



Particle Swarm Optimization-Based Joint Integrated Probabilistic Data Association Filter for Multi-Target Tracking

Yun Zhu¹, Hao Wang¹, Shuang Liang^{2,*}, Mahendra Mallick³, Tianyu Guo¹ and Jilei Liao¹

¹School of Computer Science, Shaanxi Normal University, Xi'an 710119, China

²State Key Laboratory of Integrated Service Networks, School of Telecommunications Engineering, Xidian University, Xi'an 710071, China

³Independent Consultant, Anacortes, WA 98221, United States

Abstract

The joint integrated probabilistic data association (JIPDA) filter is effective for automatic multi-target tracking in cluttered environments. However, it is well-known that when targets are closely spaced, the JIPDA filter encounters the track coalescence problem, leading to inaccurate state estimations. This paper proposes a novel particle swarm optimization-based JIPDA (PSO-JIPDA) algorithm, which improves the state estimation accuracy by optimizing the posterior probability density, effectively addressing the information fusion challenge in multi-target tracking scenarios with closely spaced targets. The trace of the covariance matrix of the posterior density serves as the objective function for the optimization problem. Minimizing the trace enhances the accuracy of target state estimation by refining

the posterior density. Specifically, all possible permutations of the targets are enumerated, with each permutation assigned a unique index. These indices are mapped to association hypothesis events within a probabilistic fusion framework, where each mapping corresponds to a particle in the PSO algorithm. The particles are initialized by stochastically assigning indices to hypothesis events, forming the initial swarm. During iterations, the particles dynamically adjust their positions and velocities based on individual and global optimal solutions, guided by the trace minimization objective. Experimental results demonstrate that the PSO-JIPDA algorithm significantly improves the accuracy of Gaussian approximation and makes notable progress in addressing the track coalescence problem.

Keywords: multi-target tracking, particle swarm optimization, probabilistic fusion, joint integrated probabilistic data association.



Academic Editor:

Yunfei Guo

Submitted: 30 January 2025

Accepted: 26 May 2025

Published: 28 June 2025

Vol. 2, No. 2, 2025.

10.62762/CJIF.2025.506643

*Corresponding author:

✉ Shuang Liang

sliang@xidian.edu.cn

Citation

Zhu, Y., Wang, H., Liang, S., Mallick, M., Guo, T., & Liao, J. (2025). Particle Swarm Optimization-Based Joint Integrated Probabilistic Data Association Filter for Multi-Target Tracking. *Chinese Journal of Information Fusion*, 2(2), 182–193.



© 2025 by the Authors. Published by Institute of Central Computation and Knowledge. This is an open access article under the CC BY license (<https://creativecommons.org/licenses/by/4.0/>).

1 Introduction

Multi-Target Tracking (MTT) serves as a critical technique in radar, sonar and other surveillance systems [1, 2], with recent advances spanning deep learning-based trajectory analysis [3], multi-sensor fusion frameworks [4], and graph-based feature representation [5]. The main challenge of MTT lies in effectively resolving the data association problem between targets and collected measurements. There are two traditional MTT methods: the multiple hypothesis tracking (MHT) filter [6, 7] and the joint probabilistic data association (JPDA) filter [8]. While MHT theoretically achieves near-optimal solutions by maintaining multiple association hypotheses over time, its computational complexity grows exponentially with the number of targets and measurements due to the combinatorial hypothesis space. This makes it challenging for real-time applications without aggressive pruning and approximation techniques. In comparison, JPDA reduces computational burden through probabilistic weighting of association hypotheses. However, it suffers from two inherent limitations: (1) it assumes a fixed and known number of targets during tracking; and (2) it is prone to track coalescence in dense target environments.

The joint integrated probabilistic data association (JIPDA) filter [9] plays a critical role in information fusion for multi-target tracking by integrating measurement data from different sensors and targets. It extends the concept of target existence probability, enabling joint estimation of data associations and target existence states while efficiently handling the fusion of probabilistic information. JIPDA evaluates feasible measurement-to-track association hypotheses, fuses their probabilistic weights, and updates both target existence probabilities and state estimates. By dynamically terminating tracks with low existence probabilities (e.g., below a threshold), this probabilistic framework allows JIPDA to adapt to scenarios where the number of targets varies over time. However, the JIPDA filter remains susceptible to the track coalescence problem. This occurs when targets are in close proximity, leading to overlapping tracking gates and ambiguous data associations. As a result, measurements from different targets may be incorrectly weighted, biasing state estimates. Prolonged proximity can cause merged tracks that cannot be distinguished, even after targets are separated. This limitation is particularly problematic in dense target scenarios and restricts the practical effectiveness of JIPDA.

To address the issue of track coalescence caused by overlapping validation gates and the resultant association uncertainty problem, we propose an adaptive particle swarm optimization-based joint integrated probabilistic data association (PSO-JIPDA) filter. By optimizing the posterior association probabilities, the proposed method overcomes the tracks coalescence problem. This geometric constraint induces two critical issues: (1) substantial ambiguity in measurement-to-track association; and (2) pronounced multimodal characteristics in the posterior density function. Specifically, during state estimation, measurements originating from distinct targets may be erroneously associated with the same track, leading to a marked increase in the trace of the state estimation covariance matrix (established research has demonstrated this metric as an effective quantitative measure of posterior distribution multimodality [10, 11]). For enhanced accuracy in multi-target state estimation, the PSO approach is used to optimize the posterior probability density and minimize the trace of the covariance matrix. PSO operates through the following sequential implementation steps: First, in the initialization phase, particle positions are randomly generated within the solution space. These position vectors, whose dimensionality equals the problem's free parameters, represent possible solutions. Initial positions and velocities are randomly generated for all particles to ensure uniform distribution across the search space. Position coordinates are typically bounded within predefined ranges to comprehensively cover feasible regions. Each particle evaluates its fitness value through a predefined objective function, quantifying the quality of its current solution. During iterative updates, particles dynamically adjust their states by tracking two critical extremum values: the personal best, denoting the optimal solution discovered by an individual particle, and the global best, representing the best solution identified by the entire swarm. The particle state update mechanism combines three components: inertial momentum, cognitive learning (based on individual historical performance), and social learning (guided by swarm collaboration). A velocity vector adjusts the particle's movement direction and step size, while position updates translate these velocities into spatial displacements. Additionally, we have implemented several improvements to the PSO algorithm, including optimization of the inertia weight adjustment strategy and enhancements to the particle position update method, among others. These modifications will be

elaborated in detail in Section 4. To summarize, our contributions are threefold as follows.

