

Article

Tracking by Risky Particle Filtering over Sensor Networks

Jaechan Lim ^{1,2}  and Hyung-Min Park ^{1,*} ¹ Department of Electronic Engineering, Sogang University, Seoul 04107, Korea; jaechan@gmail.com² Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109, USA

* Correspondence: hpark@sogang.ac.kr

Received: 29 April 2020; Accepted: 29 May 2020; Published: 31 May 2020



Abstract: The system of wireless sensor networks is high of interest due to a large number of demanded applications, such as the Internet of Things (IoT). The positioning of targets is one of crucial problems in wireless sensor networks. Particularly, in this paper, we propose minimax particle filtering (PF) for tracking a target in wireless sensor networks where multiple-RSS-measurements of received signal strength (RSS) are available at networked-sensors. The minimax PF adopts the maximum risk when computing the weights of particles, which results in the decreased variance of the weights and the immunity against the degeneracy problem of generic PF. Via the proposed approach, we can obtain improved tracking performance beyond the asymptotic-optimal performance of PF from a probabilistic perspective. We show the validity of the employed strategy in the applications of various PF variants, such as auxiliary-PF (APF), regularized-PF (RPF), Kullback–Leibler divergence-PF (KLDPF), and Gaussian-PF (GPF), besides the standard PF (SPF) in the problem of tracking a target in wireless sensor networks.

Keywords: degeneracy; minimax; particle filtering; risk; received signal strength; target tracking; wireless sensor networks

1. Introduction

Recently, research related to Internet of Things (IoT) is a hot area where many people from diverse fields are working due to the demand of smart systems that enable the ability to transfer data over a network without requiring human interactions by using smart devices, such as smartphones [1–4]. In this smart system, a sensor network is formed to interconnect densely populated sensors for data communications and processing among the sensors. A traditional field of wireless sensor networks contributes to enabling the IoT [5,6]. There are many applications that can be enabled by wireless sensor network-based IoT, such as health care systems, transportation, defense applications, and smart-home systems.

Depending on applications, a sensor network can comprise different types of sensors. Tracking a moving target is an important problem in wireless sensor networks where the position and/or velocity of a moving object within the region of interest is estimated. The problem can be classified by three main methods that can be employed for the tracking application according to the type of physical measurement read at sensors. The main types of measurements are the time delay of arrival (TDOA) [7,8], the direction of arrival (DOA) [9,10], and the received signal strength (RSS) [11–19] of the signal generated from the tracked target, respectively. RSS measurement of the sensors represents received energy or power emitted by the target. RSS-based sensors have advantages of simpler architecture and less expensive cost than the cases of TDOA and DOA. In this paper, we propose a

tracking approach that uses multi-RSS measurements. In particular, we propose the minimax particle filtering (PF) approach for a target tracking, which estimates the trajectory and velocity.

Within the past few decades, since PF was proposed [20], PF has shown asymptotically optimal performance in various time-varying state estimation problems, particularly for nonlinear problems. Further, to advance the performance of the standard PF, various versions of PF have been proposed since its initial proposal [21–30]. Based on the theoretical background, PF can perform optimally by using an infinite number of particles in nonlinear problems. Nonetheless, from time to time, we cannot obtain ideal results via PF due to various unexpected reasons. One main reason is the degeneracy phenomenon, i.e., after a few time steps since the initialization of the algorithm, there is only one particle that has significant weight, whereas all the other particles have nearly zero weights. Consequently, the variance of the weights of particles only increases over time, which results in insufficient performance due to the loss of particle diversity [31]. To overcome the degeneracy issue, the resampling [32,33] can be applied that regenerates high-quality particles duplicated. On the other hand, as a by-effect of the resampling, PF undergoes particle impoverishment that we have only identical particles over a short time, especially when the state noise is little. Therefore, maintaining a constantly small variance of the weights of particles is desired to avoid the unsatisfactory performance of PF. For this goal, we adopt the strategy of the minimax approach that incurs a significantly mitigated variance of the weights of particles and results in improved performance of PF. In this approach, we minimize the maximum risk function. That is, we adopt the maximum risk when computing the weights of particles; consequently, the proposed approach minimizes the highly increased risk function compared to the cases of regular PFs. In other words, we devalue the quality of a particle as low as possible.

In this paper, we propose minimax-PF (MPF) for the target tracking based on multi-RSS-measurements at up to ten sensors. The minimax PF approach was initially introduced in Reference [34] recently, where a maneuvering target is tracked based on the range and bearing, which are only two measurements; therefore, the methodology was not fully investigated in terms of the number of measurements, clarity of the derivation of the approach, analysis of risk and weight concerning to all compared variants of PFs. The investigation of the method in the application of networked sensor measurements where a large number of sensors are available is more appropriate for the verification of the approach. In Reference [35], the minimax strategy was adopted in an interactive-multiple-model algorithm implemented by PF with a couple of range and bearing measurements only. In this paper, we verify with up to ten RSS measurements in wireless sensor networks to generalize the validity of the proposed approach with extensive experimental results and analysis. The number of employed measurements is a key factor to assess and generalize the validity of the proposed minimax PF approach. We show the outperforming results of the proposed MPFs compared to the various regular PFs, i.e., standard PF, APF, GPF, RPF, and KLDPF.

For notations, matrices, vectors, and scalars are denoted by bold uppercase letters, bold lowercase letters, and lowercase letters, respectively, and \top denotes the transpose of a matrix.

2. Problem Formulation

We track a target based on a number of RSS measurements in a two-dimensional space. The scenario also can be a localization problem, but tracking is necessary when we have limited time or computational complexity in positioning the target.

2.1. Dynamic State Model

The movement of the target is mainly driven by the state noise that is also the random acceleration of the target. The state and measurement variables are denoted by $\boldsymbol{\psi}$ and \boldsymbol{m} , respectively, and the dynamic state function is described referring to Reference [36–38] as follows:

$$\boldsymbol{\psi}_t = \boldsymbol{D}\boldsymbol{\psi}_{t-1} + \boldsymbol{E}\boldsymbol{\theta}_t, \quad (1)$$

where

$$\boldsymbol{\psi}_t = [\psi_{1,t}, \psi_{2,t}, \dots, \psi_{1,L}]^\top \quad (2)$$

$$= [r_{x,t}, r_{y,t}, v_{x,t}, v_{y,t}]^\top, \quad (3)$$

$$\boldsymbol{\psi}_{t-1} = [r_{x,t-1}, r_{y,t-1}, v_{x,t-1}, v_{y,t-1}]^\top, \quad (4)$$

$$\boldsymbol{\theta}_t = [\theta_{x,t}, \theta_{y,t}]^\top \quad (5)$$

$$\mathbf{D} = \begin{bmatrix} 1 & 0 & T_s & 0 \\ 0 & 1 & 0 & T_s \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{E} = \begin{bmatrix} \frac{T_s^2}{2} & 0 \\ 0 & \frac{T_s^2}{2} \\ T_s & 0 \\ 0 & T_s \end{bmatrix}, \quad (6)$$

where r , v , θ , and (x, y) denote the position, speed, acceleration, and 2-D coordinates, respectively. L is the number of state elements which is composed of 2-D location and 2-D velocity, T_s is the sampling time period, and t is the discrete-time step. The state is driven by the random process of $\boldsymbol{\theta}_t$.

2.2. Measurement Model

The vector representation of the noisy measurements of the received signal strength (RSS) at M multi-sensors can be described as

$$\mathbf{m}_t = \mathbf{g}(\boldsymbol{\psi}_t) + \boldsymbol{\eta}_t, \quad (7)$$

where $\mathbf{g}(\cdot)$ and $\boldsymbol{\eta}_t$ are the observation function with respect to the state and additive measurement noise, respectively, and can be described in detail as follows.

$$\mathbf{m}_t = [m_{1,t}, m_{2,t} \dots m_{M,t}]^\top, \quad (8)$$

$$\mathbf{g}(\boldsymbol{\psi}_t) = [g_1(\boldsymbol{\psi}_t), g_2(\boldsymbol{\psi}_t) \dots g_M(\boldsymbol{\psi}_t)]^\top, \quad (9)$$

$$\boldsymbol{\eta}_t = [\eta_{1,t}, \eta_{2,t} \dots \eta_{M,t}]^\top. \quad (10)$$

The RSS model is discussed in Reference [39], where the received power is expressed in dBm. Thus, the received signal power at the sensor i at the time step t is expressed as follows:

$$m_{i,t} = g_i(\boldsymbol{\psi}_t) + \eta_{i,t} = P_0 - 10\alpha \log_{10} \left(\frac{|\mathbf{l}_i - \mathbf{r}_t|}{d_0} \right) + \eta_{i,t}, \quad \text{for } i = 1, \dots, M, \quad (11)$$

where P_0 is the received power at the sensor from a reference distance d_0 , \mathbf{l}_i is the position of the i -th sensor, $\mathbf{r}_t = [r_{x,t}, r_{y,t}]^\top$ is the location of the target at the time step t , α is an attenuation parameter that depends on the transmission medium, $\eta_{i,t} \sim \mathcal{N}(0, \sigma_{\eta_{i,t}}^2)$ is a Gaussian noise process with the variance $\sigma_{\eta_{i,t}}^2$, and M is the number of employed sensors in the networks. In this case, the likelihood function with respect to only $m_{i,t}$ becomes $\mathcal{N}(m_{i,t}; g_i(\boldsymbol{\psi}_t), \sigma_{\eta_{i,t}}^2)$, which denotes a Gaussian distribution function at $m_{i,t}$ with the mean $g_i(\boldsymbol{\psi}_t)$ and the variance $\sigma_{\eta_{i,t}}^2$, respectively.

3. Proposed Approach

In this section, we derive the minimax PF based on a fused manner of classical and Bayesian methods. This approach is derived by adopting the minimax strategy to standard PF (SPF). In the SPF, the risk function defined as the expected squared error is minimized, while the maximum risk function is minimized when the minimax strategy is adopted. The minimax strategy is reflected in the process of computing the weights of particles in the algorithm. Before we describe the minimax approach, we

recall minimum mean square error estimator and PF first in order to derive the proposed approach smoothly. A Bayesian estimator that minimizes a predefined risk function can be described as follows.

$$\min_{\hat{\psi}_t} \mathcal{R}_t(\psi_t, \hat{\psi}_t), \quad (12)$$

where, $\psi_t \in \Psi_t$, and the estimator $\hat{\psi}_t$ minimizes a risk function with respect to the random variable ψ_t . In the case of minimum mean square error (MMSE) estimator, the risk function can be described by

$$\mathcal{R}_t(\psi_t, \hat{\psi}_t) = \mathbb{E}(C_t), \quad (13)$$

where $\mathbb{E}(\cdot)$ denotes the expected value, and the cost function $C_t(\cdot)$ is defined by the squared error if we let ϵ_t denotes the error of the estimator as follows.

$$C_t = \epsilon_t^2, \quad \epsilon_t = \psi_t - \hat{\psi}_t. \quad (14)$$

The MMSE estimator that minimizes the mean square error defined as the risk function is described as follows:

$$\hat{\psi}_t^{\text{MMSE}} = \min_{\hat{\psi}_t} \mathcal{R}_t^{\text{MMSE}} = \min_{\hat{\psi}_t} [\mathbb{E}(C_t)] \quad (15)$$

$$= \int \psi_t p(\psi_{0:t} | m_{1:t}) d\psi_t, \quad (16)$$

where $p(\psi_{0:t} | m_{1:t})$ is the posterior density.

