International Journal of Applied Mathematics,
Volume 38 No. 11s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

HYBRID PSO–GA PARTICLE FILTERS WITH SORTED LOOKAHEAD WEIGHTING AND MINIMALISTIC RESAMPLING FOR TRACK-BEFORE-DETECT APPLICATION

Ramakrishna Gurajala[1*], Praveen B. Choppala,[2]

[1*]Research Scholar, Department. of E.C.E., AU college of Engineering, Andhra University, Visakhapatnam, India ramagurajala@gmail.com

[2]Adhoc Professor, Department. of E.C.E., AU college of Engineering, Andhra University. Visakhapatnam, India praveen.b.choppala@gmail.com

Abstract

In the Track Before Detection (TBD) concept, the target is observed in an environment characterized by a low signal-to-noise ratio prior to its identification as a target. TBD finds application in various fields such as air traffic control, maritime surveillance, underwater autonomous vehicle (UAV) tracking, and the detection of small objects in remote sensing. A fundamental technique employed in TBD is the particle filter (PF). In nonlinear and non-Gaussian environments, the PF serves as a prominent method for target detection. However, the standard Particle Filter (PF) algorithm experiences particle diversity loss due to degradation and resampling, which hinders the particle set from accurately reflecting the true state probability density function. The Particle Swarm Optimization (PSO) algorithm can alleviate particle degradation in PF; however, its performance is significantly affected by measurement noise variance and is susceptible to becoming trapped in local optima, which restricts its filtering accuracy. To tackle these challenges, a hybrid approach that integrates the Genetic Algorithm (GA) with PSO is proposed. By merging the rapid convergence of PSO with the robust global search capabilities of GA, this method enhances particle diversity, maintains the effectiveness of high-quality particles, and improves overall accuracy. GA utilizes global search operations, including selection, crossover, and mutation. In this paper, we propose two selection mechanisms: sorted weighting lookahead and minimalistic resampling schemes. The integration of the optimization algorithms PSO and GA with these selection mechanisms will enhance particle diversity, mitigate the issue of sample impoverishment, and improve the root mean square value. The tracking accuracy of the proposed methods is evaluated in this paper using two nonlinear models.

Keywords— ***Particle filter, K-means Clustering, Track before detect (TBD), RMSE.***

I. INTRODUCTION

One of the signal processing approach that track and detect dim targets is the track before detection approach. TBD is used in sonar, radar and image tracking applications. It takes raw sensor data and accumulate over multiple frames and track the target. The major approaches used for TBD is dynamic programming, particle filter and velocity filtering. The particle filter is used for tracking the target in nonlinear and nonGaussian environment.

Received: November 18, 2025

To estimate nonlinear and nonGaussian models the Bayesian state estimator is widely used. Particle filter is the one of the Bayesian state estimator. The PF performs recursive estimation to generate a set of weighted particles. By using these particles, the posterior approximation of the state space is obtained. This process is done by the PF in two stages. In the first stage, propagate the particles through the proposal distribution and update the weights of the particles. If the choice of proposal distribution is improper, only one particle dominates the distribution of weights. To mitigate this problem, in the second step resampling process is performed. In this process, the particles with high weights are replaced the low weight particles. This process significantly improves the accuracy of the posterior approximation.

The combination of Particle Swarm Optimization and Genetic Algorithm with the Particle Filter results in distinguished enhancements in the performance of particle filtering. PSO improves the filter's exploitation capability by directing particles towards regions of the state space with high likelihood, thus enabling faster convergence. On the other hand, GA provides strong exploration capabilities through genetic processes such as selection, crossover, and mutation, which aid in preserving particle diversity and avoiding early convergence. Consequently, the hybrid PSO–GA framework effectively balances global exploration with local exploitation, resulting in enhanced estimation accuracy and robustness in nonlinear and non-Gaussian systems. Moreover, the optimized distribution of particles decreases particle degeneracy and enables the filter to achieve reliable tracking performance with a reduced number of particles.

A sorted weighting lookahead particle filter [2] that is both computationally efficient and operationally precise. The fundamental concept is to do in advance future states by sampling particles and assigning weights based on incoming observations. This lookahead set is subsequently prearranged according to its weights. A select few particles with significant weights, which are located in areas of high relevance, are then deterministically chosen for forward propagation. This deterministic selection process eliminates the need for resampling, thereby significantly enhancing the method's speed. The Sorted Weighting Lookahead Particle Filter (SLAPF) enhances the selection of particles by assessing future likelihoods; however, it incurs significant computational overhead due to the processes of weight sorting and lookahead calculations, which restrict its applicability in real-time circumstances. Particle impoverishment remains a possibility, particularly in conditions of high measurement noise or when the number of particles is limited, potentially resulting in the loss of informative particles. The algorithm's performance is also influenced by the precision of the dynamic model, as any discrepancies may result in biased estimations. Furthermore, the effectiveness of the algorithm is depending upon the careful tuning of parameters, including particle count, likelihood thresholds, and noise variances.

Stochastic resampling methods, especially systematic and residual resampling, are the most frequently employed approaches in particle filtering. These techniques perform resampling on random samples—one per particle in order—enabling full interaction among all particles, preserving the total weight, and offering an unbiased estimate of the posterior distribution.

International Journal of Applied Mathematics,
Volume 38 No. 11s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

However, this sequential and detailed procedure leads to considerable Monte Carlo variance and increases computational complexity. Partial deterministic resampling methods seek to reduce computational costs, but a thorough assessment of their unbiasedness has not yet been conducted. In the minimalistic resampling [3] particle filter (MRPF) approach, the normalized weights are sorted, and a set of indices corresponding to low weights is determined. The particles with low weights are removed and replaced with copies of those with high weights. This approach minimizes particle communication while effectively protective the unbiasedness property in practice.

The Minimalistic Resampling Particle Filter (MRPF) reduces computational demands by selectively resampling only a portion of the particles. Although efficient, it risks discarding significant particles in high-noise or dynamic conditions, which can result in biased state estimates. Particle degeneracy is only partially addressed, allowing a few particles to dominate the weight distribution over time. The method is sensitive to parameter configurations, including thresholds, particle numbers, and noise variance, necessitating careful adjustment. Its effectiveness also diminishes in high-dimensional state spaces, where selective resampling may struggle to maintain acceptable diversity.

In order to overcome these limitations, this paper introduces a hybrid particle filtering framework that combines Particle Swarm Optimization with Genetic Algorithm to enhance both particle diversity and estimation accuracy. The selection mechanism within the GA framework is implemented through the use of Sorted Weighting Lookahead Particle Filter and Minimalistic Resampling Particle Filter strategies. The resulting hybrid approaches, PSO–GA–SLAPF and PSO–GA–MRPF, improve tracking accuracy and root mean square error (RMSE) within the Particle Filter framework.

The rest of the paper is organized as follows: Section II presents the notations for the particle filter (PF). Section III introduces the particle swarm optimization algorithm. Section IV covers the genetic algorithm. Section V discusses the sorted weighting lookahead particle filter. Section VI explains the minimalistic resampling approach. In Section VII, we propose two hybrid particle filter methods: PSO-GA-SLAPF and PSO-GA-MRPF. Section VIII presents the evaluation results for these two nonlinear methods. Finally, Section IX provides the conclusion.

