

Article

δ -Generalized Labeled Multi-Bernoulli Filter Using Amplitude Information of Neighboring Cells

Chao Liu, Jinping Sun , Peng Lei * and Yaolong Qi

School of Electronic and Information Engineering, Beihang University, Beijing 100191, China; LC2016@buaa.edu.cn (C.L.); sunjinping@buaa.edu.cn (J.S.); longgny@163.com (Y.Q.)

* Correspondence: buaaray@gmail.com; Tel.: +86-10-8233-9203

Received: 5 February 2018; Accepted: 4 April 2018; Published: 10 April 2018



Abstract: The amplitude information (AI) of echoed signals plays an important role in radar target detection and tracking. A lot of research shows that the introduction of AI enables the tracking algorithm to distinguish targets from clutter better and then improves the performance of data association. The current AI-aided tracking algorithms only consider the signal amplitude in the range-azimuth cell where measurement exists. However, since radar echoes always contain backscattered signals from multiple cells, the useful information of neighboring cells would be lost if directly applying those existing methods. In order to solve this issue, a new δ -generalized labeled multi-Bernoulli (δ -GLMB) filter is proposed. It exploits the AI of radar echoes from neighboring cells to construct a united amplitude likelihood ratio, and then plugs it into the update process and the measurement-track assignment cost matrix of the δ -GLMB filter. Simulation results show that the proposed approach has better performance in target's state and number estimation than that of the δ -GLMB only using single-cell AI in low signal-to-clutter-ratio (SCR) environment.

Keywords: δ -GLMB filter; amplitude information; neighboring cells; multi-target tracking

1. Introduction

The multi-target tracking (MTT) [1,2] is an important capability for modern radar data processing. Taking the pulsed radar as an example, the pulse compression and clutter suppression are first used to remove clutter and jamming signals from received echoes. Then we can extract target information via integration and detection. A threshold is usually set in detection algorithms according to the Newman-Pearson discipline, and the echo signal beyond the threshold is extracted as plots. Due to the limitations of clutter suppression or detection, these plots may originate from not only targets but also clutter. Moreover, the missed detection of targets can also occur. The plots are then put into the data preprocessing (such as plots centroid, etc.) for the preparation of target tracking. The objective of MTT is to estimate the unknown and time-varying number of targets as well as their individual states from the observation sequence. It improves the radar sensing of moving targets in current search area.

The traditional multi-target tracking paradigms, such as Joint Probabilistic Data Association (JPDA) [3] and Multiple Hypotheses Tracking (MHT) [4], implement data association followed by the Kalman filter for single target. When targets are spatially close or a large number of false alarms are present, the data association becomes very complex. Therefore, Mahler proposes the multi-target filtering theory based on Random Finite Sets (RFS) [5]. In the framework of this theory, multi-target states and sensor measurements are modeled as random finite sets, and the recursive estimation of multi-target posterior density is achieved by Bayesian multi-target filtering. The early implementations of RFS paradigm, such as probability hypothesis density (PHD) [6], cardinalized probability hypothesis density (CPHD) [7] and Multi-Bernoulli [8] filters, ignore the data association in order to reduce the computational complexity. Hence they cannot produce target tracks. On the basis, δ -generalized

labeled Multi-Bernoulli (δ -GLMB) [9,10] and labeled Multi-Bernoulli (LMB) [11] filters incorporate track labels into target state and take data association as part of multi-target state estimation to realize the output of target tracks.

The traditional radar signal processing procedure is first detection and then tracking. Nevertheless, when the detection performance is unsatisfactory, the tracking accuracy will decline. In this case, the track-before-detect (TBD) strategy [12–16] is proposed. It gets rid of the detection in any single frame, but accumulates the likelihood ratio of continuous multiple frames. This strategy provides an effective way for detecting low observable targets. It should be noted that the tracking also carries out likelihood ratio accumulation, and the difference between them is: (1) the likelihood ratio accumulated by TBD approaches contains both dynamic states and amplitude information (AI) of the target, while the tracking approaches only contains the dynamic states; (2) TBD deals with the original echoes, and traverses all range-Doppler-azimuth cells over the observation area, while the tracking only needs to process the data after thresholding. Thus, more information amount and computational burden are involved in the TBD. This disadvantage leads to the fact that TBD is currently difficult to replace the traditional detection-and-tracking procedure in real-time radar data processing. However, it outperforms the traditional procedure in low signal-to-clutter-ratio (SCR) conditions due to the full use of amplitude difference between target and clutter. Therefore, we can see that the tracking performance in low SCR conditions could be improved by using AI.

Many works in the literature have introduced AI into tracking algorithms. In [17], the detection process of echoed signals with slow Rayleigh fluctuation in narrowband Gaussian background is analyzed. It incorporates the likelihood ratio of amplitude into the calculation of measurement-track association weights for improving performance of the tracking filter. Based on the AI aided measurement-track association (AIA-MTA) approach in [17], more studies have further been carried out. In [18], Lerro combines the AIA-MTA with the interactive multiple model (IMM), and improves the performance of track formation and maintenance in clutter environment. To solve the track initiation problem of low observable target, Cai introduces AI into the adaptive sliding window expectation-maximization algorithm combined with maximum likelihood [19]. By analytically assessing the benefits of AIA-MTA in several simple cases, Ehrman points out that the method is unsuitable for multiple targets although it is good at tracking single target in dense clutter environment. The reason is that AIA-MTA always favors the measurement with high amplitude, but ignores the measurement states [20]. Then he proposes that the normalized target amplitude likelihood function instead of the likelihood ratio should be introduced into AIA-MTA. In [21], the error probabilities of measurement-track association for Rayleigh target, fixed amplitude and Rician target are calculated. In [22,23], Ehrman further presents that the blind use of AI in the measurement-track assignment may actually reduce the association performance, because no method can be applied to all scenarios. In [24], the performance loss in heavy-tailed clutter environment is analyzed quantitatively by simulation. In [25], a modified Riccati equation is used to predict the performance of probabilistic data association filter with AI in heavy-tailed K distributed clutter. In [26,27], the amplitude likelihood ratio proposed in [17] is introduced to the PHD filter (simplified as PHD-AI). For situations of high resolution and low grazing angle, an AI assisted PHD filter in Weibull clutter background is proposed by Li [28]. In [29], Liu proposes to improve the association performance with AI in complex ground target tracking. In [30], Yuan gives an improved multi-Bernoulli filter with AI.

However, one obvious disadvantage of these existing researches is that when constructing, only the AI of the measurement cell is used for constructing the amplitude likelihood function, while that of surrounding cells is ignored. In some cases, for example, when the size of the target is large or the sampling interval of range and azimuth is small, the target echoes may occupy multiple adjacent cells. After detection process, the plots are likely to be distributed in multiple adjacent cells. This phenomenon is called spread [12] or spillover [16] of target energy. In order to reduce complexity, most of target tracking approaches require that one measurement is generated for a target in each frame [5]. Therefore, the plots centroid technology is usually applied to handle the neighboring plots,

and a central measurement is obtained according to certain principles. The central measurement would be used as the only target measurement for tracking, while the other neighboring plots are discarded. If AI of surrounding cells is preserved at the step of plots centroid, the amount of available information for filtering will be greatly increased. It could help improve the tracking performance. By now, this idea has been used in some TBD approaches [12,13,16].

Based on the above analysis, this paper proposes a new δ -GLMB filter with united likelihood ratio of AI in neighboring cells in the complex Gaussian distributed clutter (simplified as GLMB-AI-UL). First, using PHD-AI [27], a δ -GLMB filter with single cell AI (simplified as GLMB-AI) is derived. Then the amplitude measurement is modeled with the point spread function employed in TBD approaches. However, different from traversing all cells in TBD to calculate the amplitude likelihood ratio, the method only considers cells around each measurement after plots centroid to obtain the likelihood ratio. Thus, its calculation cost is significantly lower than that of TBD. Finally, the likelihood ratio is introduced into δ -GLMB update process and measurement-track assignment cost matrix to improve its filtering performance.

The remainder of this paper is organized as follows. A brief overview of RFS and the δ -GLMB filter are provided in Section 2. PHD-AI is introduced in Section 3. On this basis the GLMB with united likelihood ratio of AI in neighboring resolution cells is proposed in Section 4. Section 5 presents simulation results. Conclusions are made in Section 6.