In PF, the problem is formulated as follows.

$$\hat{\psi}_t^{\text{PF}} = \min_{\hat{\psi}_t} \mathcal{R}_t^{\text{PF}} = \min_{\hat{\psi}_t} \left[\sum_{\kappa=1}^N (w_t^\kappa C_t^\kappa) \right], \quad (17)$$

where $C_t^\kappa = (\psi_t^\kappa - \hat{\psi}_t)^2$, $\epsilon_t^\kappa = \psi_t^\kappa - \hat{\psi}_t$, N is the number of employed particles, κ is the particle index, w_t^κ is the weight of the particle κ at the time step t , and ψ_t^κ is the particle with the index κ . Then, the MMSE estimate of PF can be derived as follows.

$$\hat{\psi}_t^{\text{PF}} = \sum_{\kappa=1}^N w_t^\kappa \psi_t^\kappa. \quad (18)$$

In the problem of estimating a deterministic variable $\psi_t \in \Psi_t$ based on observation, an estimator $\hat{\psi}_t^{\mathcal{M}}$ is the minimax estimator if its maximum risk is the minimum value by all possible estimators, which can be described as follows.

$$\max_{\psi_t \in \Psi_t} \mathcal{R}_t(\psi_t, \hat{\psi}_t^{\mathcal{M}}) = \min_{\hat{\psi}_t} \max_{\psi_t \in \Psi_t} \mathcal{R}_t(\psi_t, \hat{\psi}_t), \quad (19)$$

Therefore, the criterion of this classical estimator is avoiding the maximum risk by minimizing the maximum risk. To formulate the minimax strategy in PF framework, from Equations (13) and (14), we define a new minimax risk function as follows:

$$\mathcal{R}_t^{\mathcal{M}} = \max \mathcal{R}_t^{\text{PF}} \quad (20)$$

$$= \max \sum_{\kappa=1}^N (w_t^\kappa C_t^\kappa) \quad (21)$$

$$= \sum_{\kappa=1}^N (w_t^{\mathcal{M},\kappa} C_t^\kappa), \quad (22)$$

where $w_t^{M,\kappa}$ denotes the modified weight associated with the particle index κ in minimax PF (MPF) because the minimax strategy is reflected when computing the weights of particles. Specifically, we select only one RSS-sensor-measurement among M that may incur the highest risk. In other words, concerning M RSS-sensor-measurements, we use only one measurement that provides not the maximum weight but the minimum weight, which is highly counter-intuitive approach in terms of probabilistic point of view. In Equation (22), i.e., the risk function of MPF, only $w_t^{M,\kappa}$ is the factor of the magnitude of the risk because the particles are given, and we cannot modify them. When we associate the weight with only a single sensor-measurement, the weight is described as follows.

$$w_t^{M,\kappa} \in \{w_{i,t}^{\kappa}\}_{i=1}^M, \quad (23)$$

where the weight of the particle ψ_t^{κ} can be computed in M ways, and $w_{i,t}^{\kappa}$ is the weight only based on the measurement $m_{i,t}$. Each particle is associated with the argument i , individually. Therefore, by associating the minimum weight among M weights for individual particle, we can maximize the risk function Equations (20)–(22), in minimax PF. If we consider the deterministic variable of classical minimax estimator as a random variable, we can modify the minimax Equation (19) with respect to MPF as follows.

$$\hat{\psi}_t^M = \min_{\hat{\psi}_t} \mathcal{R}_t^M \quad (24)$$

$$= \min_{\hat{\psi}_t} \max \mathcal{R}_t^{\text{PF}} \quad (25)$$

$$= \min_{\hat{\psi}_t} \max_{\{w_{i,t}^{\kappa}\}_{\kappa=1}^N} \mathcal{R}_t^{\text{PF}} \left(\{w_{i,t}^{\kappa}\}_{\kappa=1}^N, \hat{\psi}_t \right) \quad (26)$$

$$= \min_{\hat{\psi}_t} \max_{\{w_{i,t}^{\kappa}\}_{\kappa=1}^N} \sum_{\kappa=1}^N (w_{i,t}^{\kappa} C_t^{\kappa}) \quad (27)$$

$$= \min_{\hat{\psi}_t} \sum_{\kappa=1}^N \left(w_t^{M,\kappa} C_t^{\kappa} \right), \quad (28)$$

where $\{w_t^{M,\kappa}\}_{\kappa=1}^N$ are defined as the weights in minimax PF that maximize the minimax risk function. If we let $w_t^{M,\kappa} \triangleq w_{\tau,t}^{\kappa}$, where

$$\tau = \underset{i \in \{1, \dots, M\}}{\operatorname{argmin}} w_{i,t}^{\kappa}, \quad (29)$$

the estimate of MPF at time step t is obtained from Equation (18) as follows.

$$\hat{\psi}_t^M = \sum_{\kappa=1}^N \omega_t^{M,\kappa} \psi_t^{\kappa}, \quad (30)$$

$$= \sum_{\kappa=1}^N \omega_{\tau,t}^{\kappa} \psi_t^{\kappa}. \quad (31)$$

In this approach, we may select a different RSS-sensor-measurement that devalues the quality of the particle as low as possible for the computation of the weight of each particle. The minimax PF algorithm for the standard PF in wireless sensor networks is described in Algorithm 1 as follows:

Algorithm 1: The minimax PF algorithm for the standard PF in wireless sensor networks

□ Initialization

for $\kappa = 1, \dots, N$, where N is the number of particles.

1. Random generation of initial particles:

$$\boldsymbol{\psi}_0^\kappa \sim p(\boldsymbol{\psi}_0), \text{ and assign initial weights: } \omega_0^\kappa = \frac{1}{N}.$$

end

□ Sequential update

for $t = 1, \dots, T$, where T is the total time steps.for $\kappa = 1, \dots, N$,

2. Propagation of particles via a proposal density:

$$\boldsymbol{\psi}_t^\kappa \sim q(\boldsymbol{\psi}_t | \boldsymbol{\psi}_{t-1}^\kappa, \mathbf{m}_t). \quad (32)$$

end

for $\kappa = 1, \dots, N$,

3. Computing the weights of particles:

$$\omega_{i,t}^\kappa = p(m_{i,t} | \boldsymbol{\psi}_t^\kappa) \quad \text{for } i = 1, \dots, M \quad (33)$$

assuming the proposal density,

$$q(\boldsymbol{\psi}_t | \boldsymbol{\psi}_{t-1}^\kappa, \mathbf{m}_t) = p(\boldsymbol{\psi}_t | \boldsymbol{\psi}_{t-1}^\kappa).$$

end

for $\kappa = 1, \dots, N$,

4. Normalization.

$$\bar{\omega}_{i,t}^\kappa = \frac{\omega_{i,t}^\kappa}{\sum_{\kappa=1}^N \omega_{i,t}^\kappa} \quad \text{for } i = 1, \dots, M. \quad (34)$$

5. Selecting the minimum weight among M weights.

$$\omega_t^{\mathcal{M},\kappa} = \bar{\omega}_{\tau,t}^\kappa, \quad \text{where } \tau = \underset{i \in \{1, \dots, M\}}{\operatorname{argmin}} \bar{\omega}_{i,t}^\kappa.$$

end

6. Normalization of the weights:

$$\bar{\omega}_t^{\mathcal{M},\kappa} = \frac{\omega_t^{\mathcal{M},\kappa}}{\sum_{\kappa=1}^N \omega_t^{\mathcal{M},\kappa}} \quad \text{for } \kappa = 1, \dots, N. \quad (35)$$

7. Computing the estimate at the time step t :

$$\hat{\boldsymbol{\psi}}_t^{\mathcal{M}} = \sum_{\kappa=1}^N \bar{\omega}_t^{\mathcal{M},\kappa} \boldsymbol{\psi}_t^\kappa. \quad (36)$$

8. Resampling N particles.

end

Therefore, as long as the computation of the weights of particles is performed with multiple measurements, this approach can be applied to any variants of PF. We only need to modify the step of computing the weights of particles in their algorithms to significantly mitigate the degeneracy of particles.

4. Performance Assessment

We evaluate the performance of the proposed MPFs, and compare with those of original PF variants, i.e., sequential-importance-resampling PF (SPF), regularized PF (RPF), auxiliary PF (APF), Kullback–Leibler-divergence PF (KLDPF), and Gaussian PF (GPF) for tracking a target in wireless sensor networks. We employ multi-sensors from two up to ten sensors in the experiments, and obtain various results depending on the number of employed sensors. We also compare the mean-risk based on Equation (20) and the variance of the weights of particles for both MPFs and original PF variants. The performance is assessed based on 300 runs of computer simulations, and 1000 particles (we do not improve the performance anymore beyond 1000) are used for all approaches. We consider two scenarios in terms of the target-motion-pattern: relatively a large or a small variance is considered for the state process noise based on Equation (1), where the larger the noise variance is, the larger the acceleration of the motion of the target becomes. The scenarios of large and small noises are denoted by S_L and S_S , respectively. Therefore, the motion of the target is much greater under S_L compared to the case under S_S in a single time step, and particles undergo severe degeneracy under S_L . We denote the noise variances of the state and the sensor-measurement, by $\sigma_\theta^2, \sigma_\eta^2$, and $\Sigma^2 = (\sigma_\theta^2, \sigma_\eta^2)$, respectively. The noise variance for the state is assumed that $\sigma_\theta^2 = \sigma_{\theta_x}^2 = \sigma_{\theta_y}^2$. We used $(10^{-3}, 10^{-5})$ and $(10^{-1}, 10^{-5})$ for Σ^2 , in S_S and S_L , respectively. In the experiments of SPF, we used $(1, 10^{-5})$ for Σ^2 , in S_S . In the experiments of APF, we used $(10^{-3}, 10^{-3})$ and $(10^{-1}, 10^{-3})$ for Σ^2 , in S_S and S_L , respectively. The true initial state was generated with the expected values of $[1, 1, 0.1, 0.1]^T$ within a uniform distribution, and the support-interval-length of $[0.1, 1]$ were used for $[S_S, S_L]$, respectively. The initial particles were generated in the same way as the true state. The number of total time steps $T = 100$. We performed 300 runs to obtain mean square error (MSE) of distance (MSED) and mean distance error (MDE) for each location over the T time steps. Cramér-Rao lower bound (CRLB) for the problem is provided in the Appendix A. In the simulations, we generated true trajectories based on the state model first and then deploy sensors uniformly in the area based on the starting and ending points of the target.