II. PARTICLE FILTERING

The particle filter is the Bayesian estimator, estimates the hidden dynamic targets in the nonlinear and nonGaussian environment by using set of particles with discrete weights. The particle filter [1] consists of two models, state transition and observation models. These two models are considered as

$$x_t/x_{t\text{-}1} \sim p(x_t/x_{t\text{-}1}, \theta)$$
$$y_t/x_{t\text{-}1} \sim p(y_t/x_t, \theta) \tag{1}$$

Received: November 18, 2025

International Journal of Applied Mathematics,
Volume 38 No. 11s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Where t = 1, · · · , T, $x_t/x_{t\text{-}1}$ is the state transition model with initial distribution $p(x_0)$. $p(x_t/x_{t\text{-}1},\theta)$ is the state transition probability density function(pdf). The $y_t/x_{t\text{-}1}$ is the observation model and $p(y_t/x_t\,,\theta)$ is the observation pdf.

By using the available observations $p(x_1/y_{0:1})$, $p(x_2/y_{0:2})$ and so on, the approximation of target state is done by the PF. The PF approximates this posterior pdf in the Bayesian recursion using a set of weighted particles. The particles $\{x_{t\text{-}1}^i\,,\,w_{t\text{-}1}^i\}_{i=1}^N$ are generated at a time t-1. These particles are propagated through the proposal distribution q(.) and new particles are generated. The particles and their corresponding weights are given by

$$\overline{x}_t^m \sim q\left(x_t/x_{t\text{-}1}^{(i)}\,, y_t\right) \quad (2)$$

$$\overline{w}_t^m = w_{t\text{-}1}^{(i)} \frac{p(y_t/\overline{x}_t^m)p\left(\overline{x}_t^m/x_{t\text{-}1}^{(i)}\right)}{q\left(\overline{x}_t^m/x_{t\text{-}1}^{(i)}, y_t\right)},\ i=1,2,..N \quad (3)$$

These particles represent the posterior at a time t. After few iterations only few particles dominate resulting in increase of disparity among the particles. This problem is called particle degeneracy. To mitigate this problem by resampling process. In this step the particles with high weights are replaces the particles of low weights. Consequently, the posterior distribution in an unbiased PF is represented as a weighted sum of Dirac delta function

$$p(x_t/y_{1:t}) \approx \sum_{i=1}^N \overline{w}_t^i\, \delta\left(x_{t\text{-}1}\text{-}\overline{x}_t^i\right) = \sum_{i=1}^N \frac{n_t^i}{N}\, \delta\left(x_{t\text{-}1}\text{-}\overline{x}_t^i\right) \quad (4)$$

Where n_t^i represents the number of times the replication of ith particle. Resampling step introduces further problems i.e sample impoverishment and reduction of speed

III. PARTICLE SWARM OPTIMIZATION

Particle Swarm Optimization (PSO) is an optimization algorithm. The PSO method adaptively guides particles toward regions with high probability. It maintains particle diversity, moves particles in an intelligently manner rather than randomly, and accelerates convergence to accurately estimate the state. The position and velocity of the particles are updated using the equations

$$v_i = wv_i + c_1\, r_1\left(p_i^{best}\text{-}x_i\right) + c_2\, r_2\left(g^{best}\text{-}x_i\right) \quad (5)$$

$$x_i = x_i + v_i \quad (6)$$

Here c_1 , c_2 the acceleration coefficients and the random numbers r_1, r_2 follow a uniform distribution U(0,1). The g^{best} value initially considered from the measurement value and p_i^{best} is considered initially zero. After sequential iterations the position and velocity of each particle are updated.

Received: November 18, 2025

IV. GENETIC ALGORITHM

The Genetic algorithm incorporates randomized exploration, fitness-based selection, crossover to combine candidates, and mutation to introduce diversity. The genetic algorithm possesses global search capabilities and includes operations like selection, crossover, and mutation. Selection operator selects the new improved set particles. The selection mechanisms commonly employed is Roulette wheel and stochastic uniform sampling methods. However, they have certain drawbacks When particle weights become highly uneven, those with higher weights often dominate the resampling step, leading to particle degeneracy and a reduction in diversity. The crossover operator combines the two parent particles and generates a new child particle. The mutation operator adds a random noise to the child particle to improve the particle diversity.

V. SORTED WEIGHTING LOOKAHEAD PARTICLE FILTER

In the sorted weighting lookahead particle filter particles are sorted, weighting according to their weights. The particles at a time t-1 is considered as $x_{t\text{-}1}^{i}$ and their corresponding weights are $w_{t\text{-}1}^{i}$ for i=1,2 … … . N. These particles are propagated through the nonlinear state mode

$$v_t^i = f\left(x_t^i\right) + e_t, \quad e_t \sim N(0, \tau^2) \tag{7}$$

Where $f(.)$ Is the nonlinear model, and e_t is the state transition noise with mean zero and variance τ^2. The particles are weighted as

$$a_t^i = w_{t\text{-}1}^i p(y_t / x_t^i) \tag{8}$$

The particles at time t are $\left\{v_t^i \,,\, a_t^i\right\}_{i=1}^{N}$ are sorted according to their weights. Now the particles in the set are in the decreasing order. Calculate the sorted weight ratio

$$\eta_t^i = \frac{\overline{w}_t^i}{\overline{w}_t^1} \quad , \quad i=1,2 \ldots \ldots . N \tag{9}$$

Where $\overline{w}_t^i$ is the sorted weights. The sorted weight ratio is compared with some threshold value. The particles which are having $\eta_t^i \geq Thresh$ is considered as high importance region particles and remaining considered as low importance region particles. The set of indices which are having η_t^i greater than threshold are

$$\psi_t = \left\{ j=1 \ldots \ldots . L, \eta_t^i \geq Thresh \right\} \tag{10}$$

Received: November 18, 2025

Therefore, out of N particles L number of particles are having sorted weight ratio is greater than threshold value and remaining N-L particles are in low importance region. These low importance region particles are eliminated and it will be replaced with high importance region particles. The indices of particles are given by

$$\{o(i)_{i=1,\ldots\ldots,N}\} = \left\{ \psi_t \underbrace{, \ldots \ldots, \psi_t}_{\lfloor \frac{N}{L} \rfloor \text{times}}, \psi_t(1, \ldots \ldots . . N\text{-}L\lfloor \tfrac{N}{L} \rfloor) \right\}$$

(11)

L number of indices repeated until to obtain N number. Therefore, the final weighted particles are $x_k^i = v_t^{o(i)}$, $w_k^i = a_t^{o(i)}$.

VI. MINIMALISTIC RESAMPLING PARTICLE FILTER

The particles at a time t-1 is considered as $x_{t\text{-}1}^i$ and their corresponding weights are $w_{t\text{-}1}^i$ for i=1,2 …….N . These particles are propagated through the nonlinear state model

$$v_t^i = f(x_t^i) + e_t, \quad e_t \sim N(0, \tau^2)$$

(12)

Where $f(.)$ Is the nonlinear model, and e_t is the state transition noise with mean zero and variance τ^2. The particles are weighted as

$$a_t^i = w_{t\text{-}1}^i p(y_t / x_t^i)$$

(13)