2. Bayesian Multi-Target Filtering

2.1. Basic Notations

The usage of notations in the paper follows the same way as those in [9,10]. Single-target states are denoted by small letters (e.g., x) and multi-target states are denoted by capital letters (e.g., X). In labeled RFSs, labeled target states are denoted by bold face letters (e.g., \mathbf{X}). In addition, spaces are usually denoted by blackboard bold letters (e.g., \mathbb{X} represents the state space, and \mathbb{L} represents the label space). The group of all finite subsets of \mathbb{X} is denoted by $\mathcal{F}(\mathbb{X})$. A labeled single-target state \mathbf{x} consists of a kinematic state $x \in \mathbb{X}$ and a label $\ell \in \mathbb{L}$. A single measurement is denoted by a small letter (e.g., z), and a set of multiple measurements is denoted by a capital letter (e.g., Z).

The standard inner product of $f(x)$ and $g(x)$ is defined as

$$\langle f, g \rangle \triangleq \int f(x)g(x)dx.$$

The multi-target exponential function is defined as

$$h^X \triangleq \prod_{x \in X} h(x),$$

where h is a real-valued function, and it is usual to set $h^\emptyset = 1$. The generalized Kronecker delta function is defined as

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

where X and Y can be arbitrary arguments such as sets, vectors and integers. Meanwhile, the inclusion function is defined as

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

2.2. Bayesian Multi-Target Filtering

Assume that there exist $N(k)$ targets $x_{k,1}, \dots, x_{k,N(k)}$ at time k . Each of them takes values from the labeled state space $\mathbb{X} \times \mathbb{L}$, and their measurements $z_{k,1}, \dots, z_{k,M(k)}$ takes values from the observation space \mathbb{Z} . As a result, the collection of targets and measurements at time k can be modeled as [9]

$$\begin{aligned} \mathbf{X}_k &= \{x_{k,1}, \dots, x_{k,N(k)}\}, \\ Z_k &= \{z_{k,1}, \dots, z_{k,M(k)}\}. \end{aligned}$$

The aim of multi-target tracking is to estimate the multi-target posterior density $\pi_{0:k}(X_{0:k}|Z_{0:k})$ based on the measurement until now. It can be realized recursively by

$$\pi_{0:k}(X_{0:k}|Z_{0:k}) \propto g_k(Z_k|\mathbf{X}_k)f_{k|k-1}(\mathbf{X}_k|\mathbf{X}_{k-1})\pi_{0:k-1}(X_{0:k-1}|Z_{0:k-1}), \tag{1}$$

where $X_{0:k} = (X_0, \dots, X_k)$, $Z_{0:k} = (Z_0, \dots, Z_k)$, $g_k(\cdot|\cdot)$ is the multi-target likelihood function at time k , and $f_{k|k-1}(\cdot|\cdot)$ is the multi-target transition density from time $k - 1$ to time k .

In order to obtain the multi-target posterior probability density at current moment k , we omit the dependence on past measurements for simplicity. Thus, it can be realized from time $k - 1$ by prediction and update formulations [4,5] as follows

$$\pi_{k+1|k}(\mathbf{X}_{k+1}) = \int f_{k+1|k}(\mathbf{X}_{k+1}|\mathbf{X}_k)\pi_k(\mathbf{X}_k|Z_k)\delta\mathbf{X}_k, \tag{2}$$

$$\pi_{k+1}(\mathbf{X}_{k+1}|Z_{k+1}) = \frac{g_{k+1}(Z_{k+1}|\mathbf{X}_{k+1})\pi_{k+1|k}(\mathbf{X}_{k+1})}{\int g_k(Z_{k+1}|\mathbf{X}_{k+1})\pi_{k+1|k}(\mathbf{X}_{k+1})\delta\mathbf{X}_{k+1}}, \tag{3}$$

where the integral in Equation (2) is a set integral for any function $f:\mathcal{F}(\mathbb{X} \times \mathbb{L}) \rightarrow \mathbb{R}$, and defined by

$$\int f(\mathbf{X})\delta\mathbf{X} = \sum_{i=0}^{\infty} \frac{1}{i!} \int f(\{x_1, \dots, x_i\})d(x_1, \dots, x_i). \tag{4}$$

2.3. δ -GLMB Filter

Because of the paper length limit, only the update process of δ -GLMB filter is introduced herein. For more details of the filter, readers can refer to [9,10].

Let the current multi-target prediction density has the δ -GLMB as

$$\pi(\mathbf{X}) = \Delta(\mathbf{X}) \sum_{(I,\xi) \in \mathcal{F}(\mathbb{L}) \times \Xi} w^{(I,\xi)} \delta_I(\mathcal{L}(\mathbf{X})) \left[p^{(\xi)} \right]^{\mathbf{X}}, \tag{5}$$

where $w^{(I,\xi)}$ is the weight of the prediction component (I, ξ) , and $p^{(\xi)}$ is the probability density function (pdf) of a single target. Then the multi-target filtering density of current moment is also a δ -GLMB given by

$$\pi(\mathbf{X}|Z) = \Delta(\mathbf{X}) \sum_{(I,\xi) \in \mathcal{F}(\mathbb{L}) \times \Xi} \sum_{\theta \in \Theta(I)} w^{(I,\xi,\theta)} \delta_I(\mathcal{L}(\mathbf{X})) \left[p^{(\xi,\theta)}(\cdot|Z) \right]^{\mathbf{X}}, \tag{6}$$

where

$$w^{(I,\xi,\theta)} \propto w^{(I,\xi)} \left[\eta_Z^{(\xi,\theta)} \right]^I, \tag{7}$$

$$\eta_Z^{(\xi,\theta)}(\ell) = \left\langle p^{(\xi)}(\cdot, \ell), \psi_Z(\cdot, \ell; \theta) \right\rangle, \tag{8}$$

$$p^{(\xi,\theta)}(x, \ell|Z) = \frac{p^{(\xi)}(x, \ell)\psi_Z(x, \ell; \theta)}{\eta_Z^{(\xi,\theta)}(\ell)}, \tag{9}$$

$$\psi_Z(x, \ell; \theta) = \begin{cases} \frac{p_D(x, \ell)g(z_{\theta(\ell)}|x, \ell)}{k(z_{\theta(\ell)})} & \theta(\ell) > 0 \\ 1 - p_D(x, \ell) & \theta(\ell) = 0 \end{cases}, \tag{10}$$

and $\Theta(I)$ denotes a subset of association maps at current time with domain I .

3. Amplitude Information Aided Multi-Target Filter

3.1. Amplitude Information Modeling

It is supposed that an augmented state including AI of the $i - th$ target at time k is represented as

$$x_k^i := [\tilde{x}_k^i; d], \tag{11}$$

where $\tilde{x}_k^i = [p_{1,k}^i \ p_{2,k}^i \ \dot{p}_{1,k}^i \ \dot{p}_{2,k}^i]^T$ is the basic state including positions $(p_{1,k}^i, p_{2,k}^i)$ and velocities $(\dot{p}_{1,k}^i, \dot{p}_{2,k}^i)$, and d is the power ratio of target and clutter signals. For computational convenience, the clutter power is usually normalized, and then $1 + d$ represents the mean normalized SCR, which is typically expressed in a logarithmic form

$$SCR = 10 \log_{10}(1 + d). \tag{12}$$

Each measurement contains a two-dimensional target position \tilde{z} and amplitude a

$$z_k^i := [\tilde{z}_k^i; a]. \tag{13}$$

To simplify notations, we use x, \tilde{x}, z and \tilde{z} to denote $x_k^i, \tilde{x}_k^i, z_k^i$ and \tilde{z}_k^i , respectively.