- The tracking gate overlap issue has been improved. When the target spacing is small, the algorithm is capable of mitigating trajectory overlaps to some extent, thereby minimizing the potential bias in trajectory estimation.
- The posterior density has been optimized. Experimental results demonstrate that the posterior density obtained through the JIPDA update, measured by the similarity to a single Gaussian density, yields superior performance in the PSO-JIPDA filter compared to other filters, even in scenarios where the target states are unknown.
- The accuracy of target state estimation has been effectively enhanced. Based on the optimization of the classical PSO algorithm, the performance of the PSO-JIPDA filter is exceptional in both known and unknown initial state scenarios, demonstrating its effectiveness using the OSPA multi-target miss distance metric.

The remainder of the paper is structured as follows. Section 2 introduces the proposed PSO-JIPDA algorithm, detailing the utilization of PSO for posterior density optimization and providing concrete implementation examples. Section 3 validates the effectiveness of our method through experiments in two distinct scenarios. Section 4 draws conclusions.

2 Related Work

To address track coalescence in multi-target tracking, various data association techniques have been innovated. The exact nearest neighbor JPDA (ENNJPDA) filter [12] represents an early approach, reducing computational complexity by aggressively pruning all but the highest-probability association hypothesis. However, this method presents significant susceptibility to both false alarms and missed detections, as it discards potentially useful alternative hypotheses. To address these limitations, the JPDA* variant [13] was introduced, refining the hypothesis selection process through selective pruning: it retains the most consistent data association hypothesis to compute measurement-to-target assignment probabilities, balancing computational efficiency with robustness. Building on this foundation, The JIPDA* filter [14] integrates the JIPDA framework with JPDA*, creating a more resilient solution for track coalescence scenarios. JIPDA* operates by prioritizing

a single optimal association hypothesis from the set of possible track-measurement associations, pruning alternative hypotheses that involve identical tracks and measurements but differ in assignments. By eliminating competing hypotheses that could induce trajectory merging or misassignment, JIPDA* maintains coherent state estimation even in cluttered environments where traditional JPDA-based methods often fail to resolve ambiguous interactions. However, the discarded hypotheses might contain residual data association information that could potentially enhance the accuracy of tracking, despite being suboptimal in the current selection framework. To address this problem, our recent research [15] introduces the evolutionary optimization-based joint integrated probabilistic data association (EOJIPDA) method. EOJIPDA operates by probabilistically integrating all feasible association hypotheses through posterior density optimization. This mechanism preserves the probabilistic significance of primary and secondary hypotheses events, ensuring comprehensive utilization of measurement information. Nevertheless, the integration of evolutionary computation inherently increases the algorithm's computational load.

Aforementioned methods rely on data association mechanisms, and there is another class of methods based on the finite set statistics (FISST) [16]. Notable implementations include the probability hypothesis density (PHD) filter [17], its cardinality-enhanced variant CPHD [18], and the multi-Bernoulli (MB) formulation [19]. Subsequent refinements have produced labeled variants such as labeled multi-Bernoulli [20] and generalized label multi-Bernoulli filters [21, 22], all using the random finite set (RFS) theory for systematic uncertainty modeling. The PHD and CPHD techniques use moment approximations to circumvent the computational intractability inherent in full Bayesian multi-target posterior density calculations. Different from this paradigm, the MB approach parameterizes multi-Bernoulli RFS components rather than maintaining explicit density representations. Labeled extensions, such as LMB and GLMB, enable target identity preservation. These FISST-based methods avoid explicit combinatorial association by implicitly modeling multi-target states as sets, offering a principled mathematical framework for MTT.

3 Background

3.1 Target Motion and Observation Modeling

Target dynamics are modeled under the nearly constant velocity assumption in this paper. The state vector of target t at discrete time step k is represented by $\mathbf{x}_{t,k}$, with following state transition equation:

$$\mathbf{x}_{k+1} = F\mathbf{x}_k + u_k \quad (1)$$

where F denotes the state transition matrix, and $u_k \sim \mathcal{N}(0, Q)$ is Gaussian process noise with zero mean and covariance Q . The measurement z_k associated with target t follows a linear observation model:

$$z_k = H\mathbf{x}_k + w_k \quad (2)$$

where H is the observation matrix, and $w_k \sim \mathcal{N}(0, R)$ represents zero-mean Gaussian measurement noise with covariance R .

The target-originated detections occur with probability P_D , while clutter measurements follow a spatial Poisson distribution. The average number of clutter measurements per time step is $r = \lambda|\text{FoV}|$, where λ is the clutter intensity and $|\text{FoV}|$ denotes the sensor's field-of-view coverage. The validated measurement set at time k is defined as:

$$Z_k = \{z_k^1, z_k^2, \dots, z_k^{m_k}\} \quad (3)$$

where m_k is the total number of measurements. The historical measurement sequence up to time k is denoted as

$$Z_{1:k} = \{Z_1, Z_2, \dots, Z_k\} \quad (4)$$

3.2 Joint Integrated Probabilistic Data Association Filter

JIPDA generates association hypotheses by probabilistically assigning validated measurements to tracks. Each monitored target maintains a dynamic parameter tuple encapsulating its kinematic state vector $\mathbf{x}_{t,k}$, covariance matrix $P_{t,k}$, and existence probability $P_s(t, k)$ at discrete time index k . Consequently, the characterization of a track t can be achieved through the parameter set $\{P_s(t, k), \mathcal{N}(\mathbf{x}_{t,k}, P_{t,k})\}$. When advancing to time step $k+1$, the key steps for updating this parameter set are described below.

First, in the prediction stage, the Kalman filter is used to compute the predicted target state $\mathbf{x}_{t,k+1|k}$ and covariance $P_{t,k+1|k}$ as follows:

$$\begin{aligned} \mathbf{x}_{t,k+1|k} &= F\mathbf{x}_{t,k} \\ P_{t,k+1|k} &= FP_{t,k}F^T + Q_k \end{aligned} \quad (5)$$

where F denotes the state transition matrix, and Q_k denotes process noise covariance matrix at time k .

The target existence probability propagates according to a first-order Markov chain model,

$$P_s(t, k+1 | k) = p_{11}P_s(t, k) + p_{21}(1 - P_s(t, k)) \quad (6)$$

where P_{11} and P_{21} are the Markov chain coefficients.

Second, analytically derived tracking gates are generated for each target to assist measurement association. As comprehensively discussed in [23], various gating methodologies exist for measurement validation, with the ellipsoidal gating approach [24] being particularly prevalent due to its optimal statistical properties under Gaussian assumptions. At time step k , for track t , the ellipsoidal gate is defined as

$$\mathcal{G}_t^{k+1} = \{z_{k+1}^i \in \mathbb{R}^{d_z} | \Delta_{t,j}^{k+1} \leq \gamma\} \quad (7)$$

$$\begin{aligned} \Delta_{t,j}^{k+1} &= \left(z_{k+1}^j - H_k \mathbf{x}_{t,k+1|k} \right)^T \\ &\times (S_{t,k+1})^{-1} \left(z_{k+1}^j - H_k \mathbf{x}_{t,k+1|k} \right) \end{aligned} \quad (8)$$

where z_{k+1}^i denotes the i th received measurement, $S_{i,k+1} = HP_{i,k+1|k}H^T + R$ is the innovation covariance, γ is a threshold, which leads to a gating probability P_w , and d_z is the measurement dimension. This gating procedure ensures that only statistically compatible measurements are considered for potential association, thereby reducing computational complexity through effective measurement pruning.