4.1. Standard Particle Filter

We begin with the sequential importance resampling particle filter as the standard particle filter (SPF) to compare the performance of the original and minimax versions. please add a and b for the subfigures and provide explanation in the caption. Mean distance error (MDE) of RPF and MRPF is compared in Figure 1. The left of Figure 1 shows the results under the scenario of S_S with the various number of employed sensors M (i.e., 2, 5, and 10 sensors). When we use 2 sensors, both PF and MPF show similar MDE performance, while MPF outperforms PF when 5 and 10 sensors are employed. As M increases, the overall performances of both methods are improved. The right of Figure 1 shows the results under the scenario of S_L with the various number of employed sensors M . The overall results under S_L of both methods are worse than those under S_S due to larger accelerations of the target. The superiority of MPF over PF is clear when $M = 5$, while the superiority is not increasing significantly when 10 sensors are employed. When $M = 2$, the performance difference between PF and MPF is not clear between the two methods, as in the case of S_S . Overall, MPF outperforms PF in terms of MDE as M increases, while the performance gain is not significantly increased beyond five sensors. The result shows that the minimax strategy is more effective under the S_L scenario.

Based on Equation (20), we computed the risk function for both PF and MPF associated with the MDE-result of Figure 1, and the mean values over 300 runs are shown in Figure 2. The left and right of Figure 2 show the results under the scenarios of S_S and S_L with various M , respectively. The overall risk decreases as M increases regardless of the scenario. The gap of mean-risks between two methods is severer as M increases. The results show larger risks under S_L compared to the case of S_S . The result of Figure 1 was obtained by minimizing the risk shown in Figure 2.

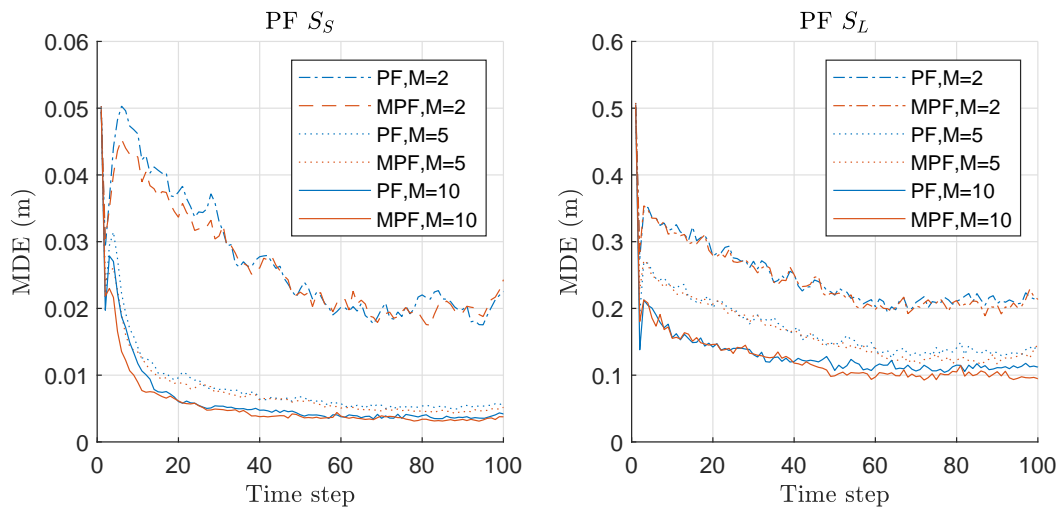


Figure 1. Mean distance error (MDE) performance comparison between particle filtering (PF) and minimax PF (MPF). Three hundred runs were performed with 1000 particles, where M , S_S , and S_L denote the number of sensors, the scenario of the small state noise variance, and the scenario of the large state noise variance, respectively.

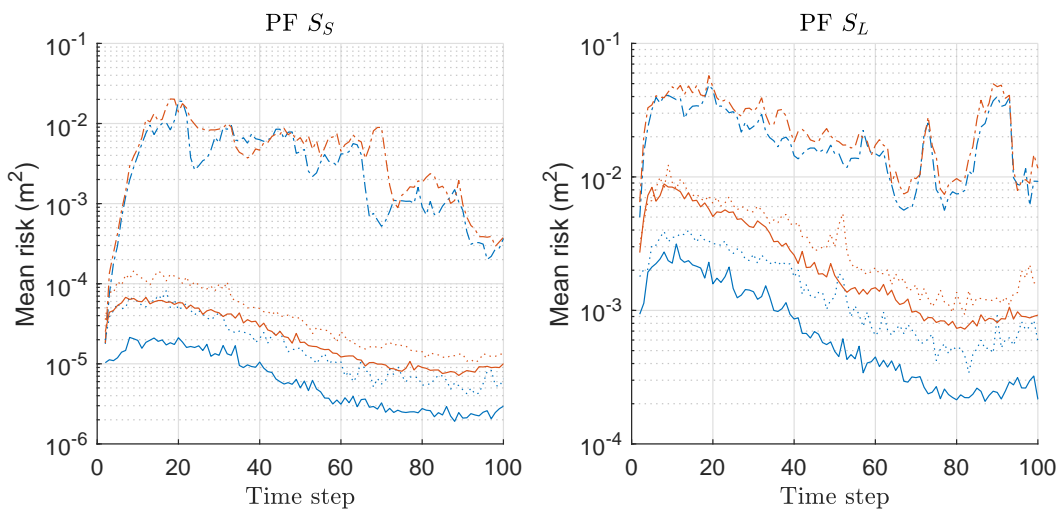


Figure 2. Mean-risk over 300 runs with 1000 particles based on Equation (20) for PF and MPF. Results regarding only r_x are shown, and those regarding r_y showed similar results. Legend: line style (dash-dot: $M = 2$; dotted: $M = 5$; solid: $M = 10$); (line color: blue: PF; red: MPF).

We computed the mean-variance of the weights of particles for both PF and MPF associated with the MDE/risk-results of Figures 1 and 2, and the mean values over 300 runs are shown in Figure 3. The overall variances show similar patterns regardless of the scenario, and only depend on M while the variance-gap between two methods are severer as M increases; therefore, the minimax strategy is more effective as M increases, which is related to the MDE performance-result of Figure 1. In all cases, we obtained the significantly reduced variances of the weights of particles by MAPF compared to those of APF. We may keep increasing the size of M to obtain improved performance; however, the performance improvement may be saturated at a certain point with highly intensive computational complexity. Therefore, we confirmed that the proposed minimax strategy is highly effective in this problem with multi-RSS-measurements of sensor networks. Further, we verify the validity of the proposed strategy in more examples of variants of PFs in the following sections.

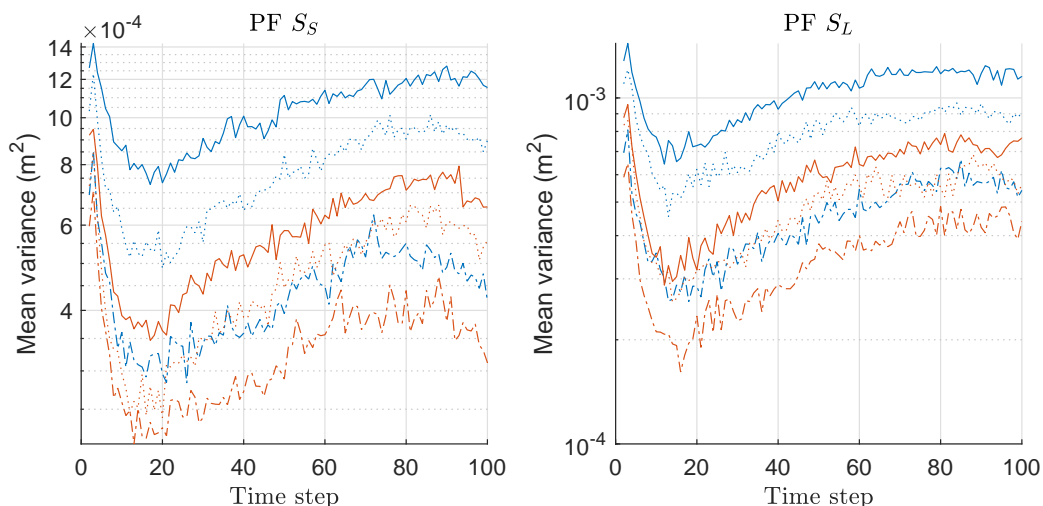


Figure 3. Mean-variance of the weights of particles over 300 runs with 1000 particles for PF and MPF. These results are associated with the results of MDE and mean-risk of Figures 1 and 2. Legend: line style (dash-dot: $M = 2$; dotted: $M = 5$; solid: $M = 10$); (line color: blue: PF; red: MPF).

4.2. Auxiliary Particle Filter (APF)

Compared to SIRPF, the advantage of APF is that it naturally generates particles from the ones at the time step $k - 1$ that are highly close to the true state [40]. In APF, the resampling is performed based on a certain particle estimate μ_k^x that characterizes $p(\psi_k | \psi_{k-1}^x)$. When the process noise is very small, APF tends to be effective because μ_k^x characterizes $p(\psi_k | \psi_{k-1}^x)$ well. Then, APF does not generate outliers often unlike SIRPF, and the weights are more even, which results in APF's better performance than that of SIRPF. Therefore, APF can effectively show the outperforming result compared to SIRPF and is robust against the PI phenomenon due to the resampling procedure. APF returns to previous time steps of currently resampled particles, and then propagates the particles again for the current time step. The weights of the particles are computed based on the ratio of the likelihood functions of the resampled particles and newly propagated particles. We used the measurement noise variances $\sigma_\eta^2 = 10^{-3}$ instead of 10^{-5} in the experiments.

MSED of APF, MAPF, and CRLB is compared in Figure 4. The left of Figure 4 shows the results under the scenario of S_S with various M . Under the scenario S_S , MAPF outperforms APF with any M , and the MSED performance gap is much severer as M increases. Overall, as M increases, the performances of both methods are improved under S_S . The right of Figure 4 shows the results under the scenario of S_L with various M . The overall performances for both methods are worse than those under S_S , and the outperforming results of MAPF over APF are similar to the case under S_S . Overall, MAPF outperforms APF in terms of the MSED performance, and the performance gap is severer as M increases and the minimax strategy is more effective, along with increasing M .

We computed the risk function based on Equation (20) for both APF and MAPF associated with the MSED-result of Figure 4, and the mean values over 300 runs are shown in Figure 5. The left and right of Figure 5 show the mean-risk results under the scenarios of S_S and S_L with various M , respectively. The overall risk is reduced as M increases regardless of the scenario. The gap of the risks between the two methods becomes severer as M increases. We obtained larger risks under S_L compared to those under S_S .