Assuming that the amplitude is independent of the dynamic state, the measurement likelihood function of the target $g(z|x)$ and that of the clutter $c(z)$ can be given by

$$g(z|x) = g_{\tilde{z}}(\tilde{z}|x)g_a(a|d), \quad a \geq 0, \tag{14}$$

$$c(z) = c_{\tilde{z}}(\tilde{z})c_a(a), \quad a \geq 0, \tag{15}$$

where $g_{\tilde{z}}(\tilde{z}|x)$ and $c_{\tilde{z}}(\tilde{z})$ are dynamic state likelihoods for target and clutter, respectively. $g_a(a|d)$ and $c_a(a)$ are amplitude likelihoods for target and clutter, respectively. Using these amplitude likelihoods we can compute the detection probability p_D^τ and false alarm probability p_{FA}^τ as

$$p_D^\tau(d) = \int_\tau^\infty g_a(a|d)da, \tag{16}$$

$$p_{FA}^\tau = \int_\tau^\infty c_a(a)da. \tag{17}$$

Then the normalized amplitude likelihood function can be expressed as

$$g_a^\tau(a|d) = \frac{1}{p_D^\tau(d)}g_a(a|d), \quad a > \tau, \tag{18}$$

$$c_a^\tau(a) = \frac{1}{p_{FA}^\tau}c_a(a), \quad a > \tau. \tag{19}$$

3.2. Amplitude Information Aided PHD Filter

The PHD-AI filter is derived in [27], and the main results are just listed as follows.

Let λ denote the expected number of Poisson distributed clutter before detection. Then the expected number after detection is λp_{FA}^τ , which is also Poisson distributed. The pdf of clutter after detection is given by

$$f_C(K) := \exp(-\lambda p_{FA}^\tau) \prod_{z \in K} \lambda p_{FA}^\tau c_a^\tau(a) c_{\tilde{z}}(\tilde{z}|\tilde{x}). \tag{20}$$

When pdf of clutter and AI of target are incorporated, the multi-target likelihood function is modified as

$$f_{k+1}(Z_{k+1}|X_{k+1}) = f_C(Z_{k+1}) \cdot \prod_i^n (1 - p_D^\tau(d_i)) \times \sum_{\theta} \prod_{i:\theta(i)>0} \frac{p_D^\tau(d_i) g_{\tilde{z}}(\tilde{z}_{\theta(i)}|\tilde{x}_i) g_a^\tau(a_{\theta(i)}|d_i)}{(1 - p_D^\tau(d_i)) \lambda p_{FA}^\tau c_{\tilde{z}}(\tilde{z}_{\theta(i)}) c_a^\tau(a_{\theta(i)})}, \tag{21}$$

where the summation notation represents the sum over all possible associations θ between X and Z . The difference between the above and the standard multi-object likelihood function is the introduction of amplitude likelihood ratio, i.e.,

$$\Lambda^\tau = \frac{g_a^\tau(a_{\theta(i)}|d_i)}{c_a^\tau(a_{\theta(i)})}. \tag{22}$$

Therefore the PHD pseudo-likelihood is given by

$$L_Z(x) = (1 - p_D^\tau(d)) + p_D^\tau(d) \times \sum_{z \in Z} \prod_{i:\theta(i)>0} \frac{p_D^\tau(d) g_a^\tau(a|d) g_{\tilde{z}}(\tilde{z}|\tilde{x})}{\lambda p_{FA}^\tau c_{\tilde{z}}(\tilde{z}) c_a^\tau(a) + \langle D_{k+1|k}, p_D^\tau(d) g_{\tilde{z}} \rangle}. \tag{23}$$

4. δ -GLMB Filter Using Amplitude Information of Neighboring Cells

In this section, we firstly introduce AI into the δ -GLMB filter on the basis of PHD-AI, namely the GLMB-AI. Compared with PHD-AI, an attractive advantage of GLMB-AI is the capability to output target tracks. Secondly, the point spread model used in TBD approaches is exploited to model the measurement set, and then the amplitude likelihood ratio of the neighboring cells in complex Gaussian clutter is derived. Finally, the united likelihood ratio of the neighboring cells is incorporated into the δ -GLMB update process and measurement-track assignment matrix. For simplicity, the new method is called GLMB-AI-UL.

4.1. GLMB-AI

4.1.1. Update

According to [10], if only the dynamic state information is considered, the multi-target likelihood function is given by

$$g(Z|X) = e^{-\langle k, 1 \rangle} k^Z \sum_{\theta \in \Theta(\mathcal{L}(X))} [\psi_{\tilde{z}}(\cdot; \theta)]^X, \tag{24}$$

where $k = \lambda \cdot c_{\tilde{z}}(\tilde{z})$ is the intensity function of clutter, and the likelihood is

$$\psi_{\tilde{z}}(\tilde{x}, \ell; \theta) = \begin{cases} \frac{p_D^\tau(\tilde{x}, \ell) g(\tilde{z}_{\theta(\ell)}|\tilde{x}, \ell)}{k(\tilde{z}_{\theta(\ell)})}, & \theta(\ell) > 0 \\ 1 - p_D^\tau(\tilde{x}, \ell), & \theta(\ell) = 0 \end{cases}. \tag{25}$$

Substitute the pdf of clutter in Equation (20) into Equation (24), and the amplitude likelihood ratio in Equation (22) into Equation (25). According to the Proposition 7 in [8], we can derive the posterior pdf of GLMB-AI as

$$\pi(\mathbf{X}|Z) = \Delta(\mathbf{X}) \sum_{(I,\xi) \in \mathcal{F}(\mathbb{L}) \times \Xi} \sum_{\theta \in \Theta(I)} w^{(I,\xi,\theta)} \delta_I(\mathcal{L}(\mathbf{X})) \left[p^{(\xi,\theta)}(\cdot|Z) \right]^{\mathbf{X}}, \tag{26}$$

where

$$w^{(I,\xi,\theta)} \propto w^{(I,\xi)} \left[\eta_Z^{(\xi,\theta)} \right]^I, \tag{27}$$

$$\eta_Z^{(\xi,\theta)}(\ell) = \left\langle p^{(\xi)}(\cdot, \ell), \psi_Z^\tau(\cdot, \ell; \theta) \right\rangle, \tag{28}$$

$$p^{(\xi,\theta)}(x, \ell|Z) = \frac{p^{(\xi)}(x, \ell) \psi_Z^\tau(x, \ell; \theta)}{\eta_Z^{(\xi,\theta)}(\ell)}, \tag{29}$$

$$\psi_Z^\tau(x, \ell; \theta) = \begin{cases} \frac{p_D^\tau(x, d, \ell) g_z(\tilde{z}_\theta(\ell) | \tilde{x}, \ell) g_a^\tau(a_\theta(\ell) | d, \ell)}{\lambda c_z(\tilde{z}_\theta(\ell)) c_a^\tau(a_\theta(\ell)) p_{FA}^\tau}, & \theta(\ell) > 0 \\ 1 - p_D^\tau(x, d, \ell), & \theta(\ell) = 0 \end{cases}. \tag{30}$$

4.1.2. Ranked Measurement-Track Assignment

In the update process, the ranked assignment algorithm proposed by Murty is used to calculate the assignment cost of every measurement to track and seek the least cost assignment matrices [10]. Under the condition that only the target dynamic information is considered, each assignment cost is given by

$$C_{i,j} = -\ln \left(\frac{\left\langle p^{(\xi)}(\tilde{x}, \ell_i), p_D(\tilde{x}, \ell_i) g(\tilde{z}_j | \tilde{x}, \ell_i) \right\rangle}{\left\langle p^{(\xi)}(\tilde{x}, \ell_i), 1 - p_D(\tilde{x}, \ell_i) \right\rangle k(\tilde{z}_j)} \right). \tag{31}$$

Now we bring the target amplitude likelihood in Equation (14) and the clutter amplitude likelihood in Equation (15) into $g(\tilde{z}_j | \tilde{x}, \ell_i)$ and $k(\tilde{z}_j)$, and then obtain a new assignment cost as

$$C_{i,j} = -\ln \left(\frac{\left\langle p^{(\xi)}(\tilde{x}, \ell_i), p_D^\tau(x, d, \ell_i) g_z(\tilde{z}_j | \tilde{x}, \ell_i) g_a^\tau(a_j | d, \ell_i) \right\rangle}{\left\langle p^{(\xi)}(\tilde{x}, \ell_i), 1 - p_D(\tilde{x}, \ell_i) \right\rangle k(z_j)} \right), \tag{32}$$

where $k(z_j) = \lambda p_{FA}^\tau c_z(\tilde{z}_j) c_a^\tau(a_j)$ is the function of clutter intensity after detection. According to the representation of each single target density, the above assignment cost can be computed by Gaussian mixture or sequential Monte Carlo, and the specific calculation process can refer to [10].

