

Article

Adaptive Particle Filter for Nonparametric Estimation with Measurement Uncertainty in Wireless Sensor Networks

Xiaofan Li ^{1,2,†}, Yubin Zhao ^{3,*}, Sha Zhang ^{1,2,†} and Xiaopeng Fan ³

¹ The State Monitoring Center and Testing Center, Beijing 100037, China; lixiaofan@srtc.org.cn (X.L.); zhangsha@srtc.org.cn (S.Z.)

² Shenzhen Institute of Radio Testing & Tech., Shenzhen 518000, China

³ Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China; xp.fan@siat.ac.cn

* Correspondence: zhaoyb@siat.ac.cn; Tel.: +86-755-8639-2359

† These authors contributed equally to this work.

Academic Editor: Davide Brunelli

Received: 27 February 2016; Accepted: 20 May 2016; Published: 30 May 2016

Abstract: Particle filters (PFs) are widely used for nonlinear signal processing in wireless sensor networks (WSNs). However, the measurement uncertainty makes the WSN observations unreliable to the actual case and also degrades the estimation accuracy of the PFs. In addition to the algorithm design, few works focus on improving the likelihood calculation method, since it can be pre-assumed by a given distribution model. In this paper, we propose a novel PF method, which is based on a new likelihood fusion method for WSNs and can further improve the estimation performance. We firstly use a dynamic Gaussian model to describe the nonparametric features of the measurement uncertainty. Then, we propose a likelihood adaptation method that employs the prior information and a belief factor to reduce the measurement noise. The optimal belief factor is attained by deriving the minimum Kullback–Leibler divergence. The likelihood adaptation method can be integrated into any PFs, and we use our method to develop three versions of adaptive PFs for a target tracking system using wireless sensor network. The simulation and experimental results demonstrate that our likelihood adaptation method has greatly improved the estimation performance of PFs in a high noise environment. In addition, the adaptive PFs are highly adaptable to the environment without imposing computational complexity.

Keywords: particle filter; nonparametric estimation; indoor positioning; wireless sensor networks; Kullback–Leibler divergence; dynamic Gaussian model

1. Introduction

With the advancement of wireless sensor network (WSNs) and RFID technologies, the demands for indoor location information have promoted the extensive study of indoor localization methods over the past few years [1]. There are increasing requirements of location-based services (LBS) in practical applications, such as indoor navigation systems, location-based social networks, indoor robot and healthcare [2]. The reliability of the location accuracy is a major concern when providing the location information. Additionally, the target position can be obtained via range-based measurement, such as time-of-arrival (TOA), time-difference-of-arrival (TDOA) and received signal strength (RSS), by using WSNs or RFID. In a dynamic high noise environment, the multi-path and non-line-of-sight (NLOS) effects cause high interference during the signal transmission, and the range measurement contains high error. In this case, the PF is a promising estimation method in the nonlinear non-Gaussian environment due to its high accuracy and is widely applied in the indoor localization systems.

A particle filter (PF), which is a typical Bayesian estimation method, has drawn much attention in the fields of state estimation and signal processing for many years [3]. Many applications employ the PF to process the collected data, such as target tracking [4,5] and localization systems [6]. Based on Bayes' theorem, PFs generate random samples, which are called particles with associated weights to represent the posterior PDF of the state. The associate weights are proportional to the prior PDF and the likelihoods and updated recursively.

However, the estimation accuracy of the PF can also be degraded due to the range measurement uncertainty. A traditional way is to generate high quality particles, since the PF is introduced to the localization problem [7]. The bootstrap particle filter (BPF) is applied to generate and re-sample the particles during the recursive position estimation [6]. Farahmand *et al.* proposed a set-membership constraint particle filter (CPF) to select particles within the constrains [8,9]. Kwok *et al.* introduce an adaptable sampling method based on the Kullback–Leibler divergence (KLD) [10]. The Gaussian particle filter (GPF) generates the particles according to the Gaussian distributions and also approximates the estimated distribution as the Gaussian distributions [11]. The accuracy of the GPF is widely evaluated in RFID and WSN-based indoor tracking [12,13]. Ristic *et al.* proposed probability hypothesis density (PHD) and cardinalised PHD (CPHD) filters for target tracking [14]. The other solution is to use an efficient wireless model to overcome the NLOS and multipath effects. The NLOS channel is believed to introduce additional bias error rather than the LOS ranging error. Thus, NLOS path identification and bias reduction methods are widely applied in the PF designs. Jung *et al.* identify the NLOS path for the TOA ranging and use PFs to estimate the target according to the biased error model [15]. Youssef *et al.* provide the biased model of the RSS NLOS ranging error [16]. In addition, the map information can also help to indicate the NLOS path [17]. The problem is that the NLOS error cannot be simply modeled as an LOS error model plus a bias. A typical error model is required for the NLOS channel. The Markov state-space model for NLOS error is usually introduced into the PF-based indoor localization. Nicoli *et al.* developed a recursive Bayesian estimator that combines the Markov transition model and the NLOS propagation in the indoor environment [18]. Wang *et al.* introduce the NLOS error model to the Markov model of the PF based on 802.15.4a [19]. Papa *et al.* proposed adaptive Bayesian methods to track the target using a sensor network [20]. In addition, more information is employed to reduce the measurement error. The hybrid information of TOA and RSS is exploited for likelihood calculation in [21]. Further, the combination of TOA/RSS with NLOS and the multipath mitigation method is introduced in [22]. Bargshady *et al.* also use the PF to process the hybrid WiFi and ultra wideband (UWB) measurements to reduce the NLOS effect [23].

The PF achieves high estimation accuracy by using the prior probability and likelihood based on the Bayesian method. The likelihood relies on the measurement error distribution. However, there is no exact distribution to represent the measurement noise in the complicated indoor environment. Even if the distribution fitting or wireless propagation modeling methods can derive the model parameters, these parameters should be reset when the environment is changed, such as an infrastructure change or the target moves to another area. Thus, the indoor measurement noise shows nonparametric features [24], which requires that the PF is robust to the unknown error models. In this paper, we propose a novel method to design the PF in order to achieve more accurate estimation in a complicated environment. We firstly analyze the measurement uncertainty for the likelihood function (LF) in the PF. The LF, which relies on the measurement and noise distribution, has a large proportion in determining the particle weights. Since the error model is unknown to the LF calculation, the posterior PDF of the state estimation is unreliable. We divide the measurement error model into two parts: the pre-assumed distribution and the unknown model, and use a dynamic Gaussian model (DGM) to comprehensively describe both parts [25]. The DGM assumes that the real-time measurement error follows the pre-assumed distribution, which is dynamically deviated by the instantaneous unknown error. Then, the likelihood function (LF) in the PF, which relies on the measurement function and noise distribution, is calculated based on the DGM.