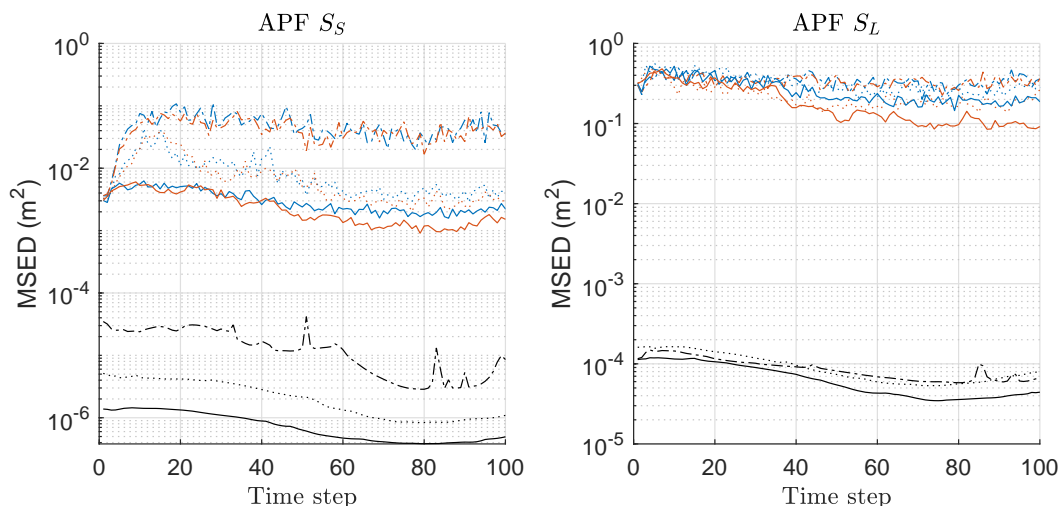


Figure 4. Mean square error of distance (MSED) performance comparison between auxiliary-PF (APF) and minimax APF (MAPF). Three hundred runs were performed with 1000 particles, where M , S_S , S_L denote the number of sensors, the scenario of the small state noise variance, the scenario of the large state noise variance, respectively. Comparison with Cramér-Rao lower bound was also shown, as derived in the Appendix A. Legend: line style (dash-dot: $M = 2$; dotted: $M = 5$; solid: $M = 10$); (line color: blue: APF; red: MAPF; black: CRB).

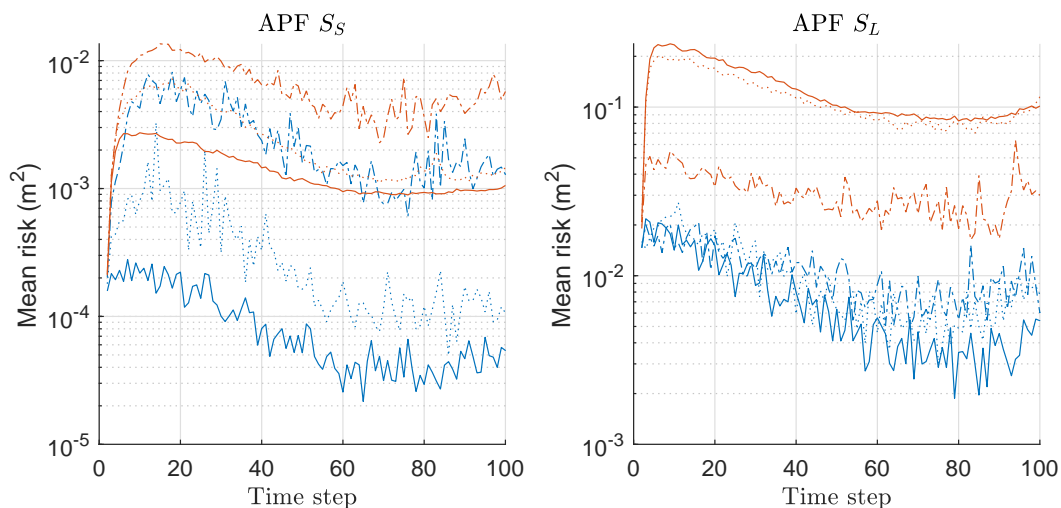


Figure 5. Mean-risk over 300 runs with 1000 particles based on Equation (20) for APF and MAPF. Results regarding only r_x are shown, and those regarding r_y showed similar results. Legend: line style (dash-dot: $M = 2$; dotted: $M = 5$; solid: $M = 10$); (line color: blue: APF; red: MAPF).

We computed the mean-variance of the weights of particles for both APF and MAPF associated with the MSED/risk-results of Figures 4 and 5, and the mean values over 300 runs are shown in Figure 6. For APF, the overall variance is similar regardless of the scenario and the size of M , whereas the variance of MAPF is significantly decreased as M increases. For both APF and MAPF, the variances do not depend on the scenario. Therefore, the variances gap between the two methods becomes severer as M increases. This means that the minimax strategy is more effective as the number of the sensor-measurements increases, which is related to the MSED performance results shown in Figure 4. The result of mean-variance is similar to that of PF/MPF shown in Figure 3, while the variance-gap between the two methods is more significant in the case of APF/MAPF than that in the case of PF/MPF. Therefore, the results show that the proposed minimax strategy is highly effective in the case of APF, as well, in this problem with multi-RSS-measurements of sensor networks.

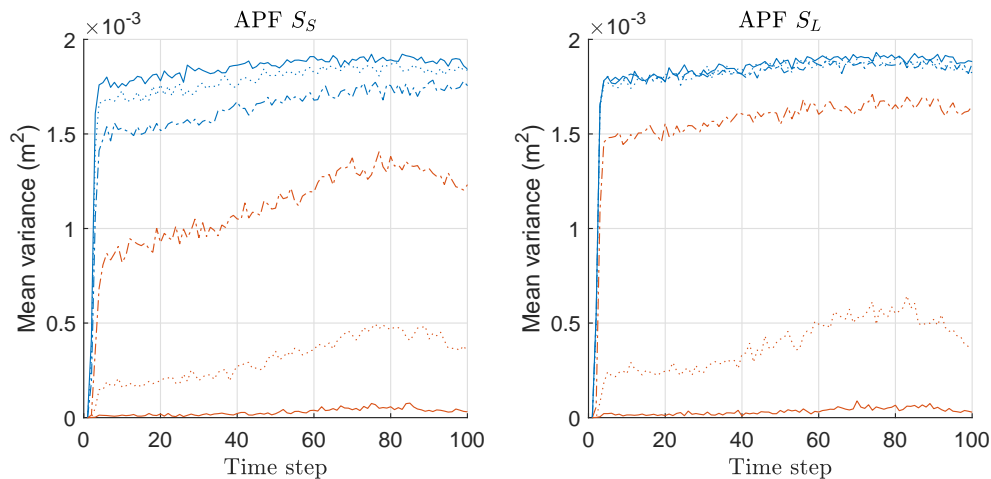


Figure 6. Mean-variance of the weights of particles over 300 runs with 1000 particles for APF and MAPF. These results are associated with the results of mean square error of distance (MSED) and mean-risk of Figures 4 and 5. Legend: line style (dash-dot: $M = 2$; dotted: $M = 5$; solid: $M = 10$); (line color: blue: APF; red: MAPF).

4.3. Regularized PF (RPF)

PI phenomenon is another defect of the SPF besides the degeneracy issue in the practical implementations. The SPF is modified to resolve the PI problem in the RPF algorithm [41]. A kernel density that perturbs the state of particles is employed to develop the diversity of the states of the particles. In RPF, the posterior density is approximated by a rescaled kernel density.

Mean distance error (MDE) of RPF and MRPF is compared in Figure 7. We assess the MDE performance instead of MSED because we could see the result more clearly with MDE rather than MSED in this case of RPF/MRPF. The left of Figure 7 shows the results under the scenario of S_S with various M . Under the S_S scenario, MRPF outperforms RPF when $M = 5, 10$, while the performance is similar when $M = 2$. Overall, as M increases, the performances of both methods are improved. The right of Figure 7 shows the results under the scenario of S_L with various M . The overall performance is worse than that under the S_S scenario, and the performances of the two methods are similar when $M = 2$ under both scenarios. Overall, MRPF outperforms RPF in terms of MDE, and the performance gap becomes severer as M increases, and the minimax strategy becomes more effective.

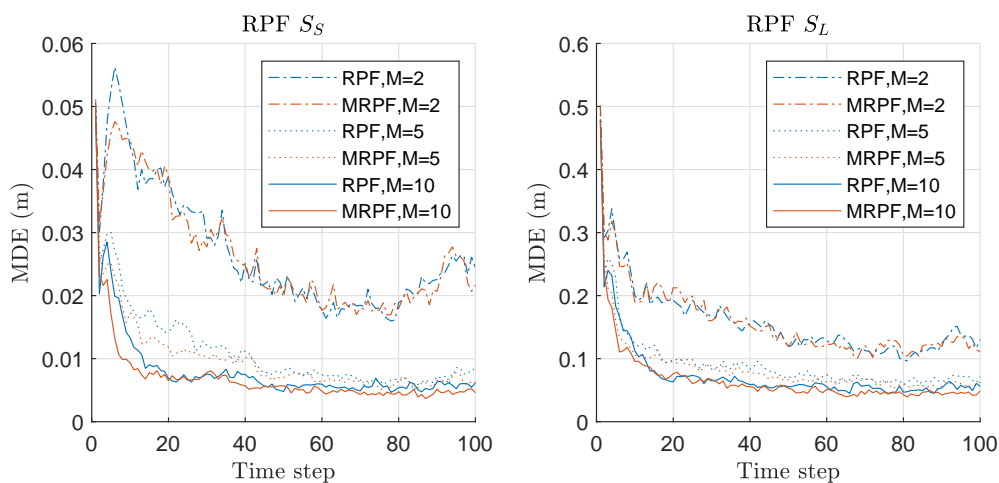


Figure 7. Mean distance error (MDE) performance comparison between regularized PF (RPF) and minimax RPF (MRPF). Three hundred runs were performed with 1000 particles, where M , S_S , and S_L denote the number of sensors, the scenario of the small state noise variance, and the scenario of the large state noise variance, respectively.

We computed, the risk function based on Equation (20) for both RPF and MRPF associated with the MDE-result of Figure 7, and the mean values over 300 runs are shown in Figure 8. The left and right of Figure 8 show the mean-risk results under the scenarios of S_S and S_L with various M , respectively. The overall risk is reduced as M increases regardless of the scenario as in the case of previous results of PFs. The gap of the mean-risks between the two methods becomes severer as M increases. We have larger risks under S_L compared to the case of S_S . The mean-risk result of RPF/MRPF is similar to that of PF/MPF.

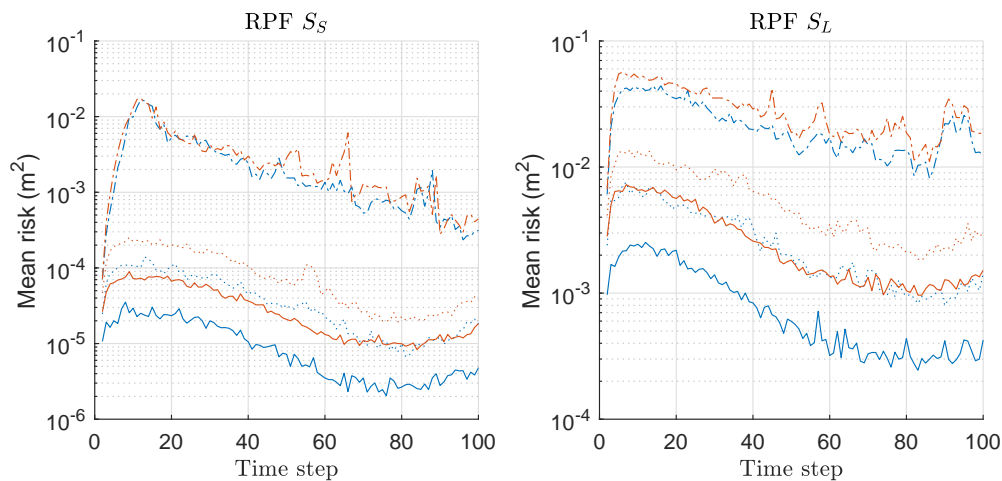


Figure 8. Mean-risk over 300 runs with 1000 particles based on Equation (20) for RPF and MRPF. Results regarding only r_x are shown, and those regarding r_y showed similar results. Legend: line style (dash-dot: $M = 2$; dotted: $M = 5$; solid: $M = 10$); (line color: blue: RPF; red: MRPF).