Third, association hypotheses are generated by establishing probabilistic correspondences between validated measurements and existing tracks. The posterior probability of an event θ_h is computed as

$$\begin{aligned} P(\theta_h) &= C^{-1} \prod_{t \in T_0^h} (1 - P_D P_W P_s(t, k+1 | k)) \\ &\times \prod_{t \in T_1^h} \left(P_D P_W P_s(t, k+1 | k) \frac{p(z_{k+1}^j | \theta_h, Z_{1:k})}{\lambda} \right) \end{aligned} \quad (9)$$

where C is the normalization constant, P_D denotes probability of detection, T_0^h denotes the set of tracks associated without measurement, and T_1^h corresponds to the set of tracks associated with exactly one measurement.

The existence probability of track t is updated under

the association hypothesis event θ_h as

$$P_s(t | h, k+1) = \left(1 - \sum_{i=1}^{m_k} \xi_i^{t|h}\right) P_{s,0}(t | h, k+1) + \sum_{i=1}^{m_k} \xi_i^{t|h} P_{s,i}(t | h, k+1) \quad (10)$$

where $\xi_i^{t|h}$ is a binary indicator variable, i.e.,

$$\begin{cases} \xi_i^{t|h} = 1 & \text{if track } t \text{ is associated with measurement } z_k^i \\ \xi_i^{t|h} = 0 & \text{otherwise} \end{cases} \quad (11)$$

and $P_{s,0}(t|h, k+1)$ denotes the probability of track t being unassociated with any measurement and $P_{s,i}(t|h, k+1)$ denotes the probability of track t being associated with measurement z_{k+1}^i , which are computed as

$$P_{s,0}(t | h, k+1) = \frac{(1 - P_D^t P_W^t) P_s(t, k+1 | k)}{1 - P_D^t P_W^t P_s(t, k+1 | k)} P(\theta_h) \\ P_{s,i}(t | h, k+1) = P(\theta_h) \quad (12)$$

The state of track t under hypothesis θ_h is computed as:

$$\mathbf{x}_{t|h,k} = \left(1 - \sum_{i=1}^{m_k} \xi_i^{t|h}\right) \mathbf{x}_{t,k|k-1} + \sum_{i=1}^{m_k} \xi_i^{t|h} \mathbf{x}_{t,k}^i \quad (13)$$

where $\mathbf{x}_{t,k}^i$ is the state updated with measurement z_k^i . The error covariance is updated as:

$$P_{t|h,k+1} = \left(1 - \sum_{i=1}^{m_k} \xi_i^{t|h}\right) P_{t,k+1|k} + \sum_{i=1}^{m_k} \xi_i^{t|h} \left[P_{t,k+1|k} - K_{t,k+1} \mathbf{S}_{t,k+1} (K_{t,k+1})^T \right] \quad (14)$$

where $K_{t,k+1} = P_{t,k+1|k} H^T (S_{t,k+1})^{-1}$ denotes the Kalman filter gain.

Ultimately, the track's existence probability, state estimate and covariance are approximated by

$$P_s(t, k+1) = \sum_{h=1}^{N_H} P_s(t | h, k+1) \quad (15)$$

$$\mathbf{x}_{t,k} = \frac{1}{P_s(t, k+1)} \sum_{h=1}^{N_H} P_s(t | h, k+1) \mathbf{x}_{t|h,k} \quad (16)$$

$$P_{t,k} = \frac{1}{P_s(t, k+1)} \sum_{h=1}^{N_H} P_s(t | h, k+1) \times \left[P_{t|h,k+1} + \mathbf{x}_{t|h,k} (\mathbf{x}_{t|h,k})^T - \mathbf{x}_{t,k} (\mathbf{x}_{t,k})^T \right] \quad (17)$$

where N_H denotes the number of all association hypothesis events.

4 Methodology

4.1 Objective Function Proposal

Automatic target tracking systems account for time-varying number of targets. Following [16], we employ random finite sets (RFS) to characterize the inherently stochastic and disordered nature of target states. Following the work in [10], which demonstrates that posterior probability densities can be transformed within their corresponding RFS families, this paper introduces an innovative approach. Specifically, it proposes a method to improve the multi-target tracking performance by optimizing ordered posterior density functions within the RFS family framework.

Theoretical analysis and experimental studies [11] have shown that minimizing the error covariance matrix can overcome the overlapping tracking gate problem. In practical applications, we employ the covariance matrix trace [10] as the optimization criterion for posterior density refinement, formulated as

$$C_k = \sum_{t=1}^{n_k} \text{trace}(P_{t,k}) \quad (18)$$

where n_k denotes the number of targets and the function $\text{trace}(\cdot)$ represents the trace of the matrix.

4.2 Particle Swarm Optimization of the Posterior Density

For posterior density optimization, target permutations across distinct association hypotheses are reconfigured to minimize the cost function defined in (18). The optimization is effectuated by means of PSO [5], with the principal procedures of the PSO-JIPDA approach as follows.

Step 1: PSO setup for the optimization of the posterior density.

For PSO implementation, we adopt (18) as the global fitness function. The solution space comprises every permutation of the current target indices and is defined as:

$$\mathcal{P} = \{\Pi_h | h = 1, \dots, N_H\} \quad (19)$$

where Π_h denotes the permutation set corresponding to the h th association hypothesis and Π_h has cardinality $|\Pi_h| = n_k!$. We initialize a PSO swarm with N_p particles. At time k , the position x_p^k of each particle p represents a candidate solution vector, where each element corresponds to the index (from 1 to n_k) of a specific target permutation chosen for a particular hypothesis. The goal is to find the vector of permutation indices (one index per hypothesis) that optimizes a global fitness function.

Step 2: PSO Iterative Optimization.

Execute the PSO algorithm for defined generations and update the inertia weight w at time k . In each generation, for a particle, we perform the following steps.

First, compute the inertia weight. The original linear decreasing inertia weight strategy was enhanced with a nonlinear adaptive approach. The dynamic inertia weight is calculated based on the iteration progress and fitness values, as follows

$$w = \begin{cases} w_{\min} + \frac{(w_{\max} - w_{\min}) \times (f_{\text{current}} - f_{\min})}{(f_{\text{mean}} - f_{\min})} & \text{if } f_{\text{current}} < f_{\text{mean}} \\ w_{\max} & \text{otherwise} \end{cases} \quad (20)$$

where $w_{\max}$ and $w_{\min}$ represent the maximum and minimum inertia weights, respectively; $f(\cdot)$ denotes fitness function; f_{current} , f_{mean} , and $f_{\min}$ denote the current fitness function, the mean fitness function and the minimum fitness function, respectively.