Our goal is to reduce the instantaneous unknown noise in LF according to our analysis, and the major contributions of our work are four-fold: (1) We propose a likelihood adaptation method, which is based on the DGM. The method combines the prior information and a tuning parameter to reduce the instantaneous unknown error: the prior information is the predicted measurement based on the predicted state; and the belief factor θ is the tuning parameter, which adapts the likelihood function to a more accurate one. By tuning θ , the impact of unknown error for likelihood calculation is reduced; (2) In order to obtain the optimal performance, we use the Kullback–Leibler divergence (KLD), which is an efficient metric to compare two distributions, to derive the optimal θ . The optimal θ can achieve the minimum KLD and attain the lowest estimation error of the PF; (3) We formulate the optimal performance of the adaptive method by using the Cramér–Rao lower bound (CRLB) analysis. The analytical results indicate that our method outperforms the conventional PF for the nonparametric measurement error models; (4) Three versions of PFs are improved based on our adaptation method, which are the BPF, the GPF and the CPF. The improved PFs are evaluated in the simulations and the real-world WSN indoor localization experiments. The results demonstrate that the proposed algorithms effectively reduce the estimation error and have robust performance in a high noisy wireless environment.

The rest of the paper is organized as follows: Section 2 analyzes the measurement uncertainty effect of the likelihood function in the PF algorithms. Section 3 describes a dynamic Gaussian modeling method for the uncertainty and outlier measurement errors. Section 4 proposes the likelihood adaptation method, and the adaptive PFs are designed in Section 5. The theoretical analysis based on CRLB is described in Section 6. Section 7 presents the simulation and real experiment results of adaptive PFs in the indoor localization systems. Finally, Section 8 concludes this paper.

2. Particle Filter Estimation Model

2.1. State-Space Model

In general, the state-space model is applied to describe the nonlinear estimation problem based on the hidden Markov process. The estimated state vector x_t is evolved from the current observation z_t and the previous state x_{t-1} , which are defined as the prediction equation and the update equation. The prediction equation is:

$$x_t = \mathbf{f}_t(x_{t-1}) + q_t \quad (1)$$

where $\mathbf{f}_t(\cdot)$ is the prediction function, x_{t-1} is the previous state and q_t is the prediction noise, which implies the prediction error of x_t . Here, we assume that q_t is additive to the prediction.

At time t , the system obtains a noisy measurement vector z_t . The relationship between x_t and z_t follows:

$$z_t = \mathbf{h}_t(x_t) + v_t \quad (2)$$

where $\mathbf{h}_t(\cdot)$ is the measurement function and v_t is the measurement noise at time t . In this paper, we employ the TOA measurement for range information. The TOA is a simple and straightforward range measurement method for indoor localization. The distance is proportional to the transmission delay, and the noise comes from the additive delay, which is introduced by the multi-path, NLOS and unsynchronized distributed clock within the sensors. The TOA is generally formulated as:

$$\tau = \frac{1}{c} \|\mathbf{x}_t - \mathbf{a}\| \quad (3)$$

where τ is the transmission delay, c is the microwave speed, $\mathbf{a} = [s_x, s_y]^T$ is the position vector of the anchor and $\|\mathbf{x}_t - \mathbf{a}\|$ indicates the distance between the target and the anchor. Considering the noise, the range measurement for each anchor j can be formulated as:

$$z_t^j = h_t^j(x_t) + v_t^j = \|\mathbf{x}_t - \mathbf{a}^j\| + v_t^j \quad (4)$$

where $\mathbf{x}_t = [p_x^t \ p_y^t]$ is the target position and $\mathbf{a}^j = [s_x^j \ s_y^j]^T$ is the position of anchor j . For a 2D positioning system, the distance measurement function can be expressed as:

$$h_t^j(\mathbf{x}_t) = \|\mathbf{x}_t - \mathbf{a}^j\| = \sqrt{(p_x^t - s_x^j)^2 + (p_y^t - s_y^j)^2} \quad (5)$$

then, $\mathbf{z}_t = [z_t^1 \ \dots \ z_t^j \ \dots \ z_t^N]$, where N indicates the number of anchors. Accordingly, $p(\mathbf{z}_t|\mathbf{x}_t)$ is defined as the measurement likelihood for \mathbf{z}_t , which is essential to the whole estimation.

2.2. Particle Filter

PF consists of three parts: the first part is to generate a set of state samples, which are called particles; the second part is to calculate the associated weight for each particle; the final part is to re-sample the particles and to eliminate the unimportant particles. The estimated state is obtained by calculating the weighted average values of the particles. PF first generates a particle set $\{\mathbf{x}_t^i, w_t^i\}_{i=1}^{N_s}$ according to the importance density function $I_s(\mathbf{x}_t^i)$, where $\{\mathbf{x}_t^i\}_{i=1}^{N_s}$ is the sample set with associate weights $\{w_t^i\}_{i=1}^{N_s}$; i and N_s denote the particle number and the total number, respectively. The purpose of using $I_s(\mathbf{x}_t^i)$ is to generate high quality particles and then to attain the accurate estimation. Since the particles represent the possible state in the state space, it is necessary to sample a finite particles with high probability that can represent the state due to the limited computational overhead. Thus, the high quality particles are the samples with a high probability generated and calculated from $I_s(\mathbf{x}_t^i)$. Based on $I_s(\mathbf{x}_t^i)$, most particles are drawn only in the interested area, and a few particles are sampled on the other places. Thus, the posterior PDF $p(\mathbf{x}_t|\mathbf{z}_t)$ of state \mathbf{x}_t can be approximated by using the delta function:

$$p(\mathbf{x}_t|\mathbf{z}_t) \approx \sum_{i=1}^{N_s} w_t^i \delta(\mathbf{x}_t - \mathbf{x}_t^i) \quad (6)$$

For the Bayesian estimation in a non-Gaussian environment, it hardly provides the analytical formulation. Thus, the approximation is used with the set of particles, which is the main advantage of the PF [26]. Additionally, the weight of each particle can be calculated based on the Bayes rule [27]:

$$w_t^i \propto w_{t-1}^i \frac{p(\mathbf{z}_t|\mathbf{x}_t^i)p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i)}{I_s(\mathbf{x}_t^i|\mathbf{x}_{0:t-1}^i, \mathbf{z}_t)} \quad (7)$$

where $p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i)$ is the transition probability for particle \mathbf{x}_t^i and $p(\mathbf{z}_t|\mathbf{x}_t^i)$ is the measurement likelihood of \mathbf{z}_t for \mathbf{x}_t^i . The importance density function $I_s(\mathbf{x}_t^i) = I_s(\mathbf{x}_t^i|\mathbf{x}_{0:t-1}^i, \mathbf{z}_t)$ draws new samples $\{\mathbf{x}_t^i\}_{i=1}^{N_s}$, which evolves from the previous samples $\{\mathbf{x}_{t-1}^i\}_{i=1}^{N_s}$, and $0 : t - 1$ denotes the time sequence from the initial state to the previous state.