We computed the mean-variance of the weights of particles for both RPF and MRPF associated with the MDE/risk-results of Figures 7 and 8, and the mean values over 300 runs are shown in Figure 9. For RPF, the variance of RPF increases along with the increased M , whereas that of MRPF remains similarly regardless of the increased M ; therefore, the gap of variance difference is increased along with the increased M regardless of the scenario. This means that the minimax strategy is more effective as M increases, which is related to the MDE performance result of Figure 7. The result of mean-variance is similar to the case of PF/MPF as shown in Figure 3. Therefore, we also confirmed that the proposed minimax strategy is highly effective in the case of RPF with multiple measurements of sensor networks.

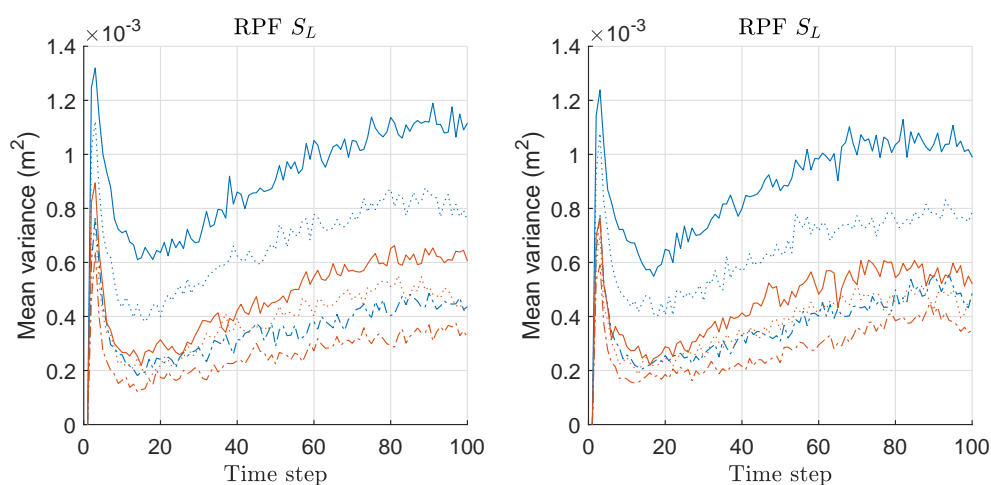


Figure 9. Mean-variance of the weights of particles over 300 runs with 1000 particles for RPF and MRPF. These results are associated with the results of MDE and mean-risk of Figures 7 and 8. Legend: line style (dash-dot: $M = 2$; dotted: $M = 5$; solid: $M = 10$); (line color: blue: RPF; red: MRPF).

4.4. Kullback-Leibler Divergence PF (KLDPF)

In KLDPF, the number of employed particles is optimized based on a predetermined error bound at each time step [23]. The number of employed particles is adjusted to reduce redundant particles and also unnecessary computations, accordingly, while the performance is not affected. In this assessment, we apply the error bound of 0.001, the initial number of employed particles is 500 and the maximum number of particles is bounded by 1000; the probability bound is 0.01, the bin size is $1/2 \times \sqrt{\sigma_\theta} \times 4(L)$; $1/2$ is determined from $T_s/2$ in E . Although KLDPF adaptively optimizes the number of particles at every time step, the algorithm inherently requires a higher computational cost than those of other PF variants.

MDE of KLDPF and MKLDPF is compared in Figure 10. The left of Figure 10 shows the results under the scenario of S_S with various M . Under the S_S scenario, MKLDPF outperforms KLDPF when M is equal to 5 or larger, while the performance is similar to each other when $M = 2$. Overall, MKLDPF outperforms KLDPF when M is increased above 2. The right of Figure 10 shows the results under the scenario of S_L with various M . The overall results are worse than those under S_S for both methods. The superiority of MKLDPF over KLDPF is clear when $M = 5, 10$. When $M = 2$ under S_L , the performance difference between KLDPF and MKLDPF is not clear as in the case of S_S . Overall, MKLDPF outperforms KLDPF in terms of MDE, and the performance gap is severer and the minimax strategy is more effective as M increases. The pattern of the MDE result of KLDPF/MKLDPF shown in Figure 10 is highly similar to that of RPF/MRPF shown in Figure 7.

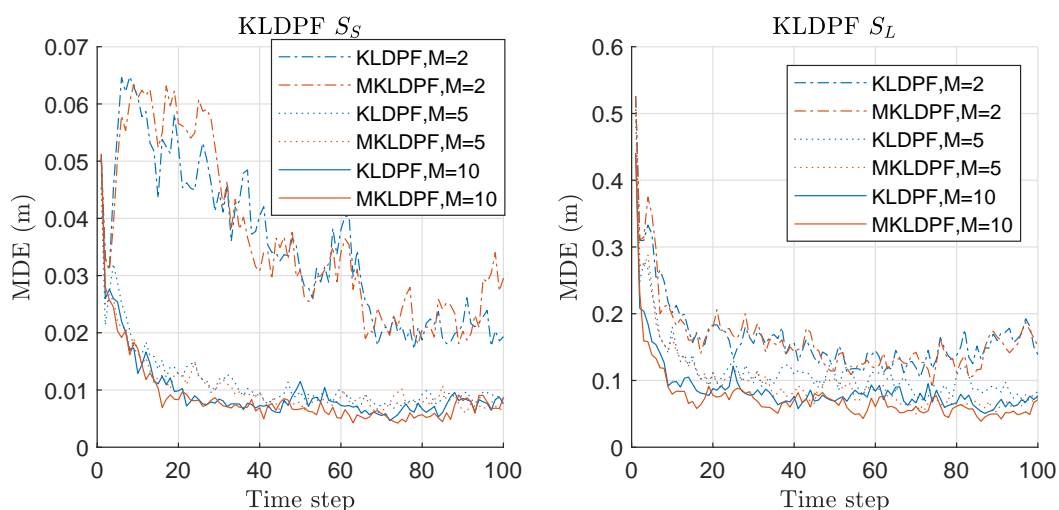


Figure 10. Mean distance error (MDE) performance comparison between Kullback–Leibler divergence-PF (KLDPF) and minimax KLDPF (MKLDPF). Three hundred runs were performed with 1000 particles, where M , S_S , and S_L denote the number of sensors, the scenario of the small state noise variance, and the scenario of the large state noise variance, respectively.

We computed the risk function based on Equation (20) for both KLDPF and MKLDPF associated with the MDE-result of Figure 10, and the mean values over 300 runs are shown in Figure 11. The left and right of Figure 11 show the mean-risk results under the scenarios of S_S and S_L with various M , respectively. The overall risk decreases as M increases regardless of the scenario. The gap of the risks between the two methods is severer as M increases. We obtain larger risks under S_L compared to the case of S_S regardless of the method and M . The MDE-result of Figure 10 was obtained by minimizing the risk in Figure 11. The pattern of the mean-risk-result shown in Figure 11 is also highly similar to that of RPF/MRPF shown in Figure 8.

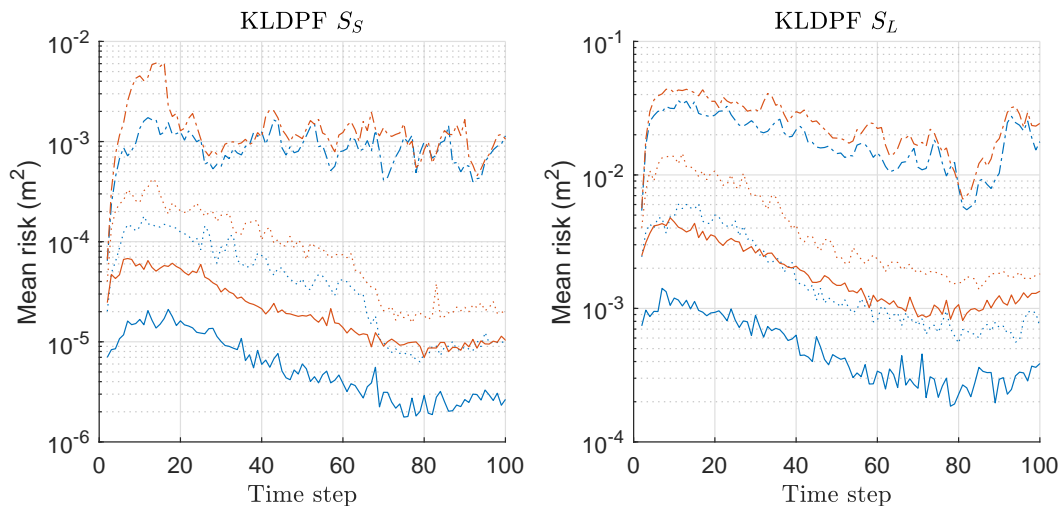


Figure 11. Mean-risk over 300 runs with 1000 particles based on Equation (20) for KLDPF and MKLDPF. Results regarding only r_x are shown, and those regarding r_y showed similar results. Legend: line style (dash-dot: $M = 2$; dotted: $M = 5$; solid: $M = 10$); (line color: blue: KLDPF; red: MKLDPF).

We computed the mean-variance of the weights of particles for both KLDPF and MKLDPF associated with the MDE/risk-results of Figures 10 and 11, and the mean values over 300 runs are shown in Figure 12. The gap of the variances between the two methods becomes severer as M increases regardless of the scenario. This means that the minimax strategy is more effective as M increases, which is closely related to the MDE performance result of Figure 10. In all cases, we obtained the significantly reduced variance of the weights of particles by the minimax strategy as shown in Figure 12. The pattern of all the results of MDE-performance, mean-risk, mean-variance of KLDPF/MKLDPF are highly similar to that of RPF/MRPF, while the computational complexities of RPF and KLDPF are quite different, as we show in the result of it shortly. Therefore, we confirmed that the proposed minimax strategy is highly effective in the case of KLDPF, as well.