The particles at time t are $\{v_t^i , a_t^i\}_{i=1}^N$. The weights are normalized, $\overline{a}_t^i$are the normalized weights. The normalized weights are sorted and the sorted order is given by $\overline{a}_t^1 \geq \overline{a}_t^2 \geq \cdots \geq \overline{a}_t^N$. These sorted particle weights are compared with threshold value $\overline{a}_t^i \leq \frac{T_h}{N}$. The set of indices $\left\{i: \overline{a}_t^i \leq \frac{T_h}{N}\right\}$ which are having less value compared to threshold value considered as low weight particles. The high weight particle count is given by

$$L = N\text{-} \left| i: \overline{a}_t^i \leq \frac{T_h}{N} \right|$$

Out N number of particles L number of particles having high weights and L-P particles having low weights. The L-P low weight particles are eliminated. The L number of high weight particles are repeated fixed number of times and the total count should be N. The first M particles are repeated such that

$$k_t^i = \lfloor \frac{N\text{-}L}{L} \rfloor + 2, \; i = 1 \ldots . . M$$

The remaining L-M particles are repeated as

International Journal of Applied Mathematics,
Volume 38 No. 11s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

$$k_t^i=\lfloor\frac{N\text{-}L}{L}\rfloor+1,\ i=M+1\ \ldots..L;$$

The final set of particles and their corresponding indices are

$$\{x_t^i\}_{i=1}^N=\left\{\underbrace{\overline{x}_t^1,\ldots\ldots,\overline{x}_t^1}_{k_t^{i=1}\text{times}},\underbrace{\overline{x}_t^2,\ldots\ldots,\overline{x}_t^2}_{k_t^{i=2}\text{times}},\ldots\ldots.,\underbrace{\overline{x}_t^L,\ldots\ldots,\overline{x}_t^L}_{k_t^{i=L}\text{times}}\right\}$$

$$\{j(i)_{i=1,\ldots\ldots,N}\}=\left\{\underbrace{1,\ldots\ldots,1}_{k_t^{i=1}\text{times}},\underbrace{2,\ldots\ldots,2}_{k_t^{i=2}\text{times}},\ldots\ldots.,\underbrace{L,\ldots\ldots,L}_{k_t^{i=L}\text{times}}\right\}$$

VII. PROPOSED METHOD

PSO-GA-SLAPF

In this section, we present a hybrid algorithm that combines particle swarm optimization (PSO) and genetic algorithm (GA). The PSO algorithm adaptively updates the particles' positions and velocities, guiding them toward regions with higher likelihood. Meanwhile, the genetic algorithm (GA) creates improved versions of particles by using global search operators such as selection, crossover, and mutation. In the PSO-GA-SLAPF algorithm, the selection operator is implemented as a sorted weighting lookahead particle filter.

The steps in the proposed algorithm as follows

- Initialize the particles at time t-1, $\{x_{t\text{-}1}^i, w_{t\text{-}1}^i\}_{i=1}^N$
- Propagate the particle into the system model

$$x_t^i \sim q(x_t/x_{t\text{-}1}^i, y_t)$$

- Update the weights by using the measurement

$$w_t^i=w_{t\text{-}1}^i\frac{p(y_t/x_t^i)P(x_t^i/x_{t\text{-}1}^i)}{q(x_t^i/x_{t\text{-}1}^i, y_t)}$$

- PSO step: Update the position and velocity of particles using the equations

$$v_i=wv_i+c_1\,r_1\left(p_i^{best}\text{-}x_i\right)+c_2\,r_2\left(g^{best}\text{-}x_i\right)$$

$$x_i=x_i+v_i$$

- GA step: It employ selection, crossover and mutation operations
- Selection: The sorted weighting lookahead scheme is employed for selection generates the particle set $\{b_t^i, d_t^i\}_{i=1}^N$

Sort weights: $\{v_t^i, a_t^i\}_{i=1}^N : a_t^{i\text{-}1} \geq a_t^i;$

Ratio function: $\gamma_t^i=\frac{a_t^i}{a_t^1};$

Set of importance indices:

International Journal of Applied Mathematics,
Volume 38 No. 11s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

$$\eta_t=\{j=1,\ldots\ldots M:\gamma_t^i \geq Th\};$$

$$\{b_t^i, d_t^i\}_{i=1}^N \underset{\leftarrow}{Replicate}\{v_t^i, a_t^i\}_{i=1}^N, Replicate\ M \rightarrow N;$$

- Crossover: It generates the parent particles from the set $\{b_t^i, d_t^i\}_{i=1}^N$ and combined to generate a child particle. In this algorithm we employed arithmetic crossover method.

$$O=(1\text{-}\alpha)P^{(1)}+\alpha P^{(2)}, \quad \alpha\sim[0,1]$$

- Mutation: It is employed to preserve diversity and prevent particles from becoming stuck in local minima by introducing random variations.
- The new particle set generated after the mutation is $\{x_t^i, w_t^i\}_{i=1}^N$
- Using the new set of particles approximate the posterior distribution at a time t.

PSO-GA-MRPF

In PSO-GA-MRPF algorithm we employ selection operator as minimalistic resampling particle filter algorithm.

The steps in the proposed algorithm as follows

- Initialize the particles at time t-1, $\{x_{t\text{-}1}^i, w_{t\text{-}1}^i\}_{i=1}^N$
- Propagate the particle into the system model

$$x_t^i \sim q(x_t/x_{t\text{-}1}^i, y_t)$$

- Update the weights by using the measurement

$$w_t^i=w_{t\text{-}1}^i\frac{p(y_t/x_t^i)P(x_t^i/x_{t\text{-}1}^i)}{q(x_t^i/x_{t\text{-}1}^i, y_t)}$$

- PSO step: Update the position and velocity of particles using the equations

$$v_i=wv_i+c_1 r_1(p_i^{best}\text{-}x_i)+c_2 r_2(g^{best}\text{-}x_i)$$
$$x_i=x_i+v_i$$

- GA step: It employ selection, crossover and mutation operations
- Selection: The minimalistic resampling scheme is employed for selection generates the particle set

Sorted normalized weights:

$$\left\{\overline{v}_t^i, \overline{a}_t^i\right\}_{i=1}^N \ :\overline{a}_t^1 \geq \overline{a}_t^2 \geq \cdots \geq \overline{a}_t^N$$

Indices of low Weights $\left\{i: \overline{a}_t^i \leq \frac{T_h}{N}\right\}$, Kill these particles

Large weight particles: L=N- $\left|i: \overline{a}_t^i \leq \frac{T_h}{N}\right|$;

Fixed number of repetitions of first L particles such that count should be N

M particles replicated as :

$$n_t^i = \lfloor \frac{N\text{-}L}{L} \rfloor + 2,\ i = 1 \ldots\ldots M$$

L-M particles repeated as :

$$n_t^i = \lfloor \tfrac{N\text{-}L}{L} \rfloor + 1,\ i = M+1 \ldots\ldots L;$$

Resampling indices : $\{j(i)_{i=1,\ldots\ldots,N}\} = \left\{ \underbrace{1, \ldots\ldots, 1}_{n_t^{i=1}\text{times}}, \underbrace{2, \ldots\ldots, 2}_{n_t^{i=2}\text{times}}, \ldots\ldots, \underbrace{L, \ldots\ldots, L}_{n_t^{i=L}\text{times}} \right\}$;

Resampling particles :

$$\{b_t^i\}_{i=1}^N = \left\{ \underbrace{\overline{v}_t^1, \ldots\ldots, \overline{v}_t^1}_{n_t^{i=1}\text{times}}, \underbrace{\overline{v}_t^2, \ldots\ldots, \overline{v}_t^2}_{n_t^{i=2}\text{times}}, \ldots\ldots, \underbrace{\overline{v}_t^L, \ldots\ldots, \overline{v}_t^L}_{n_t^{i=L}\text{times}} \right\}$$

- Crossover: It generates the parent particles from the set $\{b_t^i, d_t^i\}_{i=1}^N$ and combined to generate a child particle. In this algorithm we employed arithmetic crossover method.

$$O = (1\text{-}\alpha)P^{(1)} + \alpha P^{(2)}, \quad \alpha \sim [0,1]$$

- Mutation: It is employed to preserve diversity and prevent particles from becoming stuck in local minima by introducing random variations.
- The new particle set generated after the mutation is $\{x_t^i, w_t^i\}_{i=1}^N$
- Using the new set of particles approximate the posterior distribution at a time t.