The PF estimation relies on the high quality particles and associated weight calculations. The weight is obtained via Bayes' rule, which considers the prior probability and the measurement likelihood. The likelihood relates to the measurement and the noise distributions. A proper error distribution can lead to an accurate estimation not only for the PF, but also for other probability-based algorithms, e.g., maximum likelihood (ML) and maximum *a posterior* (MAP) algorithms.

2.3. The Measurement Uncertainty

In PFs, the measurement likelihood for each particle is calculated as [26]:

$$p(\mathbf{z}_t|\mathbf{x}_t^i) = p(\mathbf{h}_t(\mathbf{x}_t) - \mathbf{z}_t^i + \mathbf{v}_t) \quad (8)$$

where $\mathbf{z}_t^i = \mathbf{h}_t(\mathbf{x}_t^i)$ and $p(\cdot)$ is the PDF of measurement noise \mathbf{v}_t with variables of \mathbf{z}_t^i . The additive measurement noise is assumed to be zero mean and follows a Gaussian distribution. However, the statistical results indicate that the noise is not zero mean due to heavy outlier values [28]. Thus, other PDFs are applied for measurement noise modeling, e.g., exponential, Rayleigh, Weibull or Gamma

distribution [29]. If such a PDF is known to the system, the PF can obtain an accurate estimation, which approaches the optimal estimation with an increasing number of particles, as CRLB indicates. However, the PDF is not always known or precise for real applications. In real applications, the measurement error is modeled by statistical methods based on the sample dataset. On the one hand, the parameters of the error model can be changed if the sample dataset is changed. On the other hand, the outlier values can deviate from the error distribution, which makes the error model inaccurate for the PF estimation.

For instance, we construct an experiment to collect all of the ranging measurement noise throughout the whole building. During such an experiment, we employ a robot moving automatically through every possible position of the building, including the hallway, the office room and classroom in a building at Freie University Berlin. Then, all of the ranging errors are collected, and the histogram is constructed. The error data can be attained freely via the website of [30]. The large positive noise, which may contribute a heavy tail for a histogram, is shown in Figure 1. In Figure 1, the histogram can be divided into two parts: the left part is similar to a Gaussian distribution, for which the mean and variance can be calculated; the right part is a kind of heavy tail, which represents the outlier values. To describe the histogram, researchers use an arbitrary assumed error model or a combined model to approach the real application. However, this makes the estimation algorithms sensitive to the environment. If the environment is changed, the model should be adapted accordingly; otherwise, it can lead to the wrong estimation.

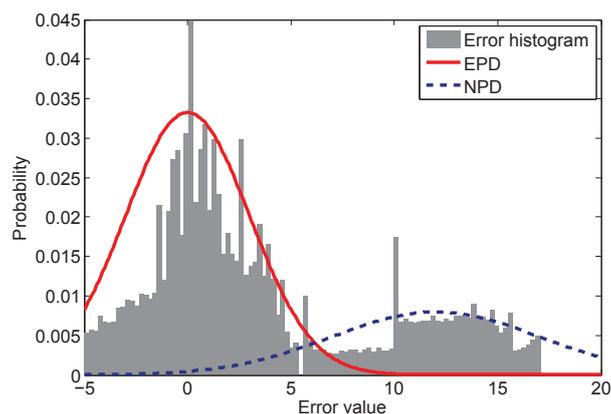


Figure 1. Measurement error histograms and the dynamic Gaussian model for measurement uncertainty.

3. Dynamic Gaussian Model

To address the measurement uncertainty problem and to make the PF robust to environmental change, we firstly introduce a dynamic Gaussian model (DGM) to the likelihood calculation. The main idea of the DGM is to classify the measurement error distribution into two parts: the first part is the expected parametric distribution (EPD), which is a pre-assumed distribution and known to the system. The EPD is obtained based on the knowledge or experiences of the system design, and the parameters can be attained via the system model or some pre-assumptions. The most popular EPD is the normal distribution. The second part is the non-parametric distribution (NPD), which is unknown to the system and for which it is hard to get the parameters. The conventional PFs only use EPD as the pre-assumed distribution model for estimation. The NPD is the compensation for the EPD when the PFs are in the dynamic environment without the knowledge of the noise model. The histogram of the EPD and the NPD is depicted in Figure 1. In Figure 1, the error value between -5 and 5 can mainly be modeled as the normal distribution, which is the EPD and denoted by the solid curve. For the outlier values, another Gaussian distribution NPD attempts to cover such values, which is depicted by the dashed curve.

The DGM is the drifted EPD, which is deviated by the instantaneous NPD value. Here, we use $v_t = w_t + n_t$, where w_t is the EPD error and n_t is the NPD error. Then, v_t follows the EPD, but deviated by n_t . The DGM is different from the Gaussian mixture model. The Gaussian mixture model uses different Gaussian distributions to fit the non-Gaussian distributions. The results are still a static distribution model. However, DGM is dynamic and does not attempt to fit the error histogram, because the NPD part is not an exact model to fit the outlier values. It is rather a fuzzy function to cover such values. In addition, the Gaussian mixture model is a parametric model for the entirety of the error samples, and the DGM only contains a single instantaneous value of the NPD instead of the entirety of the data samples. In general, if $n_t = 0$, v_t follows the EPD. However, for a typical measurement, v_t consists of a drift value w_t that deviates from the pre-assumed normal distribution to a certain distance. Then, the EPD should be adapted dynamically according to the instantaneous value of the NPD.

The DGM for a typical NPD error is depicted in Figure 2. In Figure 2, the solid curve represents the EPD, which is a zero mean normal distribution. When the instantaneous value of the NPD is introduced into the measurement noise, the EPD is deviated. Then, we obtain the likelihood calculation:

$$p(z_t|x_t^i) = p(\mathbf{h}_t(x_t) - z_t^i + w_t + n_t) \quad (9)$$

where an unpredictable instantaneous noise n_t is introduced into the likelihood function.