4.2. GLMB-AI-UL

4.2.1. Spread Model of Target Amplitude

In this section, we model the amplitude spread phenomenon by taking Figure 1 as an example. In Figure 1, there are three spread targets in complex Gaussian clutter background. We use the point spread function $h^{il}(x_k^i)$ widely used in TBD approaches to describe this phenomenon [12], i.e.,

$$h^{il}(x_k^i) = \exp \left\{ -\frac{(r_k^j - r_k^i)^2}{2R} L_r - \frac{(\theta_k^l - \theta_k^i)^2}{2B} L_b \right\} \tag{33}$$

where (r_k^i, θ_k^i) is the distance and azimuth of the sampling cell with respect to the target x_k^i , and (r_k^j, θ_k^l) , $j = 1, \dots, N_r$, $l = 1, \dots, N_\theta$, is the distance and azimuth of any sampling cell in the observation

area. Here N_r and N_θ are the total number of range bins and azimuth bins of the observation area, respectively. R and B are constants with respect to sizes of range bins and azimuth bins, and L_r and L_b are fading coefficients. The relationship between the target state in Equation (33) and that in Equation (11) is given by

$$r_k^i = \sqrt{(p_{1,k}^i)^2 + (p_{2,k}^i)^2} \tag{34}$$

$$\theta_k^i = \arctan\left(\frac{p_{2,k}^i}{p_{1,k}^i}\right) \tag{35}$$

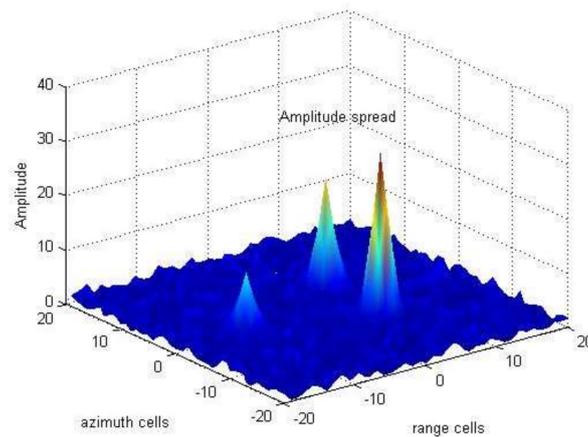


Figure 1. Spread phenomenon of target amplitude.

Suppose that the clutter in each unit is independent and identically distributed, and c_k^{jl} denotes the clutter in unit (j, l) at time k . Then the measurement of this unit can be represented as

$$z_k^{jl} = \sum_{i=1}^{N_k} A_k^i \exp(\psi_k^i) h^{jl}(x_k^i) + c_k^{jl} \tag{36}$$

where A_k^i denotes amplitude measurement of the i -th target at time k , ψ_k^i represents the phase of the target echo signal, and c_k^{jl} denotes zero mean complex Gaussian clutter. It should be noted that A_k^i may be fixed or fluctuate according to Swerling types I~IV, and ψ_k^i is usually assumed to be uniformly distributed in $[0, 2\pi)$. Because the target energy tends to spread only to adjacent units, let us suppose that targets are spatially so far away that each unit is only affected by the target closest to it. Then Equation (36) can be simplified to

$$z_k^{jl} = A_k^i \exp(\psi_k^i) h^{jl}(x_k^i) + c_k^{jl} \tag{37}$$

When there is no target energy in unit (j, l) , the measurement would be

$$z_k^{jl} = c_k^{jl} \tag{38}$$

After detection, the points exceeding the threshold $\{z_k^{(j,l)} \mid |z_k^{(j,l)}| > \tau, j = 1, \dots, N_r, l = 1, \dots, N_\theta\}$ constitute the measurement set used for tracking at current scan. Herein the amplitude measurement $|z_k^{(j,l)}|$ is the a in Equation (13).

4.2.2. Amplitude Likelihood of the Spread Unit in Complex Gaussian Background

In the situation of Swerling type I fluctuation, amplitudes of target and clutter are Rayleigh distributed. The pdf of the clutter amplitude [16] is given by

$$c_a(a) = \frac{2a}{\sigma_n} \exp\left(-\frac{a^2}{\sigma_n}\right), \quad a > 0 \quad (39)$$

where σ_n is the covariance of the clutter amplitude. On the other hand, the pdf of the target amplitude is given by

$$g_a(A_k) = \frac{2A_k}{\sigma_t} \exp\left(-\frac{A_k^2}{\sigma_t}\right), \quad A_k > 0 \quad (40)$$

where σ_t is the covariance of the target amplitude. Then according to Equation (37) the amplitude of the target spread unit is

$$B_k = |A_k \exp(j\varphi_k)h| = A_k h \quad (41)$$

By parameter transformation of Equation (40), its pdf can be obtained as follows

$$g_a(B_k) = \frac{2B_k}{h^2\sigma_t} \exp\left(-\frac{B_k^2}{h^2\sigma_t}\right), \quad B_k > 0 \quad (42)$$

Therefore, the pdf of measured amplitude of target plus clutter a is the Rayleigh distribution, i.e.,

$$g_a(a|x_k) = \frac{2a}{\sigma_n + h^2\sigma_t} \exp\left(-\frac{a^2}{\sigma_n + h^2\sigma_t}\right), \quad a > 0 \quad (43)$$

Let $h = 1$ in Equation (43). We can get the amplitude pdf of the target that produces amplitude spread as

$$g_a(a|x_k) = \frac{2a}{\sigma_n + \sigma_t} \exp\left(-\frac{a^2}{\sigma_n + \sigma_t}\right), \quad a > 0 \quad (44)$$

If the amplitude of one unit and its several neighboring cells is beyond the detection threshold due to the spread, the point of this unit can be extracted as the central measurement for tracking by using the points clustering. On the other hand, the AI of the neighboring cells is retained to construct the united likelihood ratio for distinguishing target from clutter.

In Section 3.1, we use d to represent the ratio of target power to clutter power, i.e., $d = \sigma_t/\sigma_n$. To simplify the calculation, we can get $\sigma_n = 1$ and $d = \sigma_t$ by normalizing the clutter power [17]. Now substituting σ_n and d into Equations (39) and (44), the simplified amplitude likelihood for clutter and target respectively is represented as

$$c_a(a) = 2a \exp(-a^2), \quad a > 0 \quad (45)$$

$$g_a(a|d) = \frac{2a}{1+d} \exp\left(-\frac{a^2}{1+d}\right), \quad a > 0 \quad (46)$$

And the simplified amplitude likelihood for neighboring cells is

$$g_a(a|d, h) = \frac{2a}{1+h^2d} \exp\left(-\frac{a^2}{1+h^2d}\right) \quad (47)$$

Substituting Equation (45) into Equations (17) and (19), we obtain the probability of false alarm and the normalized amplitude likelihood for clutter as

$$p_{FA}^{\tau} = \exp(-\tau^2) \quad (48a)$$

$$c_a^\tau(a) = 2a \exp(\tau^2 - a^2), \quad a > \tau \tag{48b}$$

where τ is the detection threshold calculated by Equation (17) based on the specified false alarm probability. Similarly, substituting Equation (46) into Equations (16) and (18), we obtain the probability of detection and the normalized amplitude likelihood for target as

$$p_D^\tau = \exp\left(-\frac{\tau^2}{1+d}\right) \tag{49a}$$

$$g_a^\tau(a|d) = \frac{2a}{1+d} \exp\left(\frac{\tau^2 - a^2}{1+d}\right), \quad a > \tau \tag{49b}$$

4.2.3. GLMB-AI-UL

The single target measurement likelihood including amplitude likelihood in the GLMB-AI update process is given by Equation (30). Now we rewrite the amplitude likelihood in the form of united amplitude likelihood of neighboring cells. In order to distinguish it from symbols in Equation (30), the improved amplitude likelihood is expressed as

$$\widetilde{g}_a^\tau(a_{\theta(\ell)}|d, \ell) = \widetilde{g}_a^\tau(|z_k^{j\ell}|_{\theta(\ell)}|d, \ell) = g_a^\tau(|z_k^{j\ell}|_{\theta(\ell)}|d, \ell) \times \prod_{\substack{(j_m, l_n) \in (N(j), M(l)) \\ (j_m, l_n) \neq (j, \ell)}} g_a(|z_k^{(j_m, l_n)}|_{\theta(\ell)}|d, \ell, h), a_{\theta(\ell)} > \tau \tag{50}$$

where $N(j)$ and $M(l)$ are range bins and azimuth bins adjacent to the range-azimuth cell where $z_k^{j\ell}$ exists, and (j_m, l_n) is any range-azimuth cell affected by the amplitude spread of $z_k^{j\ell}$. Similarly, the improved amplitude likelihood of the clutter is expressed as

$$\widetilde{c}_a^\tau(a_{\theta(\ell)}) = c_a^\tau(a_{\theta(\ell)}) \prod_{\substack{(j_m, l_n) \in (N(j), M(l)) \\ (j_m, l_n) \neq (j, \ell)}} c_a(|z_k^{(j_m, l_n)}|_{\theta(\ell)}|), \quad a_{\theta(\ell)} > \tau \tag{51}$$

Substituting both Equations (50) and (51) into Equations (30) and (32), the algorithm GLMB-AI-UL can be obtained. The update operation of GLMB-AI-UL is summarized in Table 1 via pseudo code.