The performance of proposed algorithms are evaluated using two models in the next section.

VIII. EVOLUTION STUDY

To evaluate the performance of the proposed algorithm, we examine two nonlinear models presented as follows.

$$x_t = \frac{x_{t\text{-}1}}{2} + \frac{25x_{t\text{-}1}}{1 + x_{t\text{-}1}^2} + 8cos(1.2t) + a_t$$

(14)

$$y_t = tan^{\text{-}1} x_t + e_t$$

(15)

x_t represents the state transition model, while y_t represents the observation model. $a_t \sim N(0, \tau^2 = 1)$ denotes the state transition noise with zero mean and variance, and $e_t \sim N(0, \sigma^2 = 1)$ denotes the observation noise with zero mean and variance. To evaluate the performance of the model, the following parameters are used: number of particles N = 500, time steps T = 100, threshold value of 0.15. The proposed methods performance is compared with some standard particle filters systematic resampling, residual resampling PF, minimalistic resampling (MRPF) and sorted weighting lookahead PF(SLAPF).

International Journal of Applied Mathematics,
Volume 38 No. 11s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

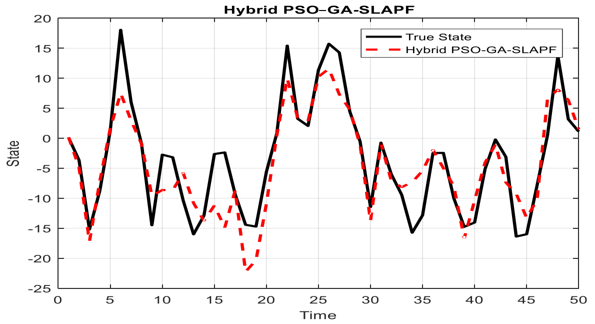

Figure 1: Tracking of performance of PSO-GA-SLAPF method

Figure 1 above illustrates the tracking performance of the proposed method PSO-GA-SLAPF. The black line indicates the true posterior, while the red line shows the posterior from our proposed method. From the figure, it is clear that our method's posterior closely follows the true posterior.

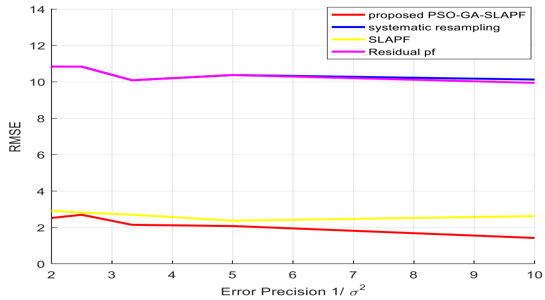

Figure 2: RMSE performance with respect to error precision

Figure 2 illustrates the root mean square error (RMSE) for various error precision values. The performance of the proposed method PSO-GA-SLAPF is compared to that of conventional particle filter (PF) techniques. This figure clearly demonstrates that the proposed method achieves better RMSE results across all examined precision levels.

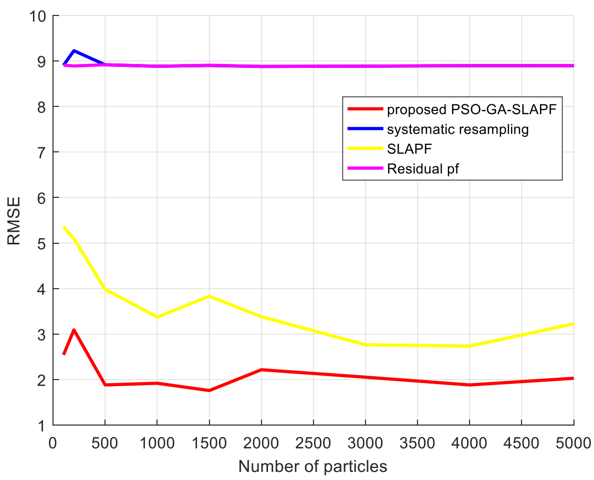

Figure 3: RMSE performance with respect to number of particles.

Received: November 18, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

The above Figure 4 represents the RMSE performance with respect to the number of particles. By increasing the number of particles, the RMSE vale is decreasing. By observing the above figure, the RMSE performance of our proposed method is good as compared to SRF, RPF and SLAPF.

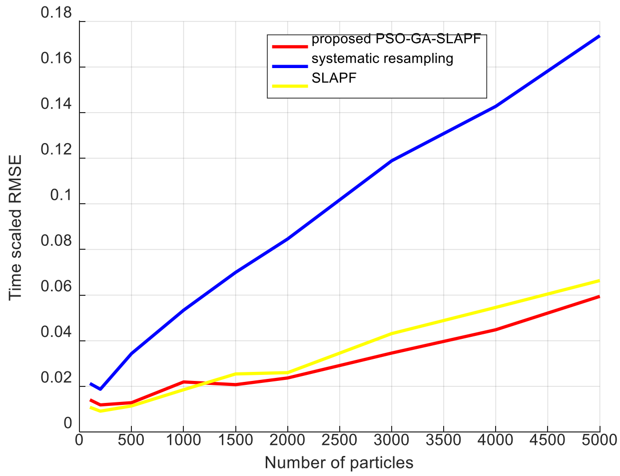

Figure 4: Time scaled RMSE performance with respect to number of particles.

Figure 4 above illustrates the time-scaled RMSE performance in relation to the number of particles. Time-scaled RMSE is calculated by multiplying the computational time by the RMSE. The PSO-GA-SLAPF method demonstrates performance that is nearly comparable to that of SLAPF. SLAPF has a very low computational time, while our method exhibits improved performance.

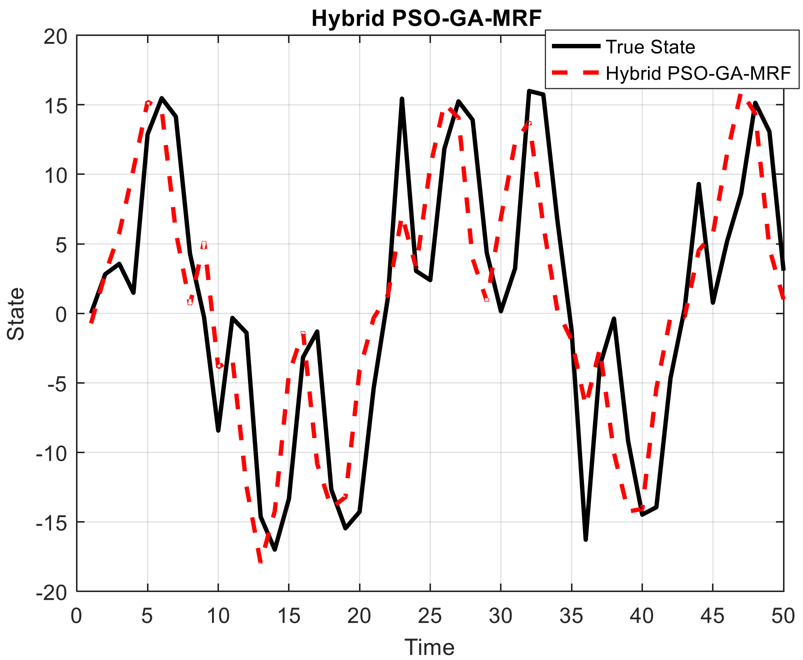

Figure 5: Tracking of performance of PSO-GA-MRPF method

Received: November 18, 2025

Figure 5 above illustrates the tracking performance of the proposed method PSO-GA-MRPF. Figure 6 illustrates the root mean square error (RMSE) for various error precision values. The performance of the proposed method PSO-GA-MRPF is compared to that of conventional particle filter (PF) techniques. This figure clearly demonstrates that the proposed method achieves better RMSE results across all examined precision levels.