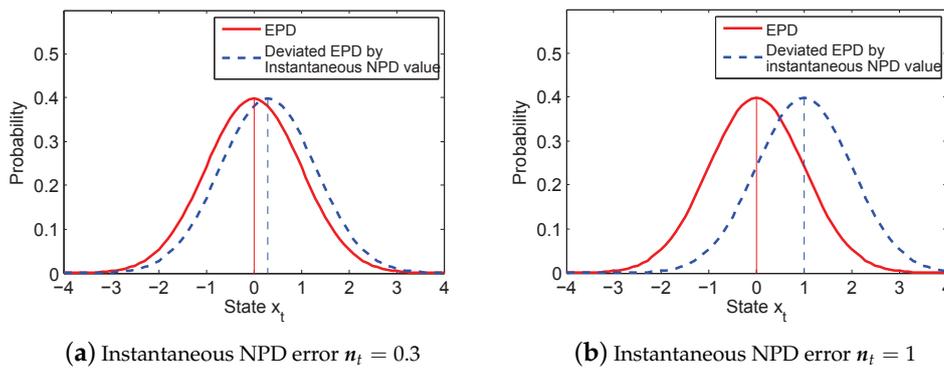


Figure 2. Example: the two likelihood functions. The solid curve represents the expected parametric distribution (EPD), and the dashed curve represents the deviated likelihood function by the instantaneous non-parametric distribution (NPD) value.

Equation (9) is illustrated as the dashed curve in Figure 2, which is a biased non-zero mean Gaussian distribution. It is deviated from the original assumption due to considering the instantaneous value, n_t . If n_t is zero, the likelihood function $p_A(z_t|x_t^i)$ is the exact EPD of measurement noise:

$$p_A(z_t|x_t^i) = p(\mathbf{h}_t(x_t) - z_t^i + w_t) \quad (10)$$

where p_A represents the actual assumed probability; then, we would have optimal filtering with the increasing number of particles. However, in most real cases, n_t is not zero, and w_t is deviated by n_t in Equation (10), which leads to inaccurate estimation. When n_t becomes larger, the gap between the two curves is increasing, as shown in Figure 2b, which degrades the estimation accuracy significantly. Therefore, our goal is to develop an adaptation method to mitigate n_t and approach the likelihood calculation of the exact assumed distribution.

4. Likelihood Adaptation

We attempt to reduce the impact of the NPD and calculate the LF $p(z_t|x_t^i)$ based on the DGM in order to improve the estimation accuracy. Our adaptation method consists of two steps: the first step is to obtain a predicted measurement \hat{z}_t according to the previous state; the second step is to adapt the

LF based on \hat{z}_t and a belief factor θ , which is a tuning parameter for adaptation. The structure of the adaptive PF, which integrates with the predicted measurement and θ , is illustrated in Figure 3. In the original PF, the LF is determined by the EPD, whereas in our algorithm, the likelihood calculation is based on the DGM. Belief factor θ is used to adapt the value between \hat{z}_t and z_t .

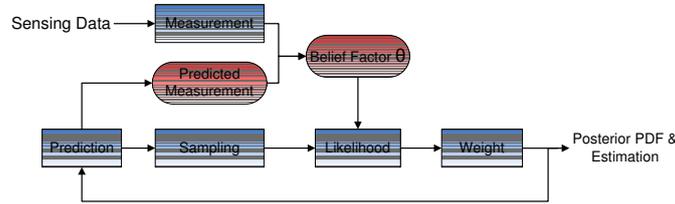


Figure 3. The architecture of the particle filter integrated with the adaptive likelihood method. The common particle filter does not contain the prior measurement and the belief factor.

4.1. Predicted Measurement

The predicted measurement is derived based on the prediction state and is the reference for the real measurement. The calculation steps are as follows: \hat{x}_t denotes the prediction value of x_t :

$$\hat{x}_t = \mathbf{f}_t(x_{t-1}) \tag{11}$$

where x_{t-1} is the estimation at previous time $t - 1$. When considering the processing noise q_t , we denote \hat{x}_t as:

$$\hat{x}_t = x_t + q_t \tag{12}$$

where q_t is assumed to be the additive noise and follows normal distribution $q_t \sim \mathcal{N}(0, Q_t)$; Q_t is the covariance at time t . Then, we obtain a predicted measurement for: sensors:

$$\hat{z}_t = \mathbf{h}_t(\hat{x}_t) = \mathbf{h}_t(x_t + q_t) \tag{13}$$

in which \hat{z}_t indicates the prediction of measurement derived from \hat{x}_t . The predicted measurement \hat{z}_t is not the actual measurement, but is used as prior information for measurement likelihood adaptation.

4.2. Belief Factor and Measurement Adaptation

Belief factor θ is the tuning parameter for \hat{z}_t , and it is used to adapt the measurement z_t to reduce n_t . Note that our goal is to achieve Equation (10). Since Equation (10) can not be achieved directly, we use \hat{z}_t and z_t to approach Equation (10) with θ . Then, the adaptive likelihood function $p_{AL}(z_t|x_t^i)$ is constructed as:

$$p_{AL}(z_t|x_t^i) = p(\theta \odot \hat{z}_t + (\mathbf{1} - \theta) \odot z_t - z_t^i) \tag{14}$$

where $p_{AL}()$ indicates the adaptive likelihood, z_t is the measurement vector, \hat{z}_t is the predicted measurement and $z_t^i = \mathbf{h}_t(x_t^i)$. The notation \odot in Equation (14) is the Hadamard operator, which is used to expressed as $\theta \odot \hat{z}_t = [\theta_t^1 \hat{z}_t^1 \dots \theta_t^j \hat{z}_t^j \dots \theta_t^N \hat{z}_t^N]^T$, and $\mathbf{1}$ is the vector of ones whose size corresponds to that of θ . Note that $\theta \prec \mathbf{1}$, which should be interpreted as the elements in $\mathbf{1} - \theta$, are non-negative. The belief factor θ indicates how much trust $p_{AL}(z_t|x_t^i)$ assigns to the \hat{z}_t . If $\theta = \mathbf{0}$, $p_{AL}(z_t|x_t^i) = p(z_t - z_t^i) = p(z_t|x_t^i)$, which equals the measurement likelihood; whereas when $\theta = \mathbf{1}$, $p_{AL}(z_t|x_t^i) = p(\hat{z}_t - z_t^i)$, which only trusts the predicted measurement.

Here, two problems remains: (1) with introducing $\theta < \mathbf{1}$, the measurement error in z_t is suppressed, but the error of \hat{z}_t is involved, which raises a question: in what condition can our method reduce n_t ? (2) If our method can reduce this effect, a proper θ is required. Therefore, is there an optimal θ that can achieve the best performance?