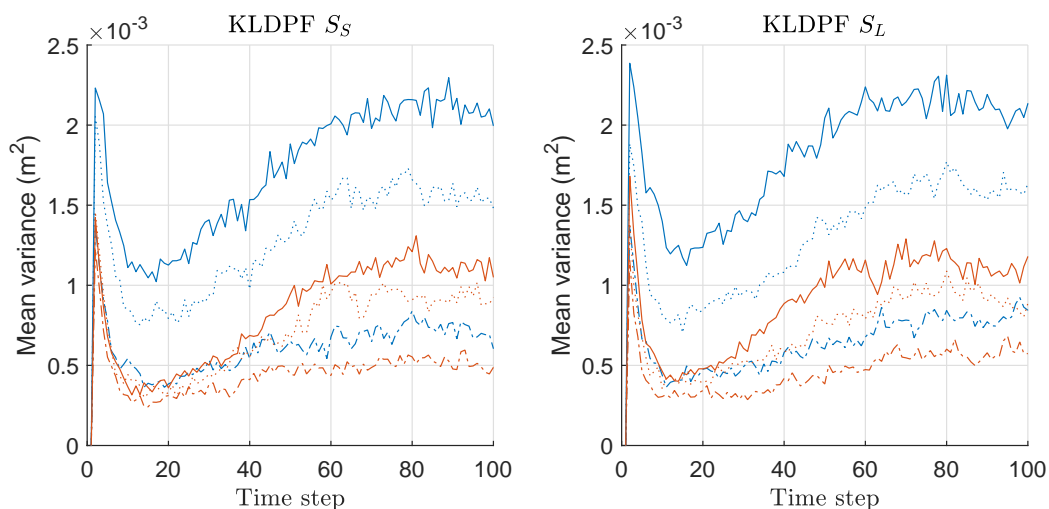


Figure 12. Mean-variance of the weights of particles over 300 runs with 1000 particles for KLDPF and MKLDPF. These results are associated with the results of MDE and mean-risk of Figures 10 and 11. Legend: line style (dash-dot: $M = 2$; dotted: $M = 5$; solid: $M = 10$); (line color: blue: KLDPF; red: MKLDPF).

4.5. Gaussian PF (GPF)

In the GPF algorithm, an additional particle generation step is required prior to the particle propagation step. That is, particles are generated based on a Gaussian distribution where the mean value and the covariance are adopted from the previous estimate ($\hat{\psi}_{k-1}$) and the weighted sample covariance, $C_{k-1} = \sum_{\kappa=1}^N \omega_{k-1}^{\kappa} (\psi_{k-1}^{\kappa} - \hat{\psi}_{k-1}) (\psi_{k-1}^{\kappa} - \hat{\psi}_{k-1})^{\top}$. Then, particles are propagated by the prior density. There is no resampling process in the GPF algorithm, which significantly reduces its computational complexity. For details about GPF, refer to Reference [26].

MDE of GPF and MGPF is compared in Figure 13. The left and right of Figure 13 show the results under the scenarios of S_S and S_L with various M , respectively. Overall, MGPF slightly outperforms GPF when M is increased above 2. The overall results under S_L are worse than those under S_S . The superiority of MGPF over GPF is not as much as that of the other PF variants that have been investigated so far. Nonetheless, overall, MGPF outperforms GPF in terms of MDE performance when we increase the number of employed sensors. It needs to be noted that GPF does not require the resampling process that also incurs the side effect, such as particle impoverishment. Therefore, the minimax strategy is not as effective as the cases of the other PF variants.

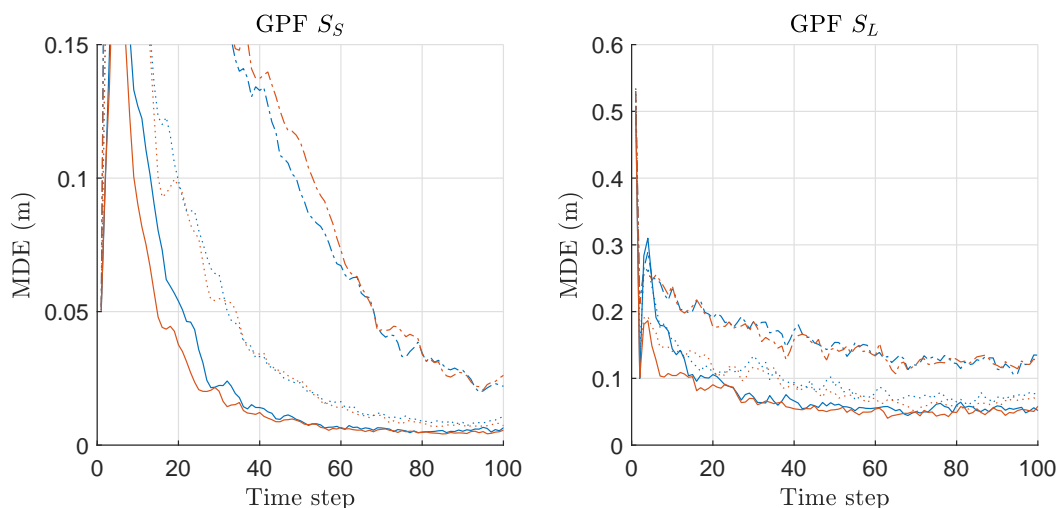


Figure 13. Mean distance error (MDE) performance comparison between Gaussian-PF (GPF) and minimax GPF (MGPF). Three hundred runs were performed with 1000 particles, where M , S_S , S_L denote the number of sensors, the scenario of the small state noise variance, the scenario of the large state noise variance, respectively. Legend: line style (dash-dot: $M = 2$; dotted: $M = 5$; solid: $M = 10$); (line color: blue: GPF; red: MGPF).

We computed the risk function based on Equation (20) for both GPF and MGPF associated with the MDE-result of Figure 13, and the mean values over 300 runs are shown in Figure 14.

The left and right of Figure 14 show the mean-risk results under the scenarios of S_S and S_L with various M , respectively. The overall risk decreases as M increases, regardless of the scenario, like the previously shown results of the other PFs. The relative gap of the risks between the two methods becomes severer as M increases for both scenarios. Overall, we have larger risks under S_L compared to the case of S_S ;

We also note that the relative gap of the mean-risks is the severest when $M = 10$ under S_L among all results in Figure 14.

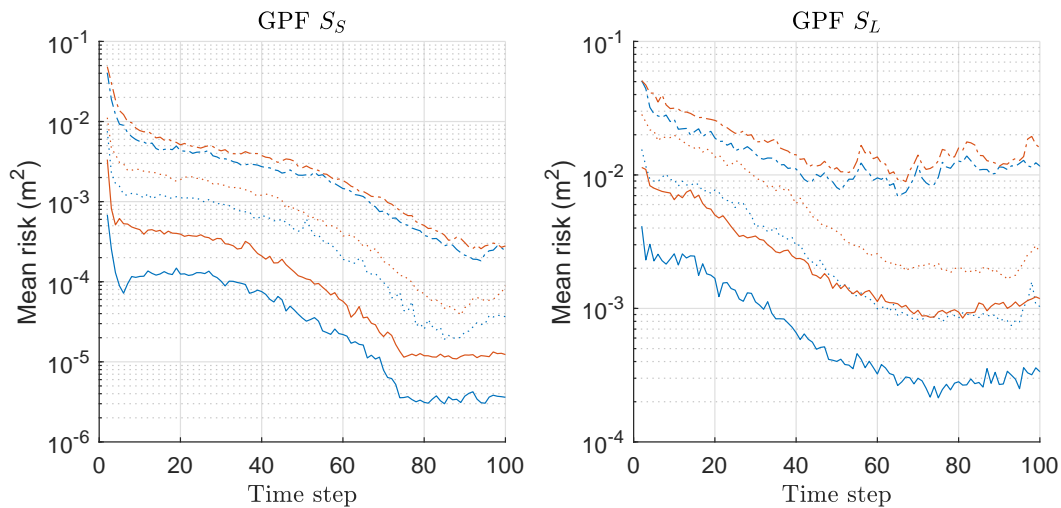


Figure 14. Mean-risk over 300 runs with 1000 particles based on Equation (20) for GPF and MGPF. Results regarding only r_x are shown, and those regarding r_y showed similar results. Legend: line style (dash-dot: $M = 2$; dotted: $M = 5$; solid: $M = 10$); (line color: blue: GPF; red: MGPF).

We computed the mean-variance of the weights of particles for both GPF and MGPF associated with the MDE/risk-results of Figures 13 and 14, and the mean values over 300 runs are shown in Figure 15. The gap of the variances between two methods becomes severer as M increases in both scenarios, and the relative gap is slightly severer under S_L compared to the case under S_S . This means that the minimax strategy is more effective as M and σ_θ^2 increases. The pattern of the overall variance-results of GPF and MGPF shown in Figure 15 is highly similar to those of RPF/MRPF and KLDPF/MKLDPF shown in Figures 9 and 12. The pattern of all results of MDE performance, mean-risk, and mean-variance of GPF/MGPF is highly similar to those of RPF/MRPF and KLDPF/MKLDPF. Therefore, we confirmed that the proposed minimax strategy is effective in the case of GPF, as well, although the effect is not as much as that in the other PF variants.

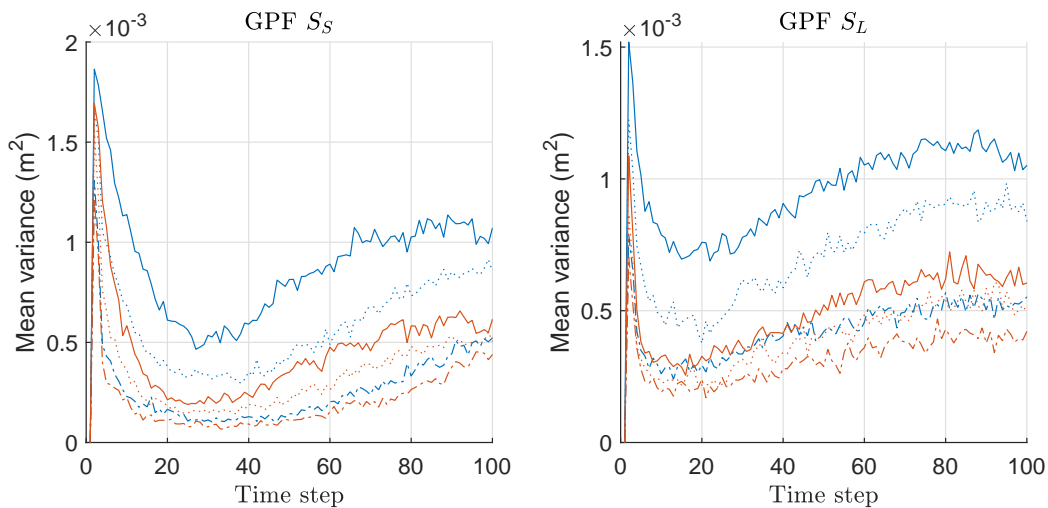


Figure 15. Mean-variance of the weights of particles over 300 runs with 1000 particles for GPF and MGPF. These results are associated with the results of MDE and mean-risk of Figures 13 and 14. Legend: line style (dash-dot: $M = 2$; dotted: $M = 5$; solid: $M = 10$); (line color: blue: GPF; red: MGPF).