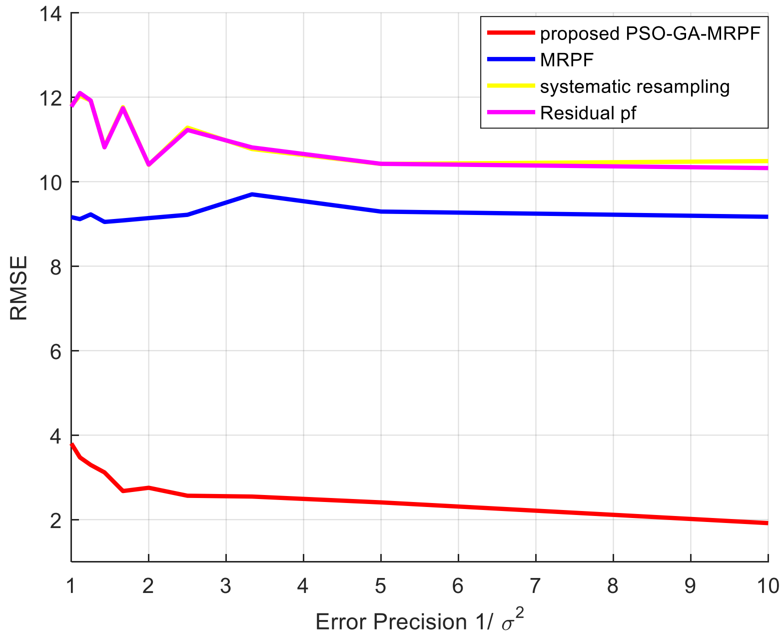

Figure 6: RMSE performance with respect to error precision

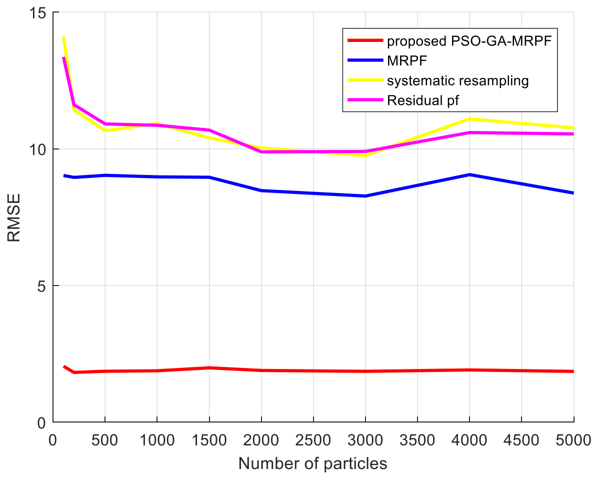

Figure 7: RMSE performance with respect to number of particles.

Figure 7 shows the RMSE performance based on the number of particles, while Figure 8 displays the time-scaled RMSE performance in relation to the number of particles. From these plots, it can be seen that the computational time of PSO-GA-MRPF is somewhat higher compared to the other methods, but its RMSE performance is better. It also demonstrates improved time-scaled RMSE.

Figures 9, 10, and 11 compare the RMSE performance of SLAPF, MRPF, PSO-GA-MRPF, and PSO-GA-SLAPF. From the plots below, it is evident that the proposed methods outperform the existing ones, with PSO-GA-SLAPF demonstrating the best performance among the two proposed approaches.

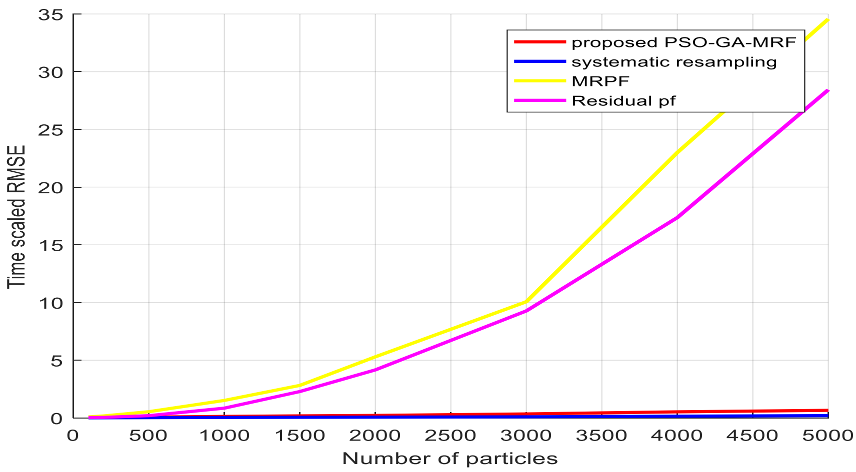

Figure 8: Time scaled RMSE performance with respect to number of particles.

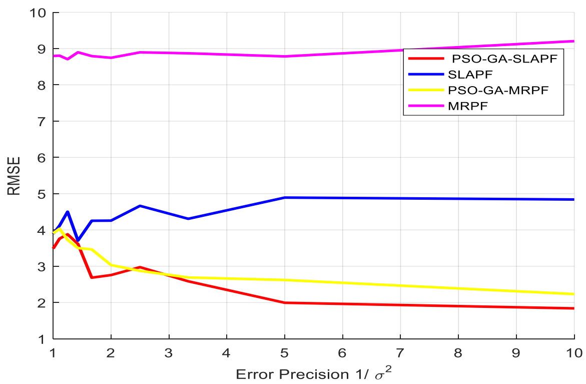

Figure 9: Comparison RMSE performance with respect to error precision

Received: November 18, 2025

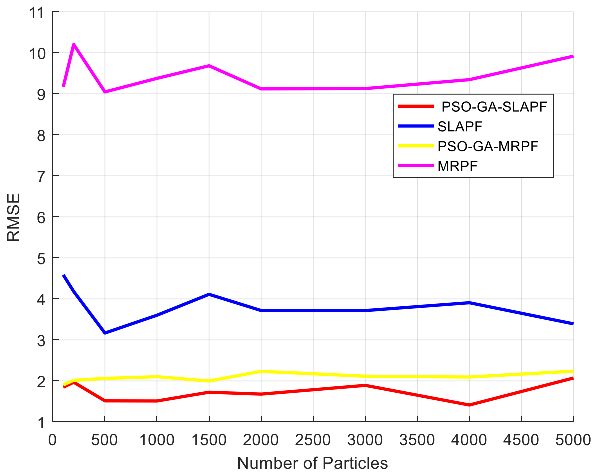

Figure 10: Comparison RMSE performance with respect to number of particles

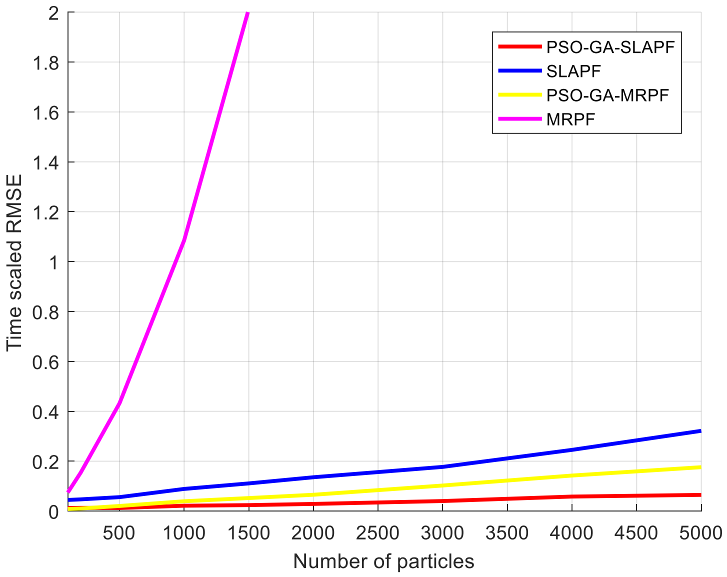

Figure 11: Comparison Time scaled RMSE performance with respect to number of particles.