Table 1. Update of the GLMB-AI-UL.

input: $\left\{ \left(I^{(h)}, \zeta^{(h)}, w^{(h)}, p^{(h)}, T^{(h)} \right) \right\}_{h=1}^H, Z$ output: $\left\{ \left(I^{(h,j)}, \zeta^{(h,j)}, w^{(h,j)}, p^{(h,j)} \right) \right\}_{(h,j)=(1,1)}^{(H,T^{(h)})}$
for $h = 1 : H$ $C_Z^{(h)} := C_Z^{(I^{(h)}, \zeta^{(h)})}$ according to Equations (32), (50) and (51) $\left\{ \theta^{(h,j)} \right\}_{j=1}^{T^{(h)}} := \text{ranked assignment} \left(Z, I^{(h)}, C_Z^{(h)}, T^{(h)} \right)$ for $j = 1 : T^{(h)}$ $\eta_Z^{(h,j)} := \eta_Z^{(\zeta^{(h)}, \theta^{(h,j)})}$ according to Equation (28) $p^{(h,j)} := p^{(\zeta^{(h)}, \theta^{(h,j)})}(\cdot Z)$ according to Equation (29) $w^{(h,j)} := w^{(h)} \left[\eta_Z^{(h,j)} \right]^{I^{(h)}}$ $I^{(h,j)} := I^{(h)}$ $\zeta^{(h,j)} := \left(\zeta^{(h)}, \theta^{(h,j)} \right)$ end end normalize weights $\left\{ w^{(h,j)} \right\}_{(h,j)=(1,1)}^{(H,T^{(h)})}$

It should be noted that the united likelihood ratio of its nearest neighboring cells may be comparable to that of the cell with only clutter when there is a non-spread target or no target in a cell. The reason is that all the surrounding cells are clutter. In this case, the tracking performance of GLMB-AI-UL is the same as that of GLMB-AI. On the contrary, when there is a spread target, not only clutter but also the target are involved in the nearest neighboring unit. Therefore, a greater likelihood ratio would be obtained by using GLMB-AI-UL, and the association weight is improved.

5. Simulations

In this paper, the performance of GLMB-AI-UL is compared with that of other two algorithms, i.e., GLMB-AI and δ -GLMB (simplified as GLMB), by simulation data in three SCR scenarios. Table 2 shows the specific parameter settings.

Table 2. Scenario parameters.

Scenario	SCR	Mean Number of Clutter Before Detection λ	False Alarm Probability P_{fa}
1	15 dB	800	0.1
2	10 dB	800	0.1
3	8 dB	800	0.1

Let the size of observation area be $[-600, 600] \text{ m} \times [-600, 600] \text{ m}$, and the observation period be 100 frames in all. In scenarios 1 and 2, there are 12 moving targets following the nearly-constant-velocity (NCV) model, and their trajectories are shown in Figure 2; while in scenario 3, there are 9 targets with some of them following the coordinated turn (CT) model, and their trajectories are shown in Figure 3. In both figures the circles denote the starting points of the target trajectories while the triangles denote corresponding endpoints, and each target has different appearance and disappearance time. Hence the target number is varying in each frame. Set the clutter number before detection to be Poisson distributed with the mean value of 800. It can be seen that after detection it is also Poisson distributed with the mean value of about 80. For GLMB without AI, the survival probability is set to be 0.9 and the detection probability to be 0.95. For GLMB-AI-UL and GLMB-AI, the survival probability is set to be 0.9, and the detection probability can be calculated by Equations (16) and (17) according to the SCR and false alarm probability.

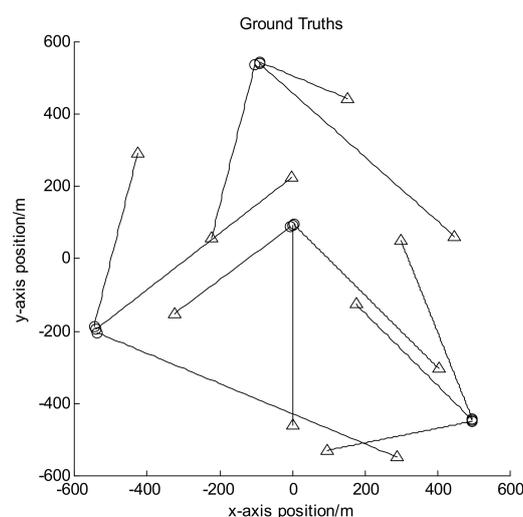


Figure 2. Target trajectories for scenario 1 and 2.

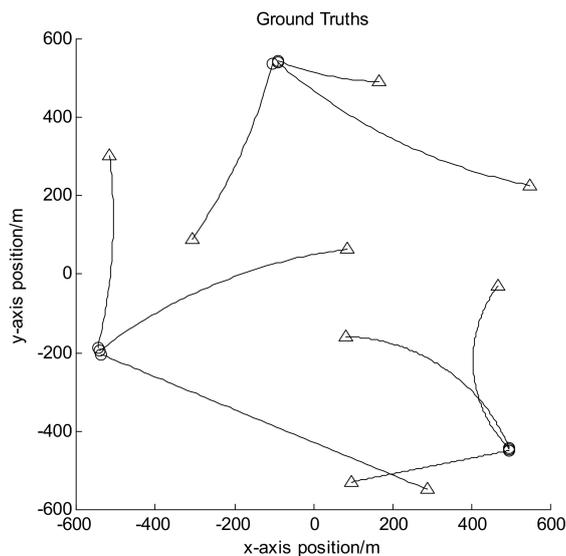


Figure 3. Target trajectories for scenario 3.

To evaluate the performance of GLMB-AI-UL, all the targets are simulated to spread their amplitude to neighboring cells. Due to the Rayleigh amplitude distribution, each target has different amplitude, and accordingly its spread scope is distinct as well. For the convenience of calculation, parameters of the point spread function in Equation (33) is set to be $L_r = L_b = R = B = 1$. When calculating AI-UL with Equations (50) and (51), only the neighbor cells next to any central measurement are considered. The OSPA metric [31] is used to evaluate the performance of three multi-target tracking algorithms mentioned above, which is defined as follows

$$\bar{d}_p^c(X, Z) := \left(\frac{1}{n} \left(\min_{\pi \in \Pi} \sum_{i=1}^m d^{(c)}(x_i, z_i)^p + c^p(n - m) \right) \right)^{\frac{1}{p}},$$

where we set $p = 1$ and $c = 100$ in simulations. In each scenario, we make 20 Monte Carlo runs for the three algorithms, and their performance are compared and analyzed by calculating the mean error.

5.1. Results of Scenario 1

Figure 4 shows the filtering output of the three algorithms in a single operation, where Figure 4a shows the true and estimated positions in x coordinate, while Figure 4b shows the true and estimated positions in y coordinate. As can be seen that the GLMB produces some missed tracks and false tracks, while the two filters using amplitude information, i.e., the GLMB-AI and GLMB-AI-UL give much more accurate estimates.

Figure 5 shows the OSPA miss distance and Figure 6 shows the comparison of cardinality estimates, and both are averaged over 20 trials. We can see from Figure 5a that the state estimation performance of GLMB-AI-UL and GLMB-AI are similar to that of the standard GLMB which only uses target dynamic information, and even at some times the GLMB performs slightly better than the GLMB-AI-UL and GLMB-AI. A possible reason for this result is that the cardinality error is ignored when calculating the location error of OSPA, which has been further discussed in [27]. GLMB may generate some false tracks due to a poorer association performance, and these tracks may be close to the actual ones of targets in space, which results in a smaller estimation error. However, in terms of the cardinality estimation performance shown in Figures 5b and 6, GLMB-AI-UL and GLMB-AI are significantly better than GLMB. This is due to the exploitation of amplitude information for calculating the measurement to track association weight in the first two algorithms. In this scenario, the cardinality estimation

performance of GLMB-AI-UL is almost same as that of GLMB-AI. The most possible reason for the similarity is that the SCR is high enough for GLMB-AI to distinguish target from clutter.