4.3. Optimal Belief Factor and Likelihood Estimation

Since we intend to compare our adapted DGM to the EPD, the effective evaluation method is KLD [31]. KLD, which also is denoted as relative entropy, quantifies the difference between two distributions. If $p_1(x)$ and $p_2(x)$ indicate two different distributions, KLD is formulated as:

$$D_{KL}(p_1||p_2) = \int p_1(x) \log \frac{p_1(x)}{p_2(x)} dx = E_{p_1}[\log \frac{p_1(x)}{p_2(x)}] \quad (15)$$

where $D_{KL}(p_1||p_2)$ denotes the KLD. The KLD is a non-negative distance between two different distributions, which is $D_{KL}(p_1||p_2) \geq 0$. It can be shown that $D_{KL}(p_1||p_2) = 0 \Leftrightarrow p_1(x) = p_2(x)$. Additionally, small $D_{KL}(p_1||p_2)$ indicates that $p_1(x)$ is similar to $p_2(x)$. In addition, KLD is a convex function [32].

We employ the KLD as a metric to find optimal θ , which minimizes the distance between $p_A(z_t|x_t)$ and $p_{AL}(z_t|x_t)$. Here, we use $p_A(z_t|x_t)$ as the objective distribution and employ $p_{AL}(z_t|x_t)$ as the tuning distribution with parameter θ . Then, the KLD function is constructed as:

$$\begin{aligned} D_{KL}(p_A||p_{AL}) &= \int p_A(z_t|x_t) \log \frac{p_A(z_t|x_t)}{p_{AL}(z_t|x_t)} dz_t^i \\ &= \int \pi_v(\mathbf{h}_t(x_t) - z_t^i) \log \frac{\pi_v(\mathbf{h}_t(x_t) - z_t^i)}{\pi_v(\theta \odot \hat{\mathbf{z}}_t + (\mathbf{1} - \theta) \odot z_t^i)} dz_t^i \end{aligned} \quad (16)$$

Then, optimal θ is attained with minimum $D_{KL}(p_A||p_{AL})$:

$$\theta = \arg \min D_{KL}(p_A||p_{AL}) \quad (17)$$

If $p_A()$ and $p_{AL}()$ are based on the same Gaussian distribution function, then Equation (16) is expressed as [33]:

$$D_{KL}(p_A||p_{AL}) = \frac{\|\mathbf{h}_t(x_t) - [\theta \odot \hat{\mathbf{z}}_t + (\mathbf{1} - \theta) \odot z_t]\|^2}{2\mathbf{R}_t} \quad (18)$$

where \mathbf{R}_t is the covariance of v_t . Since \mathbf{R}_t is independent of θ , the objective function is simplified as:

$$\theta = \arg \min \|\mathbf{h}_t(x_t) - [\theta \odot \hat{\mathbf{z}}_t + (\mathbf{1} - \theta) \odot z_t]\|^2 \quad (19)$$

which turns out to be a least-squares approximation problem [34]. Since $\hat{\mathbf{z}}_t$ is the nonlinear functions of the prediction noise q_t according to Equation (13), it is difficult to obtain an analytical optimal result. Besides, the computational complexity is increased accordingly to obtain an optimal solution. Thus, linearizing the object function is preferred. We use first order Taylor series expansion at x_t to linearize Equation (13):

$$\hat{\mathbf{z}}_t \approx \mathbf{h}_t(x_t) + \frac{\partial \mathbf{h}_t(x_t)}{\partial x_t} q_t \quad (20)$$

where $\partial \mathbf{h}_t(x_t)/\partial x_t$ is the partial differential of $\mathbf{h}_t(x_t)$ with respect to x_t . Additionally, substitute Equation (20) and Equations (2) into (19); we obtain:

$$\|\mathbf{h}_t(x_t) - [\theta \odot \hat{\mathbf{z}}_t + (\mathbf{1} - \theta) \odot z_t]\|^2 \approx \|\theta \odot \frac{\partial \mathbf{h}_t(x_t)}{\partial x_t} q_t + (\mathbf{1} - \theta) \odot v_t\|^2 \quad (21)$$

Therefore, the problem is converted into a linear optimization problem, which is solvable analytically by expressing the objective as the convex quadratic function [34]:

$$\mathbf{F}_t(\theta) = \theta \frac{\partial \mathbf{h}_t(x_t)}{\partial x_t} \mathbf{Q}_t \left[\frac{\partial \mathbf{h}_t(x_t)}{\partial x_t} \right]^T \theta^T + [\mathbf{1} - \theta] \mathbf{R}_t [\mathbf{1} - \theta]^T \quad (22)$$

where Q_t is the covariance of q_t and R_t is the covariance of n_t based on the assumed NPD model.

Then, the optimal θ can be obtained if and only if:

$$\frac{\partial F_t(\theta)}{\partial \theta} = 2\theta \frac{\partial h_t(x_t)}{\partial x_t} Q_t \left[\frac{\partial h_t(x_t)}{\partial x_t} \right]^T - 2R_t + 2\theta R_t = 0 \quad (23)$$

Then, the unique θ is derived:

$$\theta = \frac{R_t}{\frac{\partial h_t(x_t)}{\partial x_t} Q_t \left[\frac{\partial h_t(x_t)}{\partial x_t} \right]^T + R_t} \quad (24)$$

Since θ is the belief factor for the predicted measurement, Equation (24) indicates that when NPD error is high with a large R_t , \hat{z}_t offers more contribution than the noisy measurement. In other words, when the prediction covariance Q_t is larger than R_t , our method should assign more belief to z_t . In this case, \hat{z}_t is useless and can introduce more estimation error. Thus, when the measurement error is quite small and the prediction error exceeds the measurement error, using our method will introduce extra estimation error. Then, the conventional methods can outperform our algorithms. This means that our method is useful when the measurement uncertainty is higher than the predicted state uncertainty, and the optimal θ exists. Fortunately, in the indoor wireless tracking system, the measurement uncertainty and outlier values are always high. In addition, if the measurement noise is low, the prediction error is also small, due to the performance of the Bayesian filtering estimation. Therefore, our method can effectively improve the estimation accuracy in the noisy environment.

5. Adaptive Particle Filter

According to the architecture in Figure 3, we integrate our adaptation method with three main PFs, which are BPF, GPF and CPF, and design new PF versions: adaptive BPF (A-BPF), adaptive GPF (A-GPF) and adaptive CPF (A-CPF).