4.6. Processing Time

For all methods, we measured the processing time in terms of the average-elapsed-time for the one time step out of T during the MATLAB simulations, and the result is shown in Figure 16. The top

and bottom figures are concerning two and ten sensors, respectively, in the networks. The elapsed time concerning ten sensors is not five times longer than the case of two sensors but elapsed approximately between 1.5 and 2 times longer. As clearly seen, KLDPF requires the most duration of elapsed time among all the PFs. GPF and MGPF require the least duration of elapsed time because GPF does not require the resampling process. There is no significant time difference between minimax and original PF variants. The minimax strategy affects the processing time in terms of the computation of finding the minimum value of the weights among M , while original PF variants require the computation of joint probability density function of M densities; therefore, overall computational complexities of two frameworks do not show significant time-difference. The minimax PFs do not require the computation of the joint probability density, which can be intractable sometimes.

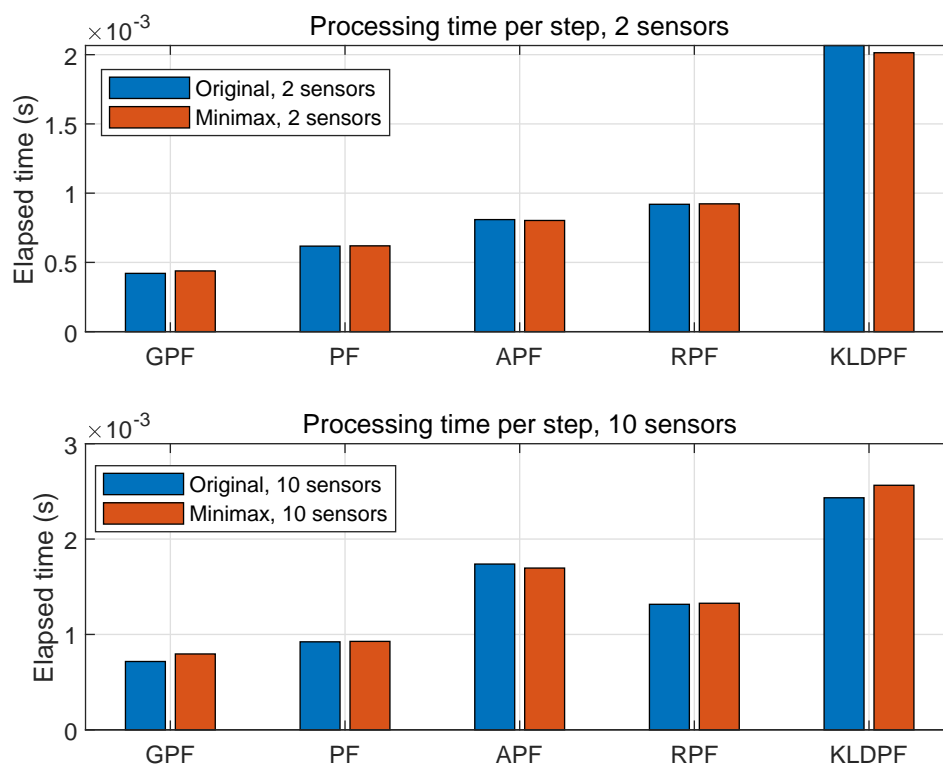


Figure 16. Mean-elapsed time for the one time step over 300 runs with 1000 particles for all PFs. Note that there is no resampling process required for GPF.

4.7. Discussion

In all PF variants, the minimax versions showed better tracking performance than original PF variants in terms of MSED/MDE, while both frameworks showed similar processing-time-performance during the MATLAB experiments. The experimental simulation results can be summarized as follows. Concerning MSED/MDE performance, overall: both original and minimax PFs showed similar performances in terms of MSED/MDE with two sensors, while minimax PFs clearly outperformed original PFs with sensors over two. Besides, the performance gap is slightly more evident under S_L compared to that under S_S due to the severer degeneracy problem under S_L . Nevertheless, the performance gap was not significantly different between the results with five sensors and those with ten sensors. Concerning the mean-risk, overall, the risk was the lowest with the maximum number of sensors under S_S , while the difference gap between two PF-frameworks is relatively the largest with the maximum number of sensors regardless of the scenario. When two sensors are employed, unlike the case of the MSED/MDE-performance, the mean-risk of the minimax PFs is clearly higher than that of original PF variants. Concerning the mean-variance of the weights of particles, the

results only depend on the number of employed sensors regardless of the scenario, and the gap of the mean-variance is severer as M increases due to the effect of minimax strategy that results in the improved tracking performance.

5. Conclusions

In this paper, we proposed the minimax PF for the problem of tracking a target based on RSS measurements in wireless sensor networks where a large number of sensor-measurements are available. We verified the validity of the proposed minimax strategy in various scenarios in terms of M (the number of sensors) and σ_{θ}^2 (the variance of the state noise) by applying to various variants of PF. The adopted minimax strategy is more effective when we employ a large number of the sensors in the networks compared to the case of a small number of the sensors employed; therefore, we obtained the significantly reduced variance of the weights of particles that derives the robustness against the degeneracy problem of generic PF. The robustness against the degeneracy resulted in the improved tracking performance beyond the asymptotically optimal performance of the original PF variants. Furthermore, the strategy is more effective when the dynamic state-space model varies with a larger variance, where the particles undergo a severer degeneracy problem. It needs to be noted that the minimax strategy is effective on condition that the signal-to-noise ratio is above a certain value that is good enough for tracking. Otherwise, MPFs may not outperform regular PFs.

Author Contributions: Conceptualization, J.L. and H.-M.P.; methodology, J.L.; software, J.L. and H.-M.P.; validation, J.L.; formal analysis, J.L. and H.-M.P.; investigation, J.L. and H.-M.P.; resources, J.L. and H.-M.P.; data curation, J.L.; writing—original draft preparation, J.L.; writing—review and editing, J.L. and H.-M.P.; visualization, J.L.; supervision, J.L. and H.-M.P.; project administration, J.L. and H.-M.P.; funding acquisition, J.L. and H.-M.P. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Basic Science Research Program through the National Research Foundation(NRF) funded by the Korea government(MIST) (No. NRF-2019R111A1A01058976) and Institute of Information & communications Technology Planning & evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2019-0-01376, Development of the multi-speaker conversational speech recognition technology).

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Appendix A. Derivation of Cramér-Rao Lower Bound

The variance of any unbiased estimate, $\hat{\boldsymbol{\psi}}$ is bounded as follows:

$$\text{Var}(\hat{\boldsymbol{\psi}}_e) \geq \left[\mathbf{I}^{-1}(\boldsymbol{\psi}) \right]_{ee}, \text{ for } e = 1, \dots, L, \quad (\text{A1})$$

where \mathbf{I} is the Fisher information matrix, $\boldsymbol{\psi}$ is a vector state, e is the element index, and L is the dimension of the state vector. When the measurement noise is Gaussian and the measurement is given by

$$m \sim \mathcal{N}(g(\boldsymbol{\psi}), \mathbf{C}_{\eta}(\boldsymbol{\psi})), \quad (\text{A2})$$

we can obtain the following, for $e, l = 1, \dots, L$,

$$\begin{aligned} [\mathbf{I}(\boldsymbol{\psi})]_{el} &= \left[\frac{\partial g(\boldsymbol{\psi})}{\partial \psi_e} \right]^{\top} \mathbf{C}_{\eta}^{-1}(\boldsymbol{\psi}) \left[\frac{\partial g(\boldsymbol{\psi})}{\partial \psi_l} \right] + \\ &\quad \frac{1}{2} \text{tr} \left[\left(\mathbf{C}_{\eta}^{-1}(\boldsymbol{\psi}) \frac{\partial \mathbf{C}_{\eta}(\boldsymbol{\psi})}{\partial \psi_e} \mathbf{C}_{\eta}^{-1}(\boldsymbol{\psi}) \frac{\partial \mathbf{C}_{\eta}(\boldsymbol{\psi})}{\partial \psi_l} \right) \right] \end{aligned} \quad (\text{A3})$$

where

$$\frac{\partial g(\boldsymbol{\psi})}{\partial \psi_i} = \left[\frac{\partial [g(\boldsymbol{\psi})]_1}{\partial \psi_i}, \frac{\partial [g(\boldsymbol{\psi})]_2}{\partial \psi_i}, \dots, \frac{\partial [g(\boldsymbol{\psi})]_M}{\partial \psi_i} \right]^{\top}, \quad (\text{A4})$$

$$\frac{\partial \mathbf{C}_\eta(\boldsymbol{\psi})}{\partial \psi_i} = \begin{bmatrix} \frac{\partial [\mathbf{C}_\eta(\boldsymbol{\psi})]_{11}}{\partial \psi_i} & \frac{\partial [\mathbf{C}_\eta(\boldsymbol{\psi})]_{12}}{\partial \psi_i} & \dots & \frac{\partial [\mathbf{C}_\eta(\boldsymbol{\psi})]_{1M}}{\partial \psi_i} \\ \frac{\partial [\mathbf{C}_\eta(\boldsymbol{\psi})]_{21}}{\partial \psi_i} & \frac{\partial [\mathbf{C}_\eta(\boldsymbol{\psi})]_{22}}{\partial \psi_i} & \dots & \frac{\partial [\mathbf{C}_\eta(\boldsymbol{\psi})]_{2M}}{\partial \psi_i} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial [\mathbf{C}_\eta(\boldsymbol{\psi})]_{M1}}{\partial \psi_i} & \frac{\partial [\mathbf{C}_\eta(\boldsymbol{\psi})]_{M2}}{\partial \psi_i} & \dots & \frac{\partial [\mathbf{C}_\eta(\boldsymbol{\psi})]_{MM}}{\partial \psi_i} \end{bmatrix}, \quad (\text{A5})$$

$\mathbf{g}(\boldsymbol{\psi}) = [g_1 \dots g_M]^\top$, and the covariance matrix $\mathbf{C}_\eta(\boldsymbol{\psi}) = \text{diag}(\sigma_{\eta_1}^2 \dots \sigma_{\eta_M}^2)$, where $\text{diag}(\cdot)$ denotes a diagonal matrix, and the variance (power) of the measurement noise η_i :

$$\sigma_{\eta_i}^2 = g_i^2 \cdot 10^{(-\text{SNR}_{m_i}/10)}, \quad (\text{A6})$$

where SNR_{m_i} is the signal-to-noise ratio (SNR) in dB for the corresponding m_i , and are computed as follows:

$$\text{SNR}_{m_i} = 10 \log_{10} \left[\frac{g_i^2}{\sigma_{\eta_i}^2} \right]. \quad (\text{A7})$$

The Fisher information is a 4×4 matrix, and $[\mathbf{I}(\boldsymbol{\psi})]_{11}$ and $[\mathbf{I}(\boldsymbol{\psi})]_{22}$ are the corresponding elements for r_x and r_y , respectively. Therefore,

$$\text{Var}(\hat{r}_x) \geq [\mathbf{I}^{-1}(\boldsymbol{\psi})]_{11}, \quad \text{Var}(\hat{r}_y) \geq [\mathbf{I}^{-1}(\boldsymbol{\psi})]_{22}. \quad (\text{A8})$$