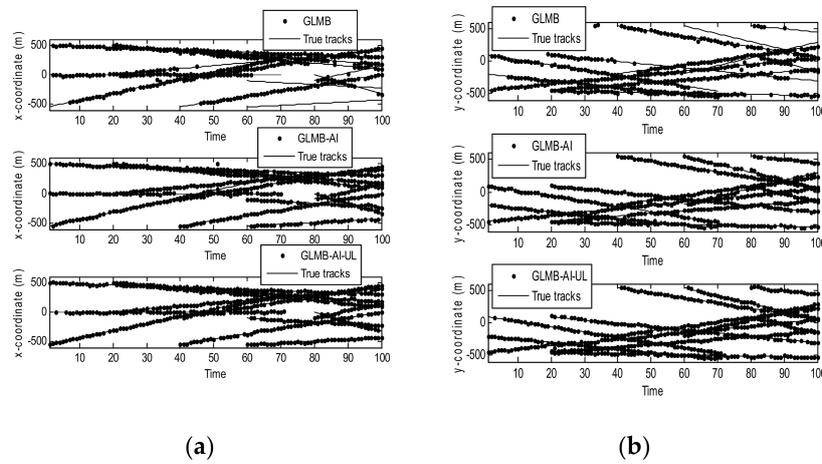


Figure 4. Estimates and tracks for scenario 1. (a) in x coordinate; (b) in y coordinate.

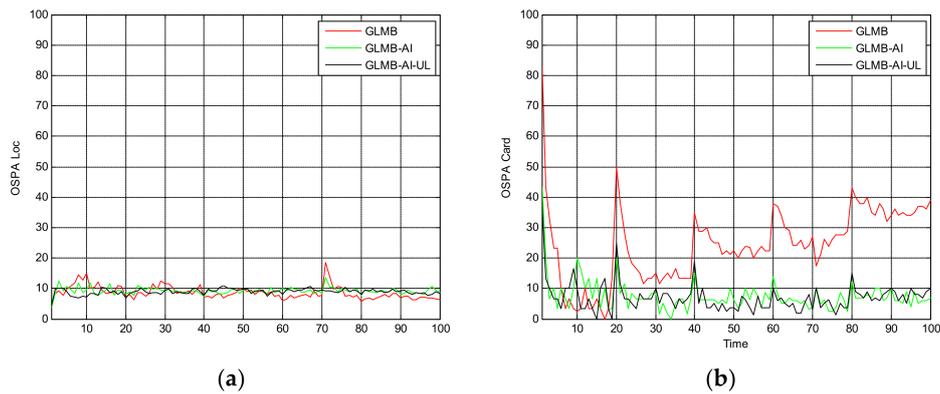


Figure 5. Average OSPA distance for scenario 1. (a) time average OSPA location distance; (b) time average OSPA cardinality distance.

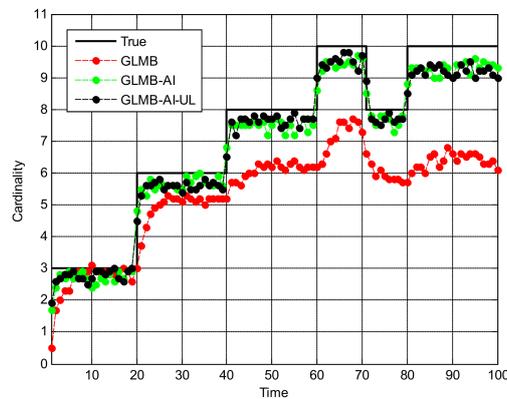


Figure 6. Cardinality estimates for scenario 1.

5.2. Results of Scenario 2

Figure 7 shows the filtering output of the three algorithms in a single run in x coordinate and y coordinate respectively. Compared with Figure 3, it can be seen that with the decrease of SCR, the state estimation performance of the three algorithms is significantly reduced. In particular, the GLMB generates more missed tracks and false tracks, and the GLMB-AI also produces some missed tracks and false tracks. Although the GLMB-AI-UL lost some plots, it still gives the most accurate state estimation, because by taking advantage of the energy of the surrounding units, a more efficient likelihood accumulation is realized.

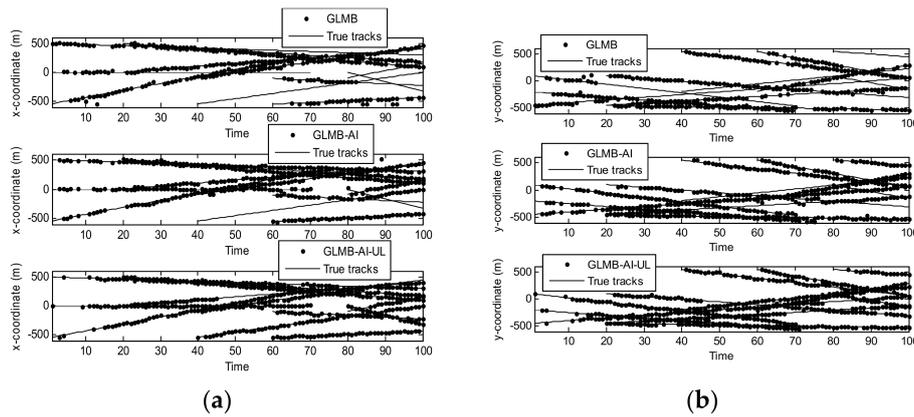


Figure 7. Estimates and tracks for scenario 2. (a) in x coordinate; (b) in y coordinate.

Figures 8 and 9 are the comparison of averaged tracking results in scenario 2. As can be seen from Figure 8a, the state estimation performance of the three algorithms is quite similar to that of scenario 1, and the superiority of GLMB is more obvious. This is because when the SCR is lower, GLMB may produce more false tracks, which can be closer to the target. Figures 6 and 8b show that the performance of cardinality estimation of GLMB-AI-UL and GLMB-AI is still significantly better than that of GLMB. However, different from scenario 1, the performance of GLMB-AI-UL is obviously better than that of GLMB-AI. The potential reason is that it is difficult to distinguish the target from clutter only by the amplitude characteristics of a single unit when the SCR is low. Nevertheless, it is possible to improve the differentiation ability by using the combined likelihood ratio of multiple resolution cells. In scenarios 1 and 2, we can also see that when the SCR is low, the deterioration degree of GLMB is the largest, and followed by GLMB-AI, which uses only single unit amplitude information, whereas GLMB-AI-UL with AI of multiple units is the most robust to strong clutter.

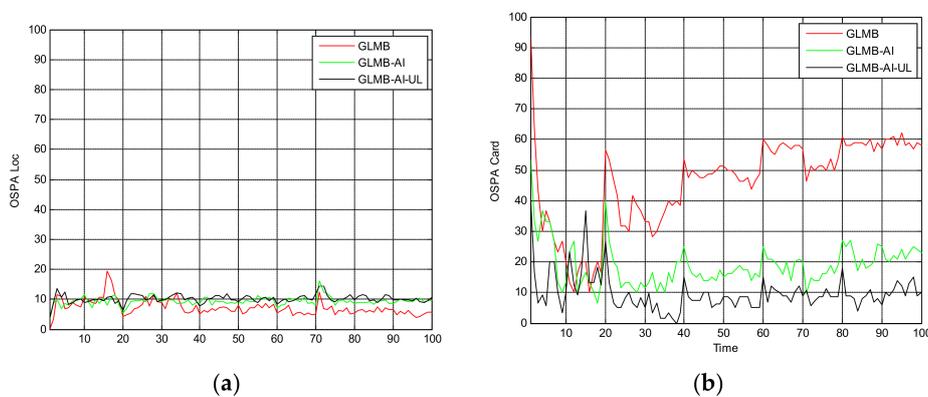


Figure 8. Average OSPA distance for scenario 2. (a) time average OSPA location distance; (b) time average OSPA cardinality distance.