Second, update each particle's velocity v_p^k using the inertia weight w which controls how much the particle's previous velocity affects its current movement direction. Then, update each particle's position vector x_p^k using the obtained velocity, guiding the search towards better permutation index vectors. The mathematical description of the update process is as follows:

$$\nu_p^{k+1} = w\nu_p^k + c_1 r_1 (p_{\text{best}} - x_p^k) + c_2 r_2 (g_{\text{best}} - x_p^k) \quad (21)$$

$$x_p^{k+1} = x_p^k + \nu_p^{k+1} \quad (22)$$

where c_1 is the individual learning factor determining the influence of the particle's own best position p_{best} on its velocity update; c_2 is the social learning factor reflecting the impact of the global best position g_{best} found by the entire swarm on the particle's velocity update; $r_1 \sim \mathcal{U}(0, 1)$ is the individual stochastic factor; $r_2 \sim \mathcal{U}(0, 1)$ is the social stochastic factor; p_{best} is the best position that the particle has reached so

far in the search process; g_{best} is the best position found by the entire particle swarm so far. Note that the boundary handling is used internally as particle positions correspond to discrete indices.

Third, evaluate the fitness of each particle using each particle's position vector x_p^k and update each particle's personal best p_{best} position and swarm's global best g_{best} position based on fitness values, as follows:

$$\begin{aligned} p_{\text{best},p}^k &= x_p^k, \quad f_{\text{best},p} = f(x_p^k), \quad \text{if } f(x_p^k) < f(p_{\text{best},p}^k) \\ g_{\text{best}} &= p_{\text{best},p}^k, \quad f(g_{\text{best}}) = f(p_{\text{best},p}^k), \\ &\quad \text{if } f(p_{\text{best},p}^k) < f(g_{\text{best}}) \end{aligned} \quad (23)$$

Step 3: Apply Optimal Permutations.

After PSO converges or reaches defined generations, retrieve the global best position vector g_{best} . For each hypothesis in g_{best} , reorder the corresponding hypothesis-specific index. Subsequently, the standard JPDA hypothesis-weighted marginalization (15)-(17) performed to estimate the target state estimate $x_{t,k}$, the covariance matrix $P_{t,k}$, and the existence probability $P_s(t, k)$. Figure 1 shows the flowchart of the proposed PSO-JIPDA algorithm. The probability density optimization procedure is enclosed in the dotted green box, while the iterative PSO update is shown in the dotted orange box. The "Termination" module determines whether the optimization process should end. If not terminated, the process returns to the "iterative PSO update" stage. If terminated, it outputs the global best position.

4.3 Illustrative Example

To illustrate the PSO-JIPDA algorithm's operational process, we consider a simplified one-dimensional case with two Gaussian-distributed targets generating two measurements. Assume two association hypothesis events ψ_1 and ψ_2 are established, with density of target t under hypothesis ψ_h is modeled as $p^{t|h}(x) = \{r^{t|h}, \mathcal{N}(x^{t|h}, p^{t|h})\}$. The initial posterior density is specified by

$$\begin{aligned} \psi_1 : \begin{cases} p^{1|1}(x) = \{0.1, \mathcal{N}(1, 1)\} \\ p^{2|1}(x) = \{0.4, \mathcal{N}(4, 1)\} \end{cases} \\ \psi_2 : \begin{cases} p^{1|2}(x) = \{0.5, \mathcal{N}(5, 1)\} \\ p^{2|2}(x) = \{0.5, \mathcal{N}(2.3, 1)\} \end{cases} \end{aligned} \quad (24)$$

Based on (15)-(17), the Gaussian-approximated probability density for each target is characterized as

$$\begin{aligned} p_1(x) &= \{0.6, \mathcal{N}(3.5, 2.25)\} \\ p_2(x) &= \{0.9, \mathcal{N}(3.5, 2.80)\} \end{aligned} \quad (25)$$

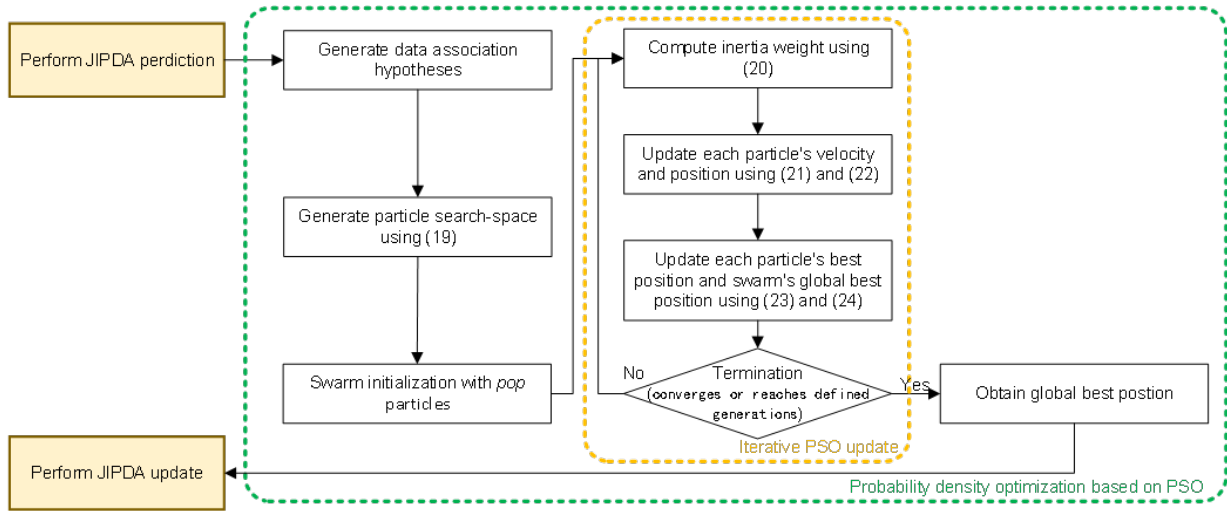


Figure 1. Flowchart of PSO-JIPDA algorithm.

Figure 2(a) and 2(b) present the posterior probability density of the joint track state and its Gaussian approximation, respectively. It can be observed that a considerable difference exists between these two densities, demonstrating the limited precision of the Gaussian approximation. In this case, the trace of the covariance matrix is

$$C_0 = 2.25 + 2.80 = 5.05 \quad (26)$$

which can be computed using (18).