Track Before Detect Application:

Track before detection will track the target before detecting in the low signal to noise ratio environment. In this section we apply our proposed method to the TBD application and evaluate the performance. The state space model we are considering as follows

$$\begin{bmatrix} x_n \\ y_n \\ \dot{x}_n \\ \dot{y}_n \end{bmatrix} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{n\text{-}1} \\ y_{n\text{-}1} \\ \dot{x}_{n\text{-}1} \\ \dot{y}_{n\text{-}1} \end{bmatrix} + \begin{bmatrix} v_{n,1} \\ v_{n,2} \\ v_{n,3} \\ v_{n,4} \end{bmatrix}$$

(18)

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

In this model the motion of the target is modeled using the two-dimensional cartesian coordinate system. The target state vector is given by X_n=$[x_n \quad y_n \; \dot{x}_n \quad \dot{y}_n]^T$. It is of size 4x1 and it consists of position and velocity components in both x and y directions. The state transition matrix is F_n=$\begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ of size 4x4. Where Δt=1 is considered for the simulation. State noise vector is V_n=$[v_{n,1} \quad v_{n,2} \; v_{n,3} \quad v_{n,3}]^T$ of size 4x1 and the Covariance matrix is Q_n=$diag[1 \;\; 1 \;\; 0.5 \;\;\; 0.5]$.

The measurement model is given by

$$Z= X_n+W$$

Where W is the *measurement noise* (or observation noise), typically assumed to be zero-mean white Gaussian noise with a known covariance matrix is R_n=$\sigma_y^2 I$. In this evaluation the number of time steps considered for simulation are 25, number of particles N=500, Δt=1 and σ_y^2=3.

PSO-GA-SLAPF:

The below Figure 12 shows the Frame 1, the distribution of particles throughout the surface for identifying the target.

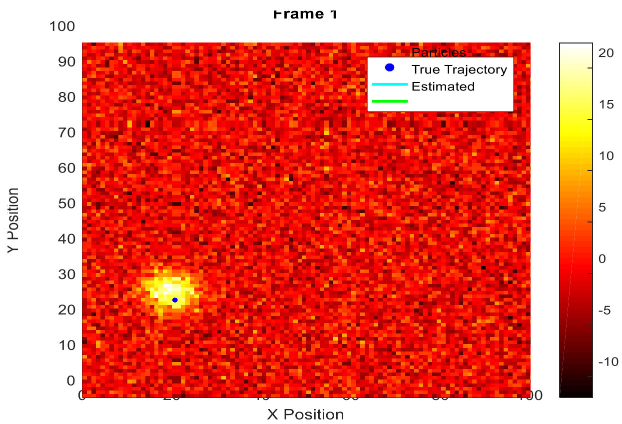

Figure 12: Frame 1 particles are distributed throughout the surface

International Journal of Applied Mathematics,
Volume 38 No. 11s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

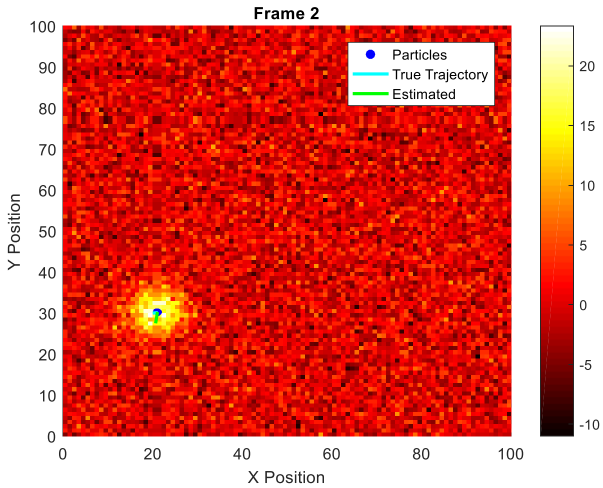

Figure 13: Frame 2 the particles are searching for the target

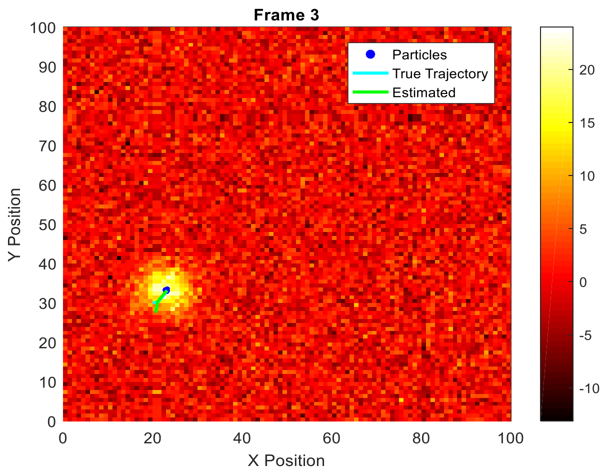

Figure 14: Frame 3 the particles start tracking the target

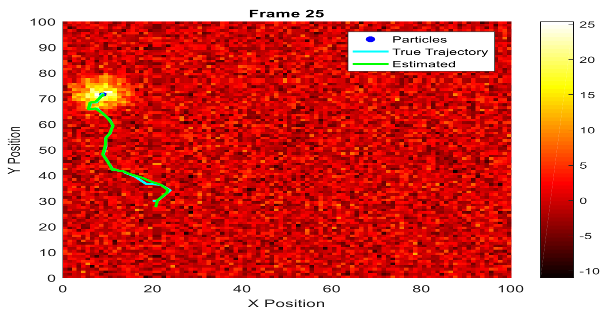

Figure 15: Frame 25 the target tracking of particles

Figure 13 depicts frame 2, showing particles gathered around an area of interest as they actively search for the target. Figure 14 represents frame 3, where the particles successfully locate the target and begin tracking it. Figure 15 displays frame 25, illustrating how the particles maintain tracking of the target over time after its identification.

Received: November 18, 2025

International Journal of Applied Mathematics,
Volume 38 No. 11s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

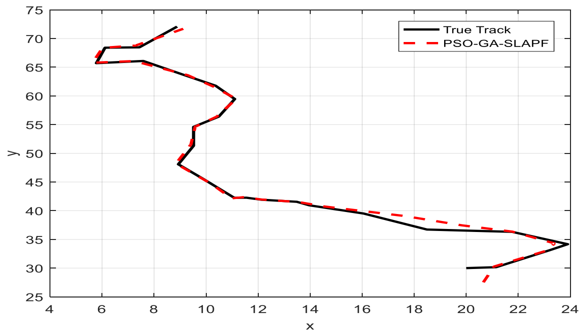

Figure 16: The tracking trajectory along X and Y directions

The Figure 16 illustrates the true and estimated trajectories of the target. The close agreement between the true and estimated trajectories demonstrates that the proposed method effectively estimates and tracks the target. The Figure 17 presents the probability of target presence in the tracking environment. Up to time step 3, the probability of target presence remains below 0.9, indicating uncertainty in target existence. After time step 3, the probability exceeds 0.9, confirming target identification, and the tracking process is subsequently continued.