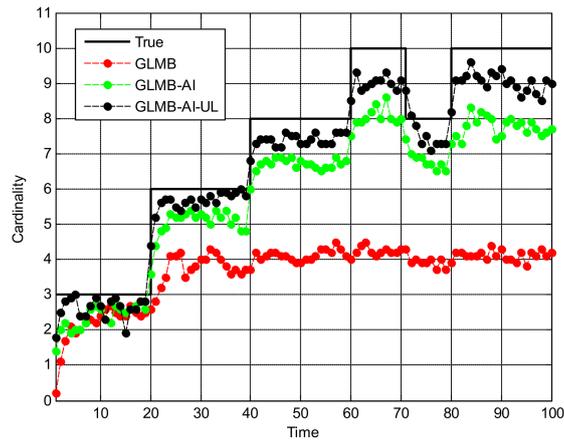


Figure 9. Cardinality estimates for scenario 2.

5.3. Results of Scenario 3

Figures 10–12 are the comparison of tracking results in scenario 3. Unlike the previous two scenarios, the motion of targets in this scenario is no longer linear, but rather maneuvering, and the SCR is set with a very low value of 8 dB.

From Figure 10 we can see that in this scenario, the tracking performance of the three algorithms is significantly reduced, and many missed tracks and false tracks are generated, but the GLMB-AI-UL still outperforms the other two algorithms, which proves the validity of the proposed method once again. From Figures 11b and 12 we can see that except for the initial 40 frames, the cardinality estimation performance of the GLMB-AI-UL is always the best in the rest of the filtering time.

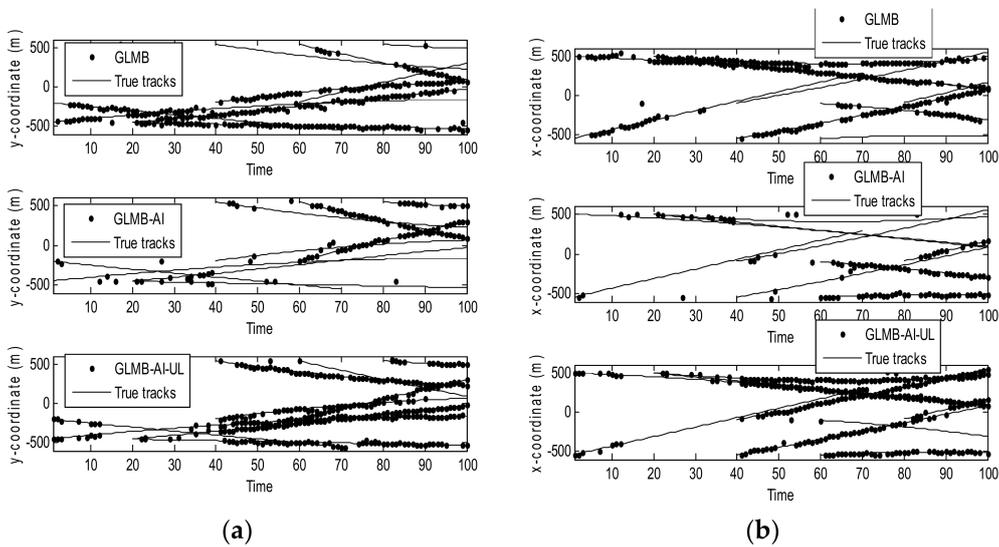


Figure 10. Estimates and tracks for scenario 3. (a) in x coordinate; (b) in y coordinate.

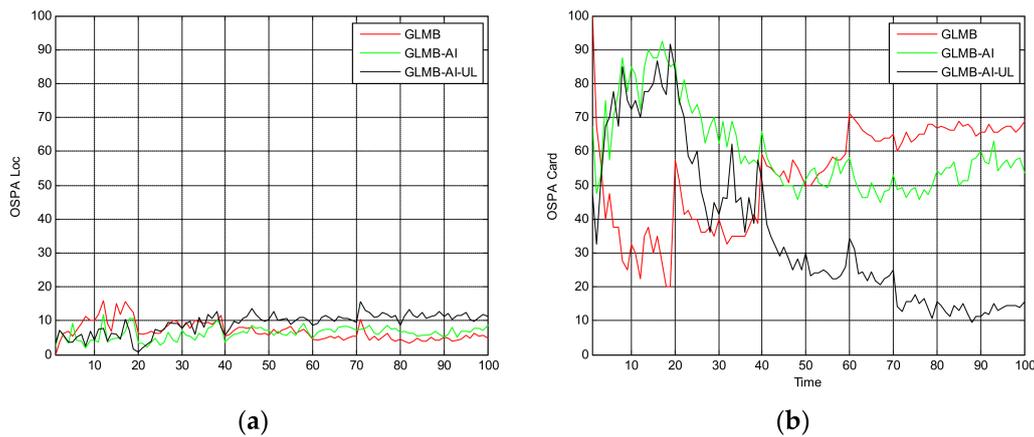


Figure 11. Average OSPA distance for scenario 3. (a) time average OSPA location distance; (b) time average OSPA cardinality distance.

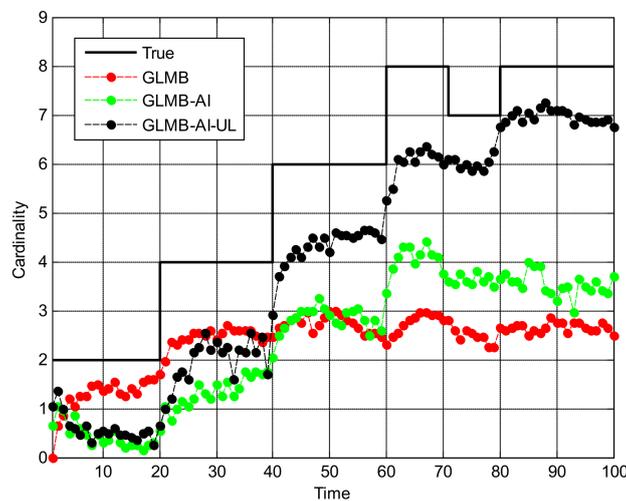


Figure 12. Cardinality estimates for scenario 3.

Table 3 is the average running time of each algorithm in the three scenarios. As can be seen from this table, in scenario 1, when the SCR is relatively high, the filtering performance of GLMB is good, and its calculation speed is the highest. This is because the calculation of amplitude information results in an increase in computational cost. However, when the SCR was reduced, in scenario 2, the GLMB's running time is only slightly reduced, but the elapsed time of the latter two algorithms with amplitude information is greatly reduced. This is because as the SCR decreases, the number of target-generated measurements decreases, and the utilization of amplitude information enhances the computational efficiency of the measurement-track assignment matrix. When the SCR is further reduced, the running time of the proposed GLMB-AI-UL continues decreasing, but that of the GLMB and GLMB-AI increase a lot, and the GLMB-AI-UL outperforms the other two algorithms in terms of computational cost. A reasonable explanation for this phenomenon is that, in this scenario, the SCR is such low that only few target-generated measurements exceed the threshold. This will reduce the computational cost on one hand, but on the other hand, the low SCR leads to a poorer computational efficiency of the measurement-track assignment matrix for the GLMB and GLMB-AI. Especially, the exploitation of amplitude information only of a single unit is less effective, which results in a poor performance for GLMB-AI, even worse than that of GLMB without using amplitude information. However, using the amplitude information of multiple adjacent units can evidently improve the performance. Thus, the proposed method is of great value in practical radar applications, especially when the SCR is very low.

Table 3. Running time averaged over 20 trials.

Scenario	GLMB	GLMB-AI	GLMB-AI-UL
1	300.6724 s	403.6110 s	433.2928 s
2	293.0126 s	343.4141 s	355.5296 s
3	340.1094 s	379.4045 s	333.7122 s

The simulations of this paper are carried out with Matlab, and the proposed algorithm seems time consuming. However, if the procedure is written in C language, the running time will be greatly reduced. Therefore, this algorithm is suitable for real-time radar applications, particularly for low SCR environments.