The posterior probability density is then optimized using the particle swarm optimization (PSO) algorithm. Following five iterations, the final optimized result is derived as follows:

$$\psi_1 : \begin{cases} p^{11}(x) = \{0.4, N(1, 1)\} \\ p^{21}(x) = \{0.1, N(5, 1)\} \end{cases}$$

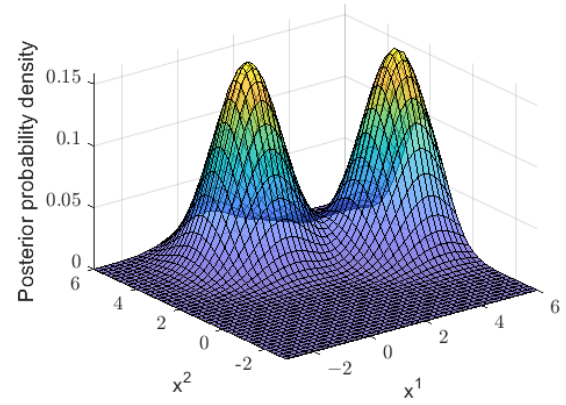
derived as follows

$$\psi_2 : \begin{cases} p^{12}(x) = \{0.5, N(2.3, 1)\} \\ p^{22}(x) = \{0.5, N(4, 1)\} \end{cases} \quad (27)$$

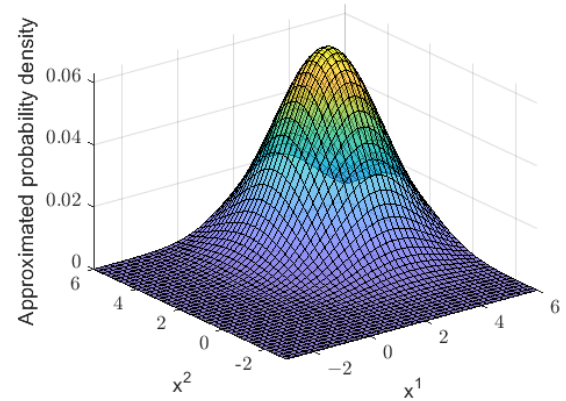
Using (15)-(17), the approximated probability density for each track is given as

$$\begin{aligned} p_1(x) &= \{0.60, N(4.44, 1.25)\} \\ p_2(x) &= \{0.90, N(2.08, 1.23)\} \end{aligned} \quad (28)$$

Figure 3(a) displays the initial posterior probability density, while its approximated single Gaussian density is presented in Figure 3(b). Obviously, Figures 3(a) and 3(b) demonstrate a strong correlation, whereas Figures 2(a) and 2(b) exhibit a notable



(a)



(b)

Figure 2. (a) Posterior and (b) approximated probability densities at iteration 0.

divergence. In this case, the trace of the covariance matrix is

$$C_0 = 1.25 + 1.23 = 2.48 \quad (29)$$

After PSO optimization, the trace decreases by 50.89% relative to the initial trace (26), demonstrating a significant improvement in Gaussian approximation and state estimation precision. These statistical analyses exhibit strong concordance with theoretical expectations.

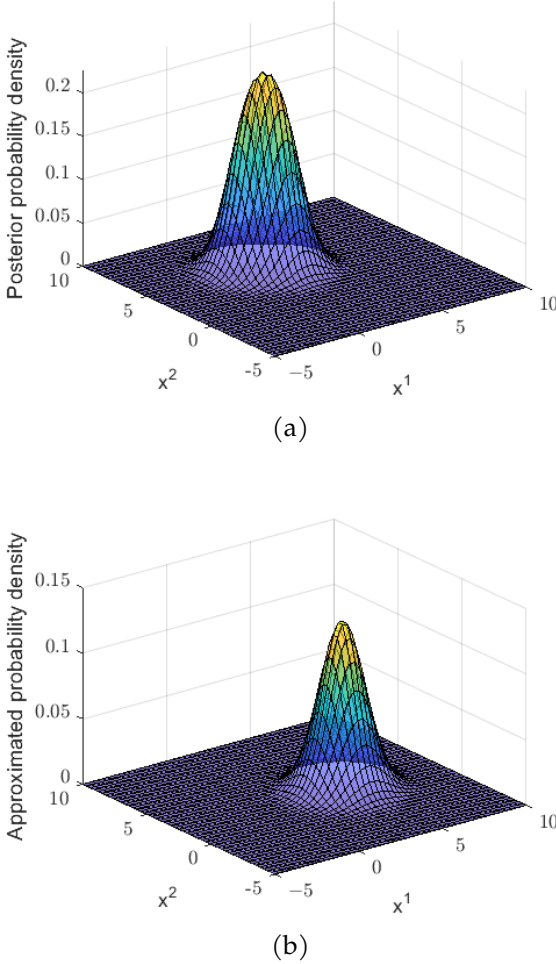


Figure 3. (a) Posterior and (b) approximated probability densities at iteration 5.

5 Experiments

We assess the PSO-JIPDA filter's efficacy in a complex 2D multi-target tracking environment, employing kinematic modeling of moving targets with

$$\mathbf{x}_k = [p_x^k, \nu_x^k, p_y^k, \nu_y^k]^T \quad (30)$$

where $[p_x^k, p_y^k]^T$ denotes the target position and $[\nu_x^k, \nu_y^k]^T$ is the target velocity. The transition matrix of the motion model is

$$F = I_2 \otimes \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix} \quad (31)$$

where $\otimes$ denotes the Kroneker product, I_2 represents a 2×2 identity matrix, and T_s indicates sampling interval. The process noise u_k follows a Gaussian distribution with covariance matrix

$$Q_k = I_2 \otimes q \begin{bmatrix} \frac{T_s^3}{3} & \frac{T_s^2}{2} \\ \frac{T_s^2}{2} & T_s \end{bmatrix} \quad (32)$$

where q is a tuning parameter. A sensor collects target measurements within the monitoring region in both experimental scenarios. For simplicity, it is assumed that the sensor only captures target position information. Therefore, the observation matrix of the measurement model is

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (33)$$

The covariance of the Gaussian measurement noise $\mathbf{w}_k$ is

$$R_k = \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix} \quad (34)$$

where σ_x is the standard deviations of the measurement noise in x coordinate and σ_y is the standard deviations of the measurement noise in y coordinates, respectively.

The sensor operates at a fixed sampling interval of $T_s = 1$ s, with a detection probability of $P_D = 0.92$. A gating threshold of $\gamma = 9.21$ [25] is applied, corresponding to a two-dimensional gating probability of $P_W = 0.99$. The Markov chain model coefficients in (6) are the same as [9], with $p_{11} = 0.98$ and $p_{21} = 0$. Track management follows probabilistic rules: tracks are confirmed when the existence probability surpasses $P_C = 0.83$, while terminated when falling below $P_T = 0.0909$, following the criteria established in [2]. For the PSO algorithm, we use where $w_{\max} = 0.9$, $w_{\min} = 0.4$, $c_1 = 2$, and $c_2 = 2$. We employ the Optimal Sub-Pattern Assignment (OSPA) metric [26], a well-established performance measure in random finite set (RFS) based multi-target tracking [16, 19], to quantitatively evaluate tracking accuracy.