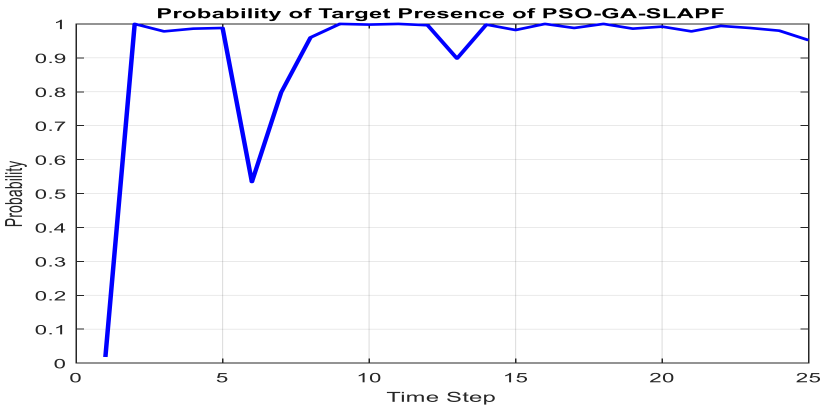

Figure 17: Probability of Target presence.

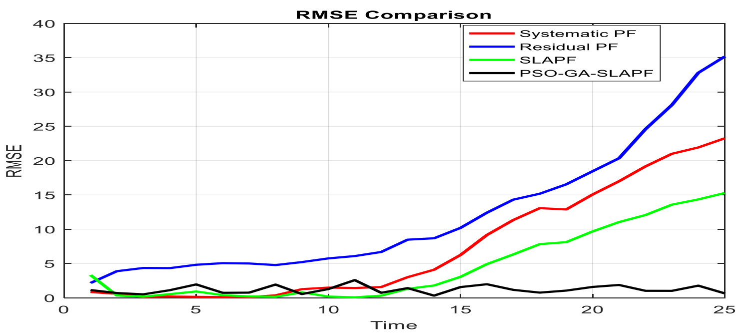

Figure 18: Comparison plot of RMSE performance of Proposed method with standard PFs

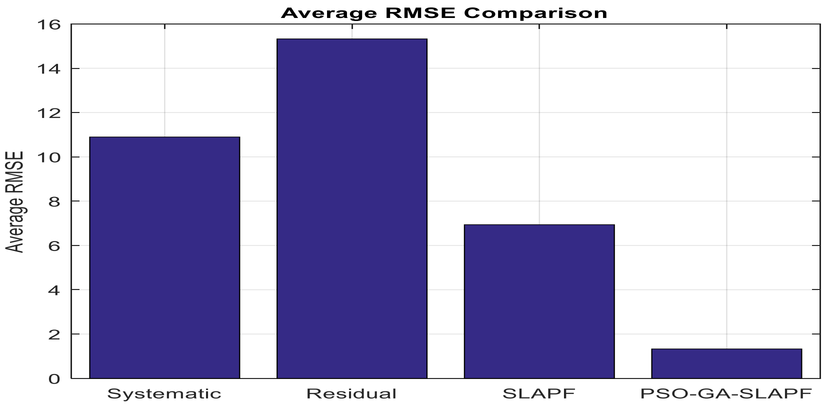

Figure 19: Average RMSE Bar Chart

Figure 18 presents the RMSE performance for the track-before-detect (TBD) application. The proposed methos performance is compared with standard particle filtering approaches, namely SRPF, RPF, and SLAPF. From the figure, it is showing that the proposed method consistently attains lower RMSE values related to the standard PFs.

Figure 19 illustrates the bar chart of the average RMSE values for all the considered methods. The proposed method attains the lowest average RMSE, further confirming its superior estimation accuracy over the conventional particle filtering techniques.

PSO-GA-MRPF

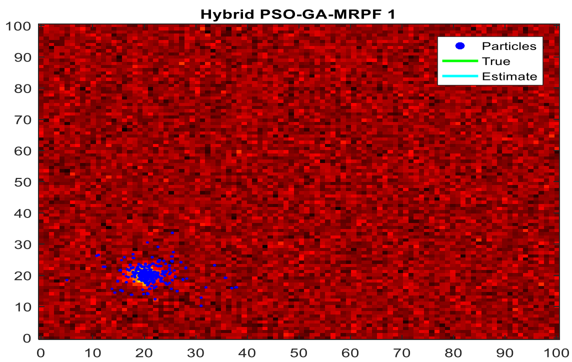

Figure 20: Frame 1 particles are distributed throughout the surface

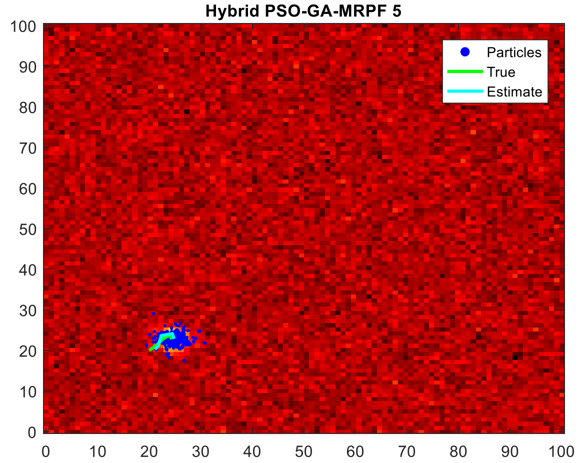

Figure 21: Frame 5 the particles start tracking the target

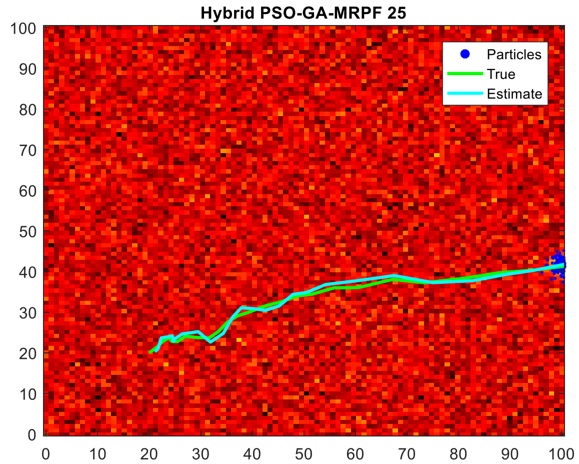

Figure 22: Frame 25 the target tracking of particles

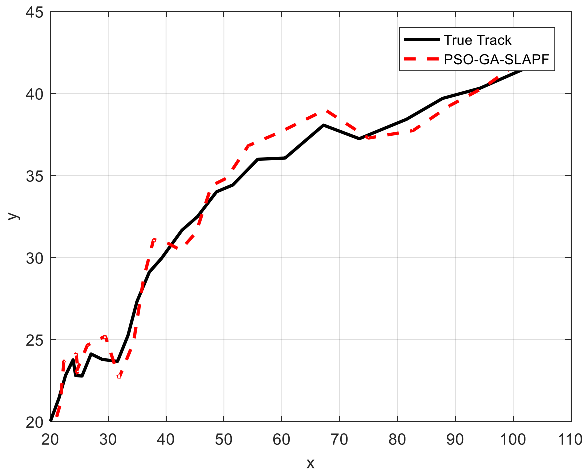

Figure 23: The tracking trajectory along X and Y directions

International Journal of Applied Mathematics,
Volume 38 No. 11s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

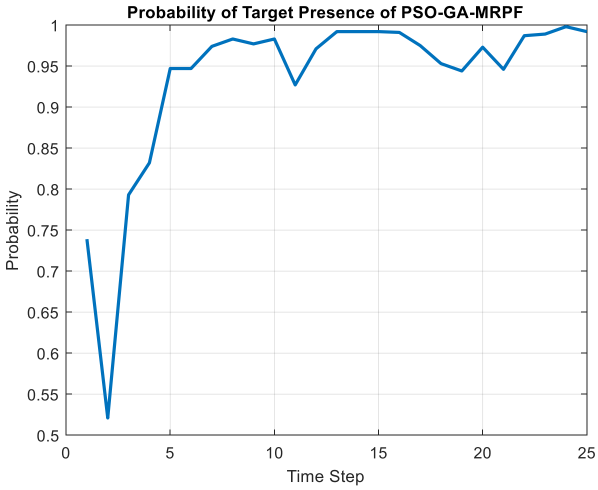

Figure 24: Probability of Target presence.