5.1. Adaptive Bootstrap Particle Filter

In A-BPF, the prediction state \hat{x}_t is obtained through:

$$\hat{x}_t = \mathbf{f}_t(x_{t-1}) \quad (25)$$

where the previous state x_{t-1} is calculated based on: $x_{t-1} = \sum_{i=1}^{N_s} w_{t-1}^i x_{t-1}^i$. Then, we obtain the predicted measurement \hat{z}_t according to Equation (13).

When the measurement z_t is available, the adapted measurement likelihood $p(z_t|x_t^i)$ for each particle is calculated as Equation (14). Additionally, then, the particle weight is calculated and normalized as $w_t^i \propto w_{t-1}^i p_{AL}(z_t|x_t^i)$, which determines the posterior PDF of the estimated state x_t . Finally, x_t is attained: $x_t = \sum_{i=1}^{N_s} w_t^i x_t^i$. The importance sampling and resampling parts are still the same, and the predicted measurement and optimal θ is easy to obtain based on the analytical formulation. Therefore, our likelihood adaptation does not introduce much computational complexity to the original PFs. The algorithm of A-BPF is presented in Algorithm 1. Our adaptation method is implemented in the importance sampling part and the weight adaptation part of PF.

5.2. Adaptive Gaussian Particle Filter

The GPF approximates the estimated PDF by Gaussian distributions using the PF method. The GPF assumes that PDF of the state follows a Gaussian distribution, and it samples particles according to the estimated PDF [11]. Therefore, only the mean and covariance of the estimated PDF are calculated and propagated. Due to the simplicity, the GPF is widely used in distributed PF applications [12].

Particles are drawn from the Gaussian distribution functions, $\{\mathbf{x}_t^i\}_{i=1}^{N_s} \sim \mathcal{N}(\boldsymbol{\mu}_t, \mathbf{Q}_t)$, where $\boldsymbol{\mu}_t$ is the mean value of estimated state and \mathbf{Q}_t is the covariance of PDF. The Gaussian PDF evolves according to the transition model: $\mathbf{x}_t = \mathbf{f}_t(\mathbf{x}_{t-1})$. Thus, the covariance is assumed to propagate to the next time step, which is $\mathbf{Q}_{t|t-1} = \mathbf{Q}_{t-1}$. Then, the initial weight for each particle is calculated as:

$$w_t^i \propto \mathcal{N}(\mathbf{x}_t = \mathbf{x}_t^i, \hat{\mathbf{x}}_t, \mathbf{Q}_{t|t-1}) \quad (26)$$

When the measurements are available, the weight for each particle updates:

$$w_t^i \propto p(z_t | \mathbf{x}_t^i) \mathcal{N}(\mathbf{x}_t = \mathbf{x}_t^i, \hat{\mathbf{x}}_t, \mathbf{Q}_{t|t-1}) \quad (27)$$

where $p(z_t | \mathbf{x}_t^i)$ is the likelihood.

Algorithm 1 Adaptive bootstrap particle filter (A-BPF).

Prediction: $\hat{\mathbf{x}}_t = \mathbf{f}_t(\mathbf{x}_{t-1})$;
 Prior Measurement: $\hat{\mathbf{z}}_t = \mathbf{h}_t(\hat{\mathbf{x}}_t)$;
 //Importance Sampling
 Draw: $\{\mathbf{x}_t^i \sim p(\mathbf{x}_t^i | \mathbf{x}_{t-1}^i)\}_{i=1}^{N_s}$;
 //Measurement Adaptation
for particle $i = 1 : N_s$ **do**
 Likelihood: $p_{AL}(z_t | \mathbf{x}_t^i) = p(\boldsymbol{\theta} \odot \hat{\mathbf{z}}_t + (\mathbf{1} - \boldsymbol{\theta}) \odot z_t - \mathbf{h}_t(\mathbf{x}_t^i))$;
 Weight: $w_t^i = w_{t-1}^i p_{AL}(z_t | \mathbf{x}_t^i)$;
end for
 Normalizing: $w_t^i = \frac{w_t^i}{\sum_{i=1}^{N_s} w_t^i}$;
 Resampling: $\{\mathbf{x}_t^i, w_t^i\}_{i=1}^{N_s}$;
 State Estimation $\bar{\mathbf{x}}_t = \sum_{i=1}^{N_s} w_t^i \mathbf{x}_t^i$;

In the A-GPF, the LF fuses both the prior information and current data based on the DGM. The optimal $\boldsymbol{\theta}$ is derived based on the previous estimated $\mathbf{Q}_{t|t-1}$ and current NPD covariance \mathbf{R}_t . Thus, calculating the optimal $\boldsymbol{\theta}$ is a recursive procedure in A-GPF. The procedure of A-GPF is illustrated in Algorithm 2.

Algorithm 2 Adaptive Gaussian particle filter (A-GPF).

Prediction: $\hat{\mathbf{x}}_t = \mathbf{f}_t(\mathbf{x}_{t-1})$
 Calculate $\hat{\mathbf{z}}_t$ based on Equation (13)
 Randomly Draw: $\{\mathbf{x}_t^i\}_{i=1}^{N_s} \sim \mathcal{N}(\boldsymbol{\mu}_t, \mathbf{Q}_{t|t-1})$
 $\boldsymbol{\theta} = \frac{\mathbf{R}_t}{\frac{\partial \mathbf{h}_t(\mathbf{x}_t)}{\partial \mathbf{x}_t} \mathbf{Q}_{t|t-1} [\frac{\partial \mathbf{h}_t(\mathbf{x}_t)}{\partial \mathbf{x}_t}]^T + \mathbf{R}_t}$
for particle $i = 1 : N_s$ **do**
 Likelihood: $p_{AL}(z_t | \mathbf{x}_t^i) = p(\boldsymbol{\theta} \odot \hat{\mathbf{z}}_t + (\mathbf{1} - \boldsymbol{\theta}) \odot z_t - \mathbf{h}_t(\mathbf{x}_t^i))$;
 Weight: $w_t^i = w_{t-1}^i p_{AL}(z_t | \mathbf{x}_t^i)$;
 Gaussian Distribution Estimation:
 Mean: $\bar{\mathbf{x}}_t = \bar{\boldsymbol{\mu}}_t = \sum_{i=1}^{N_s} w_t^i \mathbf{x}_t^i$
 Covariance: $\bar{\mathbf{Q}}_t = \sum_{i=1}^{N_s} w_t^i (\bar{\boldsymbol{\mu}}_t - \mathbf{x}_t^i)(\bar{\boldsymbol{\mu}}_t - \mathbf{x}_t^i)^T$
end for

The difference between A-BPF and A-GPF is: the prediction error in A-BPF follows an arbitrary assumed distribution, whereas A-GPF uses the Gaussian distribution to indicate such a distribution.