For two sets $X = \{x_1, \dots, x_n\}$ and $Y = \{y_1, \dots, y_m\}$ representing true and estimated states (assuming $n \geq m$), the OSPA distance is defined as [26]:

$$d_{p,c}^{\text{OSPA}}(X, Y) = \left[\frac{1}{n} \left(\min_{\pi \in \Pi_n} \sum_{i=1}^m \min \left(d^{(c)}(x_i - y_{\pi(i)})^p + c^p(n-m) \right) \right) \right]^{1/p} \quad (35)$$

where $d^{(c)}(x_i - y_{\pi(i)}) := \min(c, \|x - y\|)$, $c > 0$ is the cut-off parameter, $p \geq 1$ is an order parameter, and π is a permutation function in the set of permutations Π_n ; if $n < m$, $d_{p,c}^{\text{OSPA}}(X, Y) = d_{p,c}^{\text{OSPA}}(Y, X)$. All numerical experiments were conducted using MATLAB R2022b on a computing platform equipped with a 3.40 GHz CPU.

In the surveillance area, two targets follow deterministic trajectories [10, 11], [13] as described in Figure 4. The targets initially maintain parallel motion at close proximity and ultimately they separate. In Figure 4, the parameters are selected as: $\phi = \pi/3$, $d = 100$ m, and $l_1 = l_2 = 1000$ m. Both targets travel at a constant speed $v = 100$ m/s. The measurement noise is characterized by $\sigma_x = \sigma_y = 10$ m.

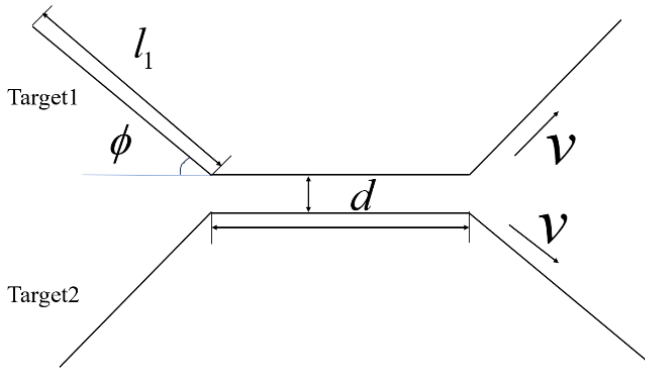
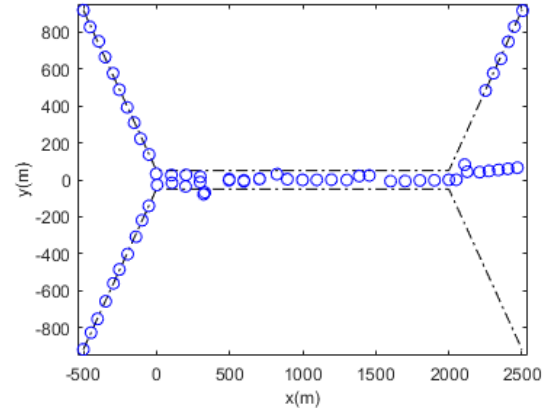


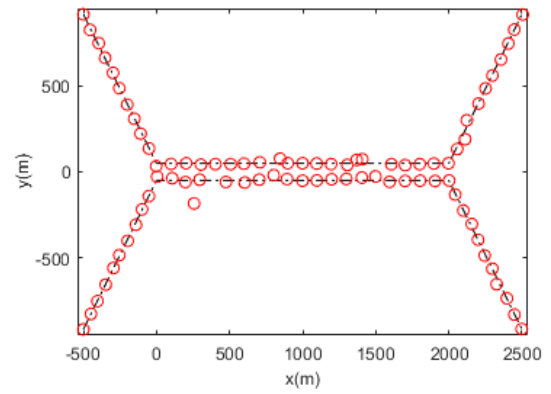
Figure 4. Simulated scenario 1.

Figure 5(a) and 5(b) present a single-run comparison of the estimated target trajectories obtained from the conventional JIPDA filter and the proposed PSO-JIPDA filter, respectively. When the two targets move in close proximity, the JIPDA filter exhibits severe track coalescence: the estimated trajectories completely overlap, despite their true tracks being distinct. Even after the targets are separated, the JIPDA filter fails to resolve their individual trajectories, leading to persistent estimation errors. In contrast, the PSO-JIPDA filter maintains highly accurate trajectory estimates which closely follow the true tracks. Notably, significantly reduced track coalescence occurs during the targets' proximity stage, and the filter successfully tracks their subsequent separation. This demonstrates that PSO-JIPDA effectively mitigates track coalescence through optimized posterior density handling, achieving superior state estimation.

To demonstrate the robustness of our proposed algorithm, we increased the noise to $\sigma_x = \sigma_y = 15$ m and reduced the distance to $d = 50$ m between the two target trajectories. Figure 6(a) clearly illustrates a



(a)



(b)

Figure 5. Estimated target trajectories for case 1. The true target trajectories are denoted with black dotted lines and the estimated target trajectories are denoted with dots: (a) JIPDA; (b) PSO-JIPDA.

pronounced track coalescence phenomenon, where the estimated positions of targets merge into a single track when the targets move in close proximity, significantly degrading the tracking performance. In contrast, Figure 6(b) demonstrates the superior capability of the PSO-JIPDA filter, which maintains high-precision state estimation for each individual target even under challenging conditions where targets are closely spaced. By effectively resolving the track coalescence issue, the PSO-JIPDA filter ensures accurate and reliable tracking performance, and these experimental results validate its enhanced robustness compared to the conventional method. The improved performance stems from the optimized posterior probability density enabled by the PSO-JIPDA framework, which successfully prevents track merging while preserving estimation accuracy.

