no. 8, pp. 2052–2065, 2019.
[40] Z. Zheng, Y. Huang, W.-Q. Wang, and H. C. So, "Augmented covariance matrix reconstruction for doa estimation using difference coarray," *IEEE Transactions on Signal Processing*, vol. 69, pp. 5345–5358, 2021.
[41] D. Schuhmacher, B. T. Vo, and B. N. Vo, "A consistent metric for performance evaluation of multi-object filters," *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3447–3457, 2008.

Xudong Dong received the M.S. degree in applied mathematics from Guilin University of Electronic Technology in 2020. He is pursuing the Ph.D. degree in communication and information systems with the College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, China. His research interests are target detection and tracking, array signal processing, deep learning, signal classification and impulsive noise.

Jun Zhao received the M.S. degree in applied mathematics from the Guilin University of Electronic Technology, Guilin, China, in 2020. She is currently working toward the Ph.D. degree in information and communication engineering with the College of Electronic and Information Engineering, Tongji University, Shanghai, China. Her research interests include target detection and tracking, array signal processing, and machine learning.



Meng Sun received his B.S. and his M.S. degrees from Northwest University and South China University of Technology, China in 2010 and 2013, respectively. He received the Ph.D. degree from IETR Laboratory at Polytech Nantes (Université de Nantes, France) in 2016. He is currently with Electronic Engineering Department, Nanjing University of Aeronautics and Astronautics, Nanjing, China. His main research focus on array signal processing, spectral analysis, and ground penetrating radar.

Xiaofei Zhang received M.S degree from Wuhan University, Wuhan, China, in 2001. He received Ph.D. degrees in communication and information systems from Nanjing University of Aeronautics and Astronautics in 2005. Now, he is a Professor in Electronic Engineering Department, Nanjing University of aeronautics and astronautics, Nanjing, China. His research is focused on array signal processing and communication signal processing.

Yide Wang received the B.S. degree in electrical engineering from the Beijing University of Post and Telecommunication, Beijing, China, in 1985, and the M.S. and the Ph.D. degrees in signal processing and telecommunications from the University of Rennes, France, in 1986 and 1989, respectively. He is currently a Professor with the Ecole Polytechnique de l'Université de Nantes. His research interests include array signal processing, spectral analysis, and mobile wireless communication systems.