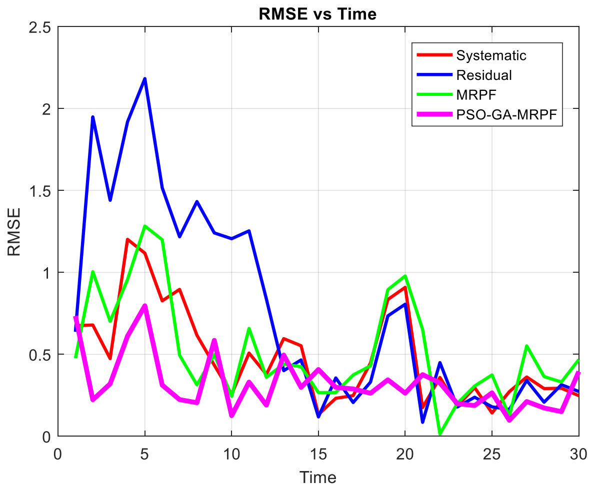

Figure 25: Comparison plot of RMSE performance of Proposed method with standard PFs

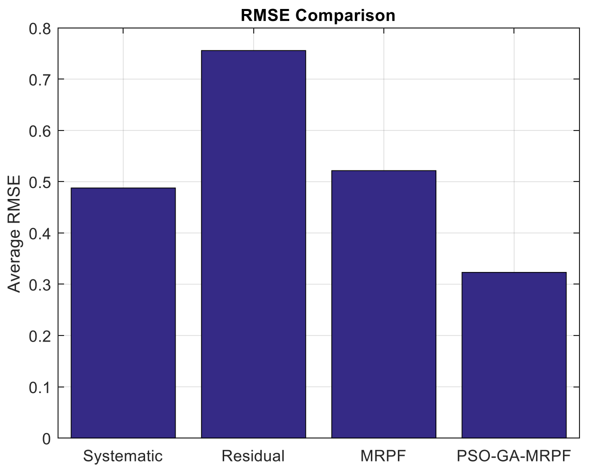

Figure 26: Average RMSE Bar Chart

The above Figure 20 shows the Frame 1, the distribution of particles throughout the surface for identifying the target. Figure 21 represents frame 5, where the particles successfully locate the target and begin tracking it. Figure 22 displays frame 25, illustrating how the particles maintain tracking of the target over time after its identification. The Figure 23 illustrates the true and estimated trajectories of the target. The Figure 24 presents the probability of target presence in the tracking environment. Up to time step 5, the probability of target presence remains below 0.9, indicating uncertainty in target existence. After time step 5, the probability exceeds 0.9, confirming target identification, and the tracking process is subsequently continued.

Figure 25 presents the RMSE performance for the track-before-detect (TBD) application. Figure 26 illustrates the bar chart of the average RMSE values for all the considered methods.

Overall, the simulation results demonstrate that the proposed methods provide improved RMSE and Time scaled RMSE for both the models when compared with standard particle filter–based approaches.

IX. CONCLUSION

In this paper, we introduced two hybrid particle filter methods: PSO-GA-SLAPF and PSO-GA-MRPF. The performance of these methods was assessed using a nonlinear 1D model and a track-before-detect application. The simulation results for the nonlinear 1D model demonstrate that both proposed approaches achieve better RMSE and time-scaled RMSE compared to standard particle filters. Among the two, PSO-GA-SLAPF exhibits superior performance. In the track-before-detect application, PSO-GA-SLAPF detects the target at frame 3, while PSO-GA-MRPF identifies the target at frame 5.

Furthermore, within the track-before-detect (TBD) framework, the proposed approach enables early and reliable target identification, allowing accurate target tracking to be initiated at an earlier time instant. Consequently, the proposed method achieves superior RMSE performance and exhibits enhanced robustness compared to conventional particle filter–based techniques.

REFERENCES

[1] Miodrug BoliC, Petar M. DjuriC, and Sangiin Hong, “New Resampling Algorithms for Particle Filters”. IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003.

[2] Ramakrishna G, P. B. Choppala, James Stephen M and Paul D. Teal, “*A sorted weighting lookahead sampling scheme for accurate and fast particle filtering,*” second international conference on Range Technology (ICORT), IEEE, August 2021.

[3] Ramakrishna G, P. B. Choppala, James Stephen M and Paul D. Teal, “*A Fast and Unbiased Minimalistic Resampling Approach for the Particle Filter,*” IEEE International conference on signal and image processing applications (ICSIPA), October 2021.

[4] Ning Zhou, Lawrence Lau, Ruibin Bai, Terry Moore, “A Genetic Optimization Resampling Based Particle Filtering Algorithm for Indoor Target Tracking”, remote sensing journal, January 2021.

[5] Gongyuan Zhang, Yongmei Cheng, Feng Yang, Quan Pan, “Particle Filter Based on PSO”, International Conference on Intelligent Computation Technology and Automation, August 2008.

[6] Tiancheng Li, Miodrag Bolic´, and Petar M. Djuric´, “Resampling Methods for Particle Filtering (Classification, implementation, and strategies)” Signal Processing Magazine IEEE, May 2015.

[7] J. D. Hol, T. B. Schon, and F. Gustafsson, “On resampling algorithms for particle filters,” Proc. Nonlinear Statistical Signal Processing Workshop IEEE, pp. 79–82, 2006

[8] Kitagawa G, “Monte Carlo filter and smoother for non-Gaussian non-linear state space models,” Journal of Computational and Graphical Statistics, pp. 1–25, 1996.

[9] J. Carpenter, P. Clifford, and P. Fearnhead, “An improved particle filter for nonlinear problems,” IEEE Proc. Radar, Sonar and Navigation, vol.146, pp. 2–7, July 1999.

[10] M. Bolic, Petar M. Djuric, and Sangjin Hong, “Resampling algorithms for particle filters: A computational complexity perspective,” Proc. EURASIP Journal on Advances in Signal Processing, pp. 2267–2277, 2004.

[11] Kennedy J, Eberhart R C. (1995) “ Particle swarm optimization”, Proceedings of the 1995 IEEE International Conference on Neural Networks，4: 1942－1948.

[12] Xuerong C, Jin Y, Juan Li, Chunlei W, “Improved Genetic Algorithm to Optimize the Wi-Fi Indoor Positioning Based on Artificial Neural Network”, IEEE Access. January 2020.

[13] Yonglin L, Chunming Y, Changping L, “Roulette selection self-adaptive harmony search algorithm”, Application Research of Computers, 31 (6): 1665-1668, August 2014.