Figure 7 presents a comparative analysis of the

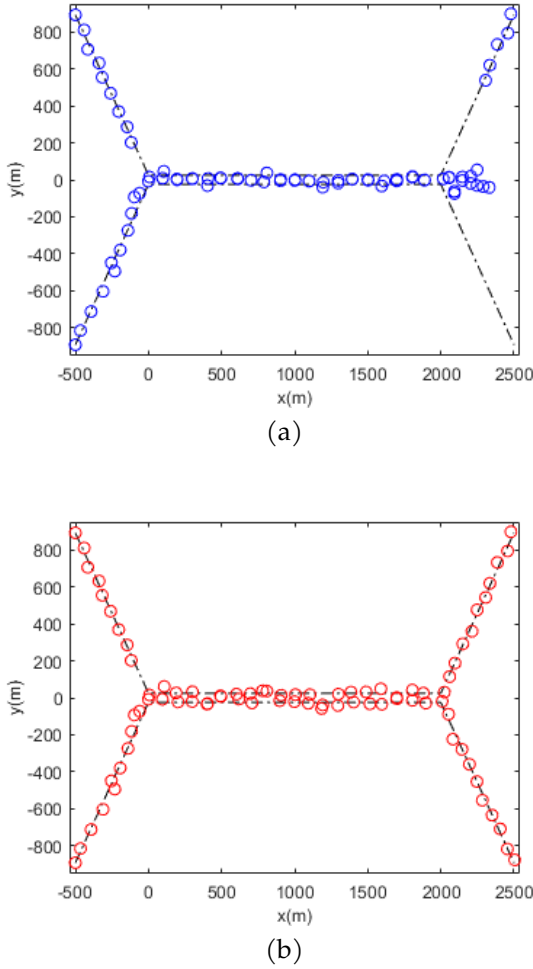


Figure 6. Estimated target trajectories for case 2. The true target trajectories are denoted with black dotted lines and the estimated target trajectories are denoted with dots: (a) JIPDA; (b) PSO-JIPDA.

average OSPA distance across 100 Monte Carlo experiments for the following algorithms: JIPDA, ENNJIPDA, JIPDA*, CBMeMber [19], EOJIPDA, and PSO-JIPDA. Experimental results indicate that the JIPDA filter exhibits significantly higher OSPA errors than comparative algorithms caused by the track coalescence problem. Notably, while the ENNJIPDA algorithm was designed to address coalescence, its performance remains highly sensitive to clutter interference and missed detection events. Compared to the JIPDA* filter, the EOJIPDA and PSO-JIPDA filters use the posterior density information more comprehensively and demonstrate superior tracking performance in terms of the OSPA error measure.

In Figure 8, the average running time of each Monte Carlo simulation for the JIPDA, ENNJIPDA, JIPDA*, CBMeMber, EOJIPDA, and PSO-JIPDA filters is presented, based on 100 Monte Carlo

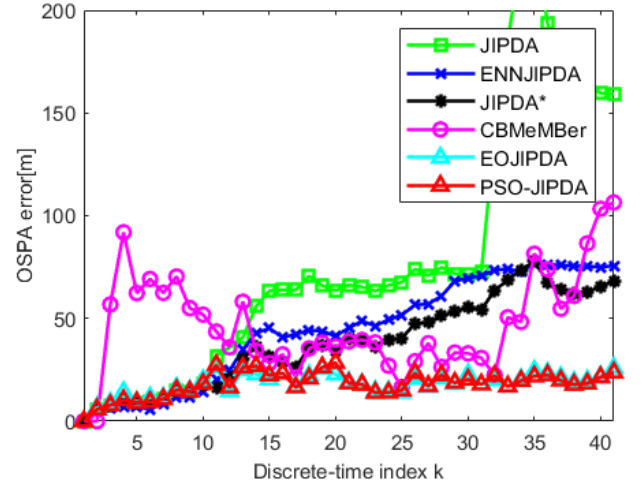


Figure 7. Average OSPA distances for scenario 1.

trials. Due to the optimization of the posterior probability density in the PSO-JIPDA algorithm, it incurs a higher computational cost compared to JIPDA, ENNJIPDA, JIPDA*, and CBMeMber. However, the PSO-JIPDA algorithm demonstrates a satisfactory trade-off between tracking performance and computational efficiency. These results suggest that, despite the increased computational time due to the posterior density optimization, the PSO-JIPDA algorithm achieves comparable tracking accuracy and improved runtime performance than the EOJIPDA algorithm, making it a more efficient choice.

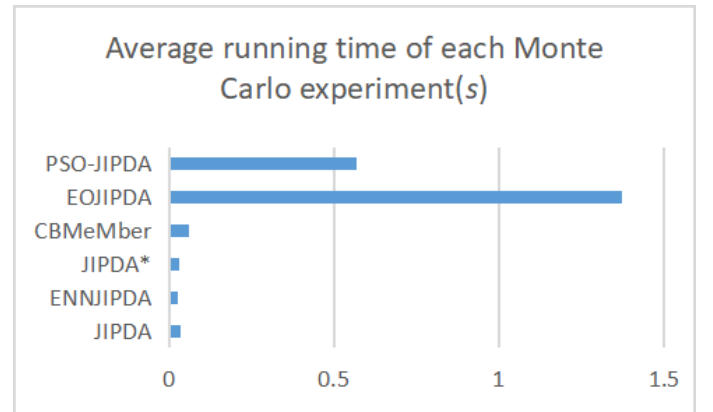


Figure 8. Running time comparison of all algorithms.

6 Conclusion

This paper proposes a novel PSO-JIPDA approach designed to effectively address the critical technical challenge of track coalescence prevalent in multi-target tracking systems operating in complex cluttered environments. Through rigorous analysis of limitations inherent in conventional JIPDA filters, this study introduces a novel optimization strategy. The

research implements the optimization of the posterior probability density in the classical JIPDA filter using an advanced particle swarm intelligence optimization approach, achieving enhancement in the tracking accuracy.

Future research directions include extending the PSO-JIPDA framework to accommodate nonlinear measurement models, optimizing swarm initialization strategies for real-time applications, and exploring hybrid optimization algorithms to further reduce computational overhead. Additionally, integrating adaptive parameter tuning mechanisms for the PSO coefficients may further enhance the robustness. This work provides a solution for intelligent swarm-based optimization in MTT systems, with potential applications in autonomous surveillance, robotics, and sensor networks.

Data Availability Statement

Data will be made available on request.

Funding

This work was supported in part by the National Natural Science Foundation of China under Grant 62306226, in part by the Key Research and Development Program of Shaanxi under Grant 2025CY-YBXM-074, and in part by the Fundamental Research Funds for the Central Universities under Grant GK202406006 and Grant XJSJ25002.

Conflicts of Interest

The authors declare no conflicts of interest.

Ethical Approval and Consent to Participate

Not applicable.