The assumed distribution in A-BPF is obtained based on the statistical analysis, and the estimation of A-BPF can be accurate if the assumed distribution is correct.

5.3. Adaptive Constraint Particle Filter

The CPF randomly samples particles not only based on both the assumed distribution and some constraint conditions, e.g., $\mathbf{x}_t^i \in \mathbf{c}(\mathbf{x}_t, \mathbf{z}_t)$, where $\mathbf{c}(\mathbf{x}_t, \mathbf{z}_t)$ is the constraint functions [9]. The constraint conditions guarantee the particle generated in the target region with a very high probability by allocating a higher previous weight. In this case, both the previous weight and adaptive parameters are maintained during the estimation, where the complexity is still high. Thus, we use the constraint conditions only to sample the particles, and the weights are calculated according to the adaptive likelihoods. Then, the complexity of A-CPF is reduced.

The constraint conditions can be set up according to different applications. We will detail the conditions for the range-based target tracking in the next section. After sampling the particles, the prediction is obtained based on the prediction function. Then, our adaptation method is used, and the weight calculation follows the same procedure as A-BPF, which is illustrated in Algorithm 3. The major difference between A-BPF and A-CPF is the additional constrains for particles, which can improve the performance of BPF with additional information.

Algorithm 3 Adaptive constraint particle filter (A-CPF).

Constraint Sampling: $\{\mathbf{x}_t^i \in \mathbf{c}(\mathbf{x}_t, \mathbf{z}_t)\}_{i=1}^{N_s}$
 Prediction: $\hat{\mathbf{x}}_t = \mathbf{f}_t(\mathbf{x}_{t-1})$
 Calculate $\hat{\mathbf{z}}_t$ based on Equation (13)
for particle $i = 1 : N_s$ **do**
 Likelihood: $p_{AL}(\mathbf{z}_t | \mathbf{x}_t^i) = p(\boldsymbol{\theta} \odot \hat{\mathbf{z}}_t + (\mathbf{1} - \boldsymbol{\theta}) \odot \mathbf{z}_t - \mathbf{h}_t(\mathbf{x}_t^i));$
 Weight: $w_t^i = w_{t-1}^i p_{AL}(\mathbf{z}_t | \mathbf{x}_t^i);$
 Normalizing: $w_t^i = \frac{w_t^i}{\sum_{i=1}^{N_s} w_t^i};$
 Resampling: $\{\mathbf{x}_t^i, w_t^i\}_{i=1}^{N_s};$
end for

5.4. Performance Comparison

Note that the performances of the proposed APFs are different and suitable for different scenarios. In general, all of the APFs can effectively reduce the estimation error in the high noise environment, which will be proven in Section 6. The A-BPF reduces the estimation error by using arbitrary prior information. Such information is set up since the initial sampling scheme. If the sampling scheme and prior information are accurate, the estimation results can be good. However, the system cannot adjust the prior information at every time step. Thus, the A-BPF cannot further reduce the error if the measurement error is small. The A-GPF is quite suitable for adjusting itself during the recursive estimation. At each time step, A-GPF calculates the estimation covariance using the current particles and propagates to the next time step as the prior information. This is quite flexible for a dynamic environment. However, the disadvantage of the A-GPF is that there is some error in the covariance calculation. Thus, if the measurement error is too high, the performance of the A-GPF is not as good as A-BPF. The most robust feature for the dynamic environment is A-CPE, which employs constraint conditions to restrict error. In a high noise environment, the estimation error of the A-CPF can be controlled. However, the A-CPF has an inherent error from the constraint condition. Thus, it is still not as good as the A-BPF if the measurement error is small.

6. CRLB Analysis

In this section, we provide the benefits of the adaptive PFs by deriving the CRLB. The CRLB, which is given by the inverse of the Fisher information matrix (FIM), sets the lower limit for the variance (or covariance matrix) of any unbiased estimators of an unknown parameter (or unknown parameters) [35].

6.1. FIM of the Adapted Likelihood

For the adaptive PFs, the prior information is integrated into the likelihood function. If the measurement noise also follows a Gaussian distribution, we can easily derive the covariance of adapted measurement:

$$\begin{aligned}
 \text{Cov}(\bar{\mathbf{z}}_t) &= \mathbf{E}\{(\mathbf{h}_t(\mathbf{x}_t) - \bar{\mathbf{z}}_t)(\mathbf{h}_t(\mathbf{x}_t) - \bar{\mathbf{z}}_t)^T\} \\
 &= \mathbf{E}\{|\theta \frac{\partial \mathbf{h}_t(\mathbf{x}_t)}{\partial \mathbf{x}_t} \mathbf{q}_t + (\mathbf{1} - \theta) \mathbf{n}_t + \mathbf{w}_t|^2\} \\
 &= \theta \frac{\partial \mathbf{h}_t(\mathbf{x}_t)}{\partial \mathbf{x}_t} \mathbf{Q}_t \left[\frac{\partial \mathbf{h}_t(\mathbf{x}_t)}{\partial \mathbf{x}_t} \right]^T \theta^T + (\mathbf{1} - \theta) \mathbf{R}_t (\mathbf{1} - \theta)^T + \mathbf{W}_t \\
 &= \bar{\mathbf{R}}_t
 \end{aligned} \tag{28}$$

If the measurement noises are independent of each anchor, the covariance matrix can be simplified as $\bar{\mathbf{R}}_t = \text{diag}_{N \times N}(\bar{\mathbf{R}}_j)$, where $\bar{\mathbf{R}}_j$ can be derived independently in each anchor. Then, $\mathbf{I}_D(\mathbf{x}_t)$ is formulated as:

$$\mathbf{I}_D(\mathbf{x}_t) = \begin{pmatrix} \sum_{j=1}^N \bar{\mathbf{R}}_j^{-1} \left(\frac{\partial h_t^j(\mathbf{x}_t)}{\partial p_x^i} \right)^2 & \sum_{j=1}^N \bar{\mathbf{R}}_j^{-1} \frac{\partial h_t^j(\mathbf{x}_t)}{\partial p_x^i} \frac{\partial h_t^j(\mathbf{x}_t)}{\partial p_y^i} \\ \sum_{j=1}^N \bar{\mathbf{R}}_j^{-1} \frac{\partial h_t^j(\mathbf{x}_t)}{\partial p_x^i} \frac{\partial h_t^j(\mathbf{x}_t)}{\partial p_y^i} & \sum_{j=1}^N \bar{\mathbf{R}}_j^{-1} \left(\frac{\partial h_t^j(\mathbf{x}_t)}{\partial p_y^i} \right)^2 \end{pmatrix} \tag{29}$$

for the detailed formulation, please refer to the Appendix.