6. Conclusions

This paper proposes a new δ -GLMB filter using the united amplitude likelihood ratio of neighboring cells. It can deal with radar target echoes from multiple range-azimuth cells in the background of complex Gaussian clutter. The filter needs to retain the amplitude information of a few cells around the central measurement in the plots centroid step, and then the information is exploited to construct a united likelihood ratio. By combining this likelihood ratio into the δ -GLMB update process and measurement-track assignment cost matrix, the ability of δ -GLMB to distinguish targets from clutter can be improved. Compared with δ -GLMB using amplitude information of single cells, the proposed algorithm can obtain more accurate target's state and number estimation in low SCR environment. It is known that δ -GLMB not only usually has better tracking performance than PHD and Multi-Bernoulli filters, but also outputs target tracks. Therefore, compared with existing PHD and Multi-Bernoulli filters with the amplitude information of single cells, the proposed algorithm could obtain more accurate target's state and number estimation as well as the output of target tracks.

Acknowledgments: This study was co-supported by the National Natural Science Foundation of China (61471019 and U1633122).

Author Contributions: In this study, Jinping Sun and Chao Liu conceived and designed this novel algorithm and simulation experiment; Peng Lei and Yaolong Qi provided the advices in details; Chao Liu wrote this paper; Peng Lei modified English errors.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Lian, F.; Zhang, G.H.; Duan, Z.S.; Han, C.Z. Multi-target joint detection and estimation error bound for the sensor with clutter and missed detection. *Sensors* **2016**, *16*, 169. [[CrossRef](#)] [[PubMed](#)]
- Si, W.J.; Wang, L.W.; Qu, Z.Y. Multi-target tracking using an improved Gaussian mixture CPHD Filter. *Sensors* **2016**, *16*, 1964. [[CrossRef](#)] [[PubMed](#)]
- Bar-Shalom, Y.; Fortmann, T. *Tracking and Data Association*; Academic Press: Boston, MA, USA, 1988.
- Blackman, S. Multiple hypothesis tracking for multiple target tracking. *IEEE Trans. Aerosp. Electron. Syst.* **2004**, *19*, 5–18. [[CrossRef](#)]
- Mahler, R. *Statistical Multisource–Multitarget Information Fusion*; Artech House Publishers: Boston, MA, USA, 2007.
- Mahler, R. Multitarget Bayes filtering via first-order multi-target moments. *IEEE Trans. Aerosp. Electron. Syst.* **2003**, *39*, 1152–1178. [[CrossRef](#)]
- Mahler, R. PHD filters of higher order in target number. *IEEE Trans. Aerosp. Electron. Syst.* **2007**, *43*, 1523–1543. [[CrossRef](#)]
- Vo, B.T.; Vo, B.N.; Cantoni, A. The cardinality balanced multi-target multi-Bernoulli filter and its implementations. *IEEE Trans. Signal. Process.* **2009**, *57*, 409–423.
- Vo, B.T.; Vo, B.N. Labeled random finite sets and multi-object conjugate priors. *IEEE Trans. Signal. Process.* **2013**, *61*, 3460–3475. [[CrossRef](#)]

10. Vo, B.N.; Vo, B.T.; Phung, D. Labeled random finite sets and the Bayes multi-target tracking filter. *IEEE Trans. Signal. Process.* **2014**, *62*, 6554–6567. [[CrossRef](#)]
11. Reuter, S.; Vo, B.T.; Vo, B.N.; Dietmayer, K. The labeled multi-Bernoulli filter. *IEEE Trans. Signal. Process.* **2014**, *62*, 3246–3260.
12. Boers, Y.; Driessen, J.N. Multitarget particle filter track before detect application. *IEE Proc. Radar Sonar Navig.* **2004**, *151*, 351–357. [[CrossRef](#)]
13. Punithakumar, K.; Kirubarajan, T.; Sinha, A. A sequential Monte Carlo probability hypothesis density algorithm for multitarget track-before-detect. In *Signal and Data Processing of Small Targets 2005*; International Society for Optics and Photonics: Bellingham, WA, USA, 2005; Volume 5913, pp. 587–594.
14. Lepoutre, A.; Rabaste, O.; Gland, F.L. Multitarget likelihood computation for track-before-detect applications with amplitude fluctuations of type Swerling 0, 1, and 3. *IEEE Trans. Aerosp. Electron. Syst.* **2016**, *52*, 1089–1106. [[CrossRef](#)]
15. Aprile, A.; Grossi, E.; Lops, M.; Venturino, L. Track-before-detect for sea clutter rejection tests with real data. *IEEE Trans. Aerosp. Electron. Syst.* **2016**, *52*, 1035–1045. [[CrossRef](#)]
16. Jiang, H.; Yi, W.; Kirubarajan, T.; Kong, L.; Yang, X. Multiframe radar detection of fluctuating targets using phase information. *IEEE Trans. Aerosp. Electron. Syst.* **2017**, *53*, 736–749. [[CrossRef](#)]
17. Lerro, D.; Bar-Shalom, Y. Automated tracking with target amplitude information. In Proceedings of the IEEE American Control Conference, San Diego, CA, USA, 23–25 May 1990; pp. 2875–2880.
18. Lerro, D.; Bar-Shalom, Y. Interacting multiple model tracking with target amplitude feature. *IEEE Trans. Aerosp. Electron. Syst.* **1997**, *29*, 536–540. [[CrossRef](#)]
19. Cai, J.; Sinha, A.; Kirubarajan, T. EM-ML algorithm for track initialization using possibly noninformative data. *IEEE Trans. Aerosp. Electron. Syst.* **2005**, *41*, 1030–1048.
20. Ehrman, L.M.; Burton, C.; Blair, W.D. Using target RCS to aid measurement-to-track association in multi-target tracking. In Proceedings of the IEEE Southeastern Symposium on System Theory, Cookeville, TN, USA, 5–7 March 2006; pp. 89–93.
21. Ehrman, L.M.; Blair, W.D. RCS-aided tracking: Does it always improve data association. In Proceedings of the 2007 IEEE Radar Conference, Boston, MA, USA, 17–20 April 2007; pp. 186–191.
22. Ehrman, L.M.; Blair, W.D. Comparison of methods for using target RCS to improve measurement-to-track association in multi-target tracking. In Proceedings of the IEEE Conference Information Fusion, Florence, Italy, 10–13 July 2006; pp. 1–8.
23. Ehrman, L.M.; Blair, W.D. Using Target RCS when Tracking Multiple Rayleigh Targets. *IEEE Trans. Aerosp. Electron. Syst.* **2010**, *46*, 701–716. [[CrossRef](#)]
24. Brekke, E.F.; Hallingstad, O.; Glattetre, J.H. Performance of PDAF-based Tracking Methods in Heavy-Tailed Clutter. In Proceedings of the 12th International Conference on Information Fusion, Seattle, WA, USA, 6–9 July 2009; pp. 6–9.
25. Brekke, E.F.; Hallingstad, O.; Glattetre, J.H. The Modified Riccati Equation for Amplitude-Aided Target Tracking in Heavy-Tailed Clutter. *IEEE Trans. Aerosp. Electron. Syst.* **2011**, *47*, 2874–2886. [[CrossRef](#)]
26. Clark, D.; Ristic, B.; Vo, B.N.; Vo, B.T. PHD Filtering with target amplitude feature. In Proceedings of the International Conference on Information Fusion, Cologne, Germany, 30 June–3 July 2008; pp. 1–7.
27. Clark, D.; Ristic, B.; Vo, B.N.; Vo, B.T. Bayesian multi-object filtering with amplitude feature likelihood for unknown object SNR. *IEEE Trans. Signal Process.* **2010**, *58*, 26–37. [[CrossRef](#)]
28. Li, S.; Kong, L.; Yi, W.; Wang, B. PHD filter with amplitude information in Weibull clutter. In Proceedings of the IEEE Radar Conference, Ottawa, ON, Canada, 29 April–3 May 2013; pp. 1–6.
29. Liu, J.; Kong, L.; Yi, W. Integrated clutter-knowledge-based tracking algorithm in complex ground environment. In Proceedings of the IET International Radar Conference, Hangzhou, China, 14–16 October 2015; pp. 1–5.
30. Yuan, C.S.; Wang, J.; Sun, J.P.; Sun, Z.S.; Bi, Y.X. A multi-Bernoulli filtering algorithm using amplitude information. *J. Elect. Inf. Tech.* **2016**, *38*, 464–471.
31. Schuhmacher, D.; Vo, B.T.; Vo, B.N. A consistent metric for performance evaluation of multi object Filters. *IEEE Trans. Signal Process.* **2008**, *56*, 3447–3457. [[CrossRef](#)]