References

- [1] Dos Anjos, J. C., Gross, J. L., Matteussi, K. J., González, G. V., Leithardt, V. R., & Geyer, C. F. (2021). An algorithm to minimize energy consumption and elapsed time for IoT workloads in a hybrid architecture. *Sensors*, 21(9), 2914. [Crossref]
- [2] Zhang, Y., Li, Y., Li, S., Zeng, J., Wang, Y., & Yan, S. (2023). Multi-target tracking in underwater multistatic AUV networks with a robust Poisson Multi-Bernoulli filter. *Ocean Engineering*, 284, 115167. [Crossref]
- [3] Chai, R., Liu, D., Liu, T., Tsourdos, A., Xia, Y., & Chai, S. (2022). Deep learning-based trajectory planning and control for autonomous ground vehicle parking maneuver. *IEEE Transactions on Automation Science and Engineering*, 20(3), 1633-1647. [Crossref]
- [4] Liu, L., Ji, H., Zhang, W., & Liao, G. (2020). Multi-sensor fusion for multi-target tracking using measurement division. *IET Radar, Sonar & Navigation*, 14(9), 1451-1461. [Crossref]
- [5] Salvi, D., Waggoner, J., Temlyakov, A., & Wang, S. (2013, January). A graph-based algorithm for multi-target tracking with occlusion. In *2013 IEEE Workshop on Applications of Computer Vision (WACV)* (pp. 489-496). IEEE. [Crossref]
- [6] Reid, D. (1979). An algorithm for tracking multiple targets. *IEEE transactions on Automatic Control*, 24(6), 843-854. [Crossref]
- [7] Kurien, T. (1990). Issues in the design of practical multitarget tracking algorithms. *Multitarget-multisensor tracking: advanced applications*, 43-84. [Crossref]
- [8] Fortmann, T., Bar-Shalom, Y., & Scheffe, M. (1983). Sonar tracking of multiple targets using joint probabilistic data association. *IEEE journal of Oceanic Engineering*, 8(3), 173-184. [Crossref]
- [9] Musicki, D., & Evans, R. (2004). Joint integrated probabilistic data association: JIPDA. *IEEE transactions on Aerospace and Electronic Systems*, 40(3), 1093-1099. [Crossref]
- [10] Svensson, L., Svensson, D., Guerriero, M., & Willett, P. (2011). Set JPDA filter for multitarget tracking. *IEEE Transactions on Signal Processing*, 59(10), 4677-4691. [Crossref]
- [11] Zhu, Y., Wang, J., Liang, S., & Wang, J. (2019). Covariance control joint integrated probabilistic data association filter for multi-target tracking. *IET Radar, Sonar & Navigation*, 13(4), 584-592. [Crossref]
- [12] Fitzgerald, R. J. (1986, June). Development of practical PDA logic for multitarget tracking by microprocessor. In *1986 American Control Conference* (pp. 889-898). IEEE. [Crossref]
- [13] Blom, H. A., & Bloem, E. A. (2000). Probabilistic data association avoiding track coalescence. *IEEE Transactions on Automatic Control*, 45(2), 247-259. [Crossref]
- [14] Blom, H. A., Bloem, E. A., & Musicki, D. (2015). JIPDA: Automatic target tracking avoiding track coalescence. *IEEE Transactions on Aerospace and Electronic Systems*, 51(2), 962-974. [Crossref]
- [15] Liang, S., Zhu, Y., & Li, H. (2022). Evolutionary optimization based set joint integrated probabilistic data association filter. *Electronics*, 11(4), 582. [Crossref]
- [16] Mahler, R. (2007). *Statistical multisource-multitarget information fusion*. Artech.
- [17] Mahler, R. P. (2003). Multitarget Bayes filtering via first-order multitarget moments. *IEEE Transactions on Aerospace and Electronic systems*, 39(4), 1152-1178.

- [Crossref]
- [18] Mahler, R. (2007). PHD filters of higher order in target number. *IEEE Transactions on Aerospace and Electronic systems*, 43(4), 1523-1543. [Crossref]
 - [19] Vo, B. T., Vo, B. N., & Cantoni, A. (2008). The cardinality balanced multi-target multi-Bernoulli filter and its implementations. *IEEE Transactions on Signal Processing*, 57(2), 409-423. [Crossref]
 - [20] Reuter, S., Vo, B. T., Vo, B. N., & Dietmayer, K. (2014). The labeled multi-Bernoulli filter. *IEEE Transactions on Signal Processing*, 62(12), 3246-3260. [Crossref]
 - [21] Cao, C., & Zhao, Y. (2024). Multi-sensor multi-target tracking with generalized labeled multi-Bernoulli filter based on track-before-detect observation model using Gaussian belief propagation. *Digital Signal Processing*, 153, 104618. [Crossref]
 - [22] Wang, B., Yi, W., Li, S., Morelande, M. R., Kong, L., & Yang, X. (2015, July). Distributed multi-target tracking via generalized multi-Bernoulli random finite sets. In *2015 18th International Conference on Information Fusion (Fusion)* (pp. 253-261). IEEE.
 - [23] Blackman, S. S., & Popoli, R. (1999). Design and analysis of modern tracking systems. (No Title). [Crossref]
 - [24] Breidt, F. J., & Carriquiry, A. L. (2000). Highest density gates for target tracking. *IEEE transactions on aerospace and electronic systems*, 36(1), 47-55. [Crossref]
 - [25] Bar-Shalom, Y., Chang, K. C., & Blom, H. A. P. (1989, December). Automatic track formation in clutter with a recursive algorithm. In *Proceedings of the 28th IEEE Conference on Decision and Control*, (pp. 1402-1408). IEEE. [Crossref]
 - [26] Schuhmacher, D., Vo, B. T., & Vo, B. N. (2008). A consistent metric for performance evaluation of multi-object filters. *IEEE transactions on signal processing*, 56(8), 3447-3457. [Crossref]



Hao Wang received the B.S. degree in communication engineering from Qingdao University of Science and Technology, Qingdao, China, in 2023. He is currently pursuing the M.S. degree in computer science and technology at Shaanxi Normal University, Xi'an, China. (Email: wh5027@snnu.edu.cn)



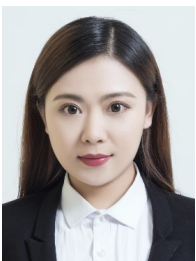
Shuang Liang received the B.S. degree in electronic information engineering and the Ph.D. degree in pattern recognition and intelligent systems from Xidian University, Xi'an, China, in 2012 and 2019, respectively. (Email: sliang@xidian.edu.cn)



Mahendra Mallick received the Ph.D. degree in quantum solid-state theory from the State University of New York, Albany, NY, USA, in 1981, and the M.S. degree in computer science from Johns Hopkins University, Baltimore, MD, USA, in 1987. (Email: mmallick.us@gmail.com)



Tianyu Guo is currently pursuing the B.S. degree in computer science and technology (innovation experimental class) at Shaanxi Normal University, Xi'an, China. (Email: guotianyu@snnu.edu.cn)



Yun Zhu received the B.S. degree in electronic information engineering and the Ph.D. degree in signal and information processing from Xidian University, Xi'an, China, in 2012 and 2019, respectively. (Email: yunzhu@snnu.edu.cn)



Jilei Liao is currently pursuing the B.S. degree in software engineering from Shaanxi Normal University, Xi'an, China. (Email: liaojilei@snnu.edu.cn)