If CRLB for the distance estimation, i.e., $\sqrt{r_x^2 + r_y^2}$ is to be computed, we can use vector parameter CRLB for transformations, and is easily derived as follows [42]. If we define $\boldsymbol{\alpha} = h(\boldsymbol{\psi}) = \sqrt{r_x^2 + r_y^2}$, CRLB is derived as

$$\text{Var}(\hat{\boldsymbol{\alpha}}) \geq \frac{\partial h(\boldsymbol{\psi})}{\partial \boldsymbol{\psi}} \mathbf{I}^{-1}(\boldsymbol{\psi}) \frac{\partial h(\boldsymbol{\psi})^\top}{\partial \boldsymbol{\psi}}, \quad (\text{A9})$$

where a Jacobian matrix $\frac{\partial h(\boldsymbol{\psi})}{\partial \boldsymbol{\psi}}$ is described as

$$\frac{\partial h(\boldsymbol{\psi})}{\partial \boldsymbol{\psi}} = \begin{bmatrix} \frac{\partial h(\boldsymbol{\psi})}{\partial \psi_1} & \frac{\partial h(\boldsymbol{\psi})}{\partial \psi_2} & \dots & \frac{\partial h(\boldsymbol{\psi})}{\partial \psi_L} \end{bmatrix}. \quad (\text{A10})$$

References

- Zanella, A.; Bui, N.; Castellani, A.; Vangelista, L.; Zorzi, M. Internet of things for smart cities. *IEEE Internet Things J.* **2014**, *1*, 22–32. [\[CrossRef\]](#)
- Lin, J.; Yu, W.; Zhang, N.; Yang, X.; Zhang, H.; Zhao, W. A survey on internet of things: Architecture, enabling technologies, security and privacy, and applications. *IEEE Internet Things J.* **2017**, *4*, 1125–1142. [\[CrossRef\]](#)
- Kuutti, S.; Fallah, S.; Katsaros, K.; Dianati, M.; McCullough, F.; Mouzakitis, A. A survey of the state-of-the-art localization techniques and their potentials for autonomous vehicle applications. *IEEE Internet Things J.* **2018**, *5*, 829–846. [\[CrossRef\]](#)
- Ramnath, S.; Javali, A.; Narang, B.; Mishra, P.; Routray, S.K. IoT based localization and tracking. In Proceedings of the 2017 International Conference on IoT and Application (ICIOT), Nagapattinam, India, 19–20 May 2017; pp. 1–4.
- Paul, A.; Sato, T. Localization in wireless sensor networks: a survey on algorithms, measurement techniques, applications and challenges. *J. Sens. Actuator Netw.* **2017**, *6*, 24. [\[CrossRef\]](#)
- Yang, Q.; Yoo, S.J. Optimal UAV path planning: Sensing data acquisition over IoT sensor networks using multi-objective bio-inspired algorithms. *IEEE Access* **2018**, *6*, 13671–13684. [\[CrossRef\]](#)
- Wu, P.; Su, S.; Zuo, Z.; Guo, X.; Sun, B.; Wen, X. Time Difference of Arrival (TDoA) Localization Combining Weighted Least Squares and Firefly Algorithm. *Sensors* **2019**, *19*, 2554. [\[CrossRef\]](#) [\[PubMed\]](#)

8. Jin, B.; Xu, X.; Zhang, T. Robust time-difference-of-arrival (TDOA) localization using weighted least squares with cone tangent plane constraint. *Sensors* **2018**, *18*, 778.
9. Chen, Z.; Wang, J. ES-DPR: A DOA-Based Method for Passive Localization in Indoor Environments. *Sensors* **2019**, *19*, 2482. [[CrossRef](#)]
10. Aboutanios, E.; Hassanien, A.; El-Keyi, A.; Nasser, Y.; Vorobyov, S.A. Advances in DOA Estimation and Source Localization. *Int. J. Antennas Propag.* **2017**, *2017*, 1–3. [[CrossRef](#)]
11. Ying, J.; Pahlavan, K. Precision of RSS-Based Localization in the IoT. *Int. J. Wireless Inf. Netw.* **2019**, *26*, 10–23. [[CrossRef](#)]
12. Tomic, S.; Beko, M.; Dinis, R. RSS-based localization in wireless sensor networks using convex relaxation: Noncooperative and cooperative schemes. *IEEE Trans. Veh. Technol.* **2014**, *64*, 2037–2050. [[CrossRef](#)]
13. Kim, S.; Chong, J.W. An efficient TDOA-based localization algorithm without synchronization between base stations. *Int. J. Distrib. Sens. Netw.* **2015**, *11*, 832351. [[CrossRef](#)]
14. Sari, R.; Zayyani, H. RSS localization using unknown statistical path loss exponent model. *IEEE Commun. Lett.* **2018**, *22*, 1830–1833. [[CrossRef](#)]
15. Achroufene, A.; Amirat, Y.; Chibani, A. RSS-based indoor localization using belief function theory. *IEEE Trans. Autom. Sci. Eng.* **2018**, *16*, 1163–1180. [[CrossRef](#)]
16. Fu, Y.; Wang, C.; Liu, R.; Liang, G.; Zhang, H.; Ur Rehman, S. Moving object localization based on UHF RFID phase and laser clustering. *Sensors* **2018**, *18*, 825. [[CrossRef](#)] [[PubMed](#)]
17. Chen, J.; Zhang, Y.; Xue, W. Unsupervised indoor localization based on Smartphone Sensors, iBeacon and Wi-Fi. *Sensors* **2018**, *18*, 1378. [[CrossRef](#)]
18. Mahapatra, R.K.; Shet, N. Localization based on RSSI exploiting gaussian and averaging filter in wireless sensor network. *Arabian J. Mater. Sci. Eng.* **2018**, *43*, 4145–4159. [[CrossRef](#)]
19. Sun, W.; Xue, M.; Yu, H.; Tang, H.; Lin, A. Augmentation of fingerprints for indoor WiFi localization based on Gaussian process regression. *IEEE Trans. Veh. Technol.* **2018**, *67*, 10896–10905. [[CrossRef](#)]
20. Liu, J.S.; Chen, R. Sequential Monte Carlo methods for dynamic systems. *J. Am. Stat. Assoc.* **1998**, *93*, 1032–1044. [[CrossRef](#)]
21. Guarniero, P.; Johansen, A.M.; Lee, A. The iterated auxiliary particle filter. *J. Am. Stat. Assoc.* **2017**, *112*, 1636–1647. [[CrossRef](#)]
22. Murangira, A.; Musso, C.; Dahia, K. A mixture regularized rao-blackwellized particle filter for terrain positioning. *IEEE Trans. Aerosp. Electron. Syst.* **2016**, *52*, 1967–1985. [[CrossRef](#)]
23. Fox, D. Adapting the sample size in particle filters through KLD-sampling. *Int. J. Rob. Res.* **2003**, *22*, 985–1003. [[CrossRef](#)]
24. Li, T.; Sun, S.; Sattar, T.P. Adapting sample size in particle filters through KLD-resampling. *Electron. Lett.* **2013**, *49*, 740–742. [[CrossRef](#)]
25. Zhou, T.; Peng, D.; Xu, C.; Zhang, W.; Shen, J. Adaptive particle filter based on Kullback–Leibler distance for underwater terrain aided navigation with multi-beam sonar. *IET Radar Sonar Navig.* **2018**, *12*, 433–441. [[CrossRef](#)]
26. Kotecha, J.H.; Djuric, P.M. Gaussian particle filtering. *IEEE Trans. Signal Process.* **2003**, *51*, 2592–2601. [[CrossRef](#)]
27. Lim, J.; Hong, D. Gaussian particle filtering approach for carrier frequency offset estimation in OFDM systems. *IEEE Signal Process Lett.* **2013**, *20*, 367–370. [[CrossRef](#)]
28. Míguez, J.; Bugallo, M.F.; Djurić, P.M. A new class of particle filters for random dynamic systems with unknown statistics. *EURASIP J. Adv. Sign. Proces.* **2004**, *2004*, 1–17. [[CrossRef](#)]
29. Lim, J.; Hong, D. Cost reference particle filtering approach to high-bandwidth tilt estimation. *IEEE Trans. Ind. Electron.* **2010**, *57*, 3830–3839. [[CrossRef](#)]
30. Lim, J. Particle filtering for nonlinear dynamic state systems with unknown noise statistics. *Nonlinear Dyn.* **2014**, *78*, 1369–1388. [[CrossRef](#)]
31. Arulampalam, M.; Maskell, S.; Gordon, N.; Clapp, T. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Trans. Signal Process.* **2002**, *50*, 174–188
32. Li, T.; Bolic, M.; Djuric, P.M. Resampling methods for particle filtering: classification, implementation, and strategies. *IEEE Signal Process Mag.* **2015**, *32*, 70–86. [[CrossRef](#)]

33. Li, T.C.; Villarrubia, G.; Sun, S.d.; Corchado, J.M.; Bajo, J. Resampling methods for particle filtering: identical distribution, a new method, and comparable study. *Front. Inf. Technol. Electron. Eng.* **2015**, *16*, 969–984. [[CrossRef](#)]
34. Lim, J.; Kim, H.S.; Park, H.M. Minimax particle filtering for tracking a highly maneuvering target. *Int. J. Robust Nonlinear Control* **2020**, *30*, 636–651. [[CrossRef](#)]
35. Lim, J.; Kim, H.S.; Park, H.M. Interactive-Multiple-Model Algorithm based on Minimax Particle Filtering. *IEEE Signal Process Lett.* **2020**, *27*, 36–40. [[CrossRef](#)]
36. Gustafsson, F.; Gunnarsson, F.; Bergman, N.; Forssell, U.; Jansson, J.; Karlsson, R.; Nordlund, P.J. Particle filters for positioning, navigation, and tracking. *IEEE Trans. Signal Process.* **2002**, *50*, 425–437. [[CrossRef](#)]
37. Djuric, P.M.; Vemula, M.; Bugallo, M.F. Target tracking by particle filtering in binary sensor networks. *IEEE Trans. Signal Process.* **2008**, *56*, 2229–2238. [[CrossRef](#)]
38. Lim, J. A target tracking based on bearing and range measurement with unknown noise statistics. *J. Electr. Eng. Technol.* **2013**, *8*, 1520–1529. [[CrossRef](#)]
39. Patwari, N.; Hero III, A.O.; Perkins, M.; Correal, N.S.; O’Dea, R.J. Relative location estimation in wireless sensor networks. *IEEE Trans. Signal Process.* **2003**, *51*, 2137–2148. [[CrossRef](#)]
40. Pitt, M.K.; Shephard, N. Filtering via simulation: Auxiliary particle filters. *J. Am. Stat. Assoc.* **1999**, *94*, 590–599. [[CrossRef](#)]
41. Lim, J. Performance degradation due to particle impoverishment in particle filtering. *J. Electr. Eng. Technol.* **2014**, *9*, 2107–2113. [[CrossRef](#)]
42. Kay, S.M. *Fundamentals of Statistical Signal Processing; Estimation Theory*; Prentice Hall PTR: Upper Saddle River, NJ, USA, 1993; Volume 1.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).