6.2. FIM of Recursive Estimation

The FIM of recursive Bayesian estimation has been initially formulated by Petr *et al.* [36], which is given as:

$$\mathbf{I}(\mathbf{x}_t) = \mathbf{A} - \mathbf{B}(\mathbf{I}(\mathbf{x}_{t-1}) + \mathbf{C})^{-1} \mathbf{B}^T \tag{30}$$

where

$$\begin{aligned}
 \mathbf{A} &= \mathbf{I}_P(\mathbf{x}_t) + \mathbf{I}_D(\mathbf{x}_t) = \frac{\partial \mathbf{f}_t(\mathbf{x}_t)}{\partial \mathbf{x}_t} \mathbf{Q}_t^{-1} \frac{\partial \mathbf{f}_t(\mathbf{x}_t)}{\partial \mathbf{x}_t}^T + \mathbf{I}_D(\mathbf{x}_t) \\
 \mathbf{B} &= -\frac{\partial \mathbf{f}_t(\mathbf{x}_t)}{\partial \mathbf{x}_t} \mathbf{Q}_t^{-1} \\
 \mathbf{C} &= \frac{\partial \mathbf{f}_t(\mathbf{x}_t)}{\partial \mathbf{x}_t} \mathbf{Q}_t^{-1} \frac{\partial \mathbf{f}_t(\mathbf{x}_t)}{\partial \mathbf{x}_t}^T
 \end{aligned} \tag{31}$$

where $\mathbf{I}(\mathbf{x}_t)$ is a part of $\mathbf{I}([\mathbf{x}_t \ \mathbf{x}_{t-1}]^T)$; \mathbf{A} , \mathbf{B} and \mathbf{C} are parts of $\mathbf{I}([\mathbf{x}_t \ \mathbf{x}_{t-1}]^T)$, which represent the previous and current information. For the detailed description, please refer to the Appendix.

6.3. Analytical Results

We set up several simulations to evaluate the performance improvement of the adaptive PF. The playing field is $100 \times 100 \text{ m}^2$. We randomly deploy 100 sensors. The target chooses a random path walking through the playing field. The range data are obtained through the TOA technique, whose range information is formulated according to Equation (4). The EPD for each sensor follows zero mean Gaussian distribution $w_t^j \sim \sigma_w^2$, where $\sigma_w = 1 \text{ m}$. Additionally, set the NPD of each sensor as $n_t^j \sim \mathcal{N}(1, \sigma_n^2)$, where $\sigma_n = 3 \text{ m}$. Root mean square error (RMSE) is mainly compared in the simulation as the estimation accuracy metric. For the CRLB analysis, the RMSE indicates the root value

of the unbiased covariance. To compare with our proposed method, we use the common recursive CRLB form in [36], where the measurement noise vector is the combination of the EPD and NPD $v_t \sim \mathcal{N}(\mathbf{1}, \mathbf{R}_t + \mathbf{W}_t)$. It should be noted that the the measurement noise in the simulation is not simply the sum of two distribution. EPD is the general error distribution, which is added during the whole simulation. However, NPD is applied to represent outlier values of the simulation. Thus, during each time step, we randomly add an NPD error or not. In this case, the NPD error is a few outlier values that can influence the measurement, and DGM must be applied for the PF estimation. The sequential estimation results are illustrated in Figure 4.

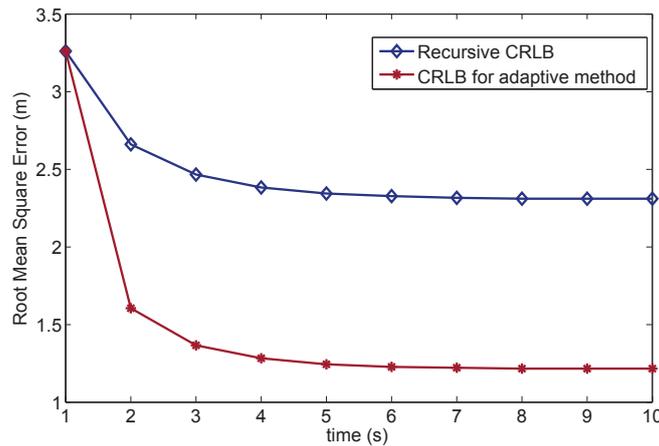


Figure 4. Cramér–Rao lower bound for target tracking.

We also adapt the measurement noise environment to examine the performance change for each process. The NPD covariance for each anchor changes from $R_j = (0.5 m)^2$ to $R_j = (5.5 m)^2$. The root mean squared error (RMSE) results of the two CRLBs are depicted in Figure 5. When the NPD covariance is small, the adaptive method cannot improve estimation, but increases the estimation error. In this case, it would rather use only the measurement data to obtain the estimation. However, with increasing the NPD covariance, the adaptive method improves the estimation gradually and outperforms the recursive Bayesian method when the error is quite high. Thus, our method is feasible for a harsh environment with high uncertainty.

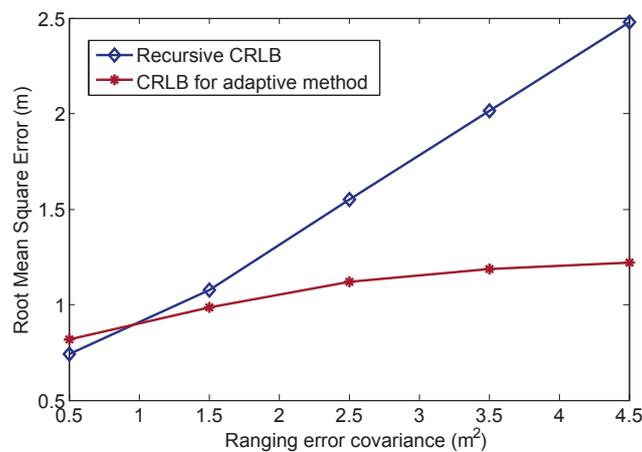


Figure 5. RMSE results of the Cramér–Rao lower bound with a multiple NPD environment.

7. Range-Based Target Tracking Simulation and Experiment

7.1. Simulation Setup

Our scheme is evaluated in the simulation of target tracking using the wireless sensor networks. We randomly deploy 100 sensors in a two-dimensional square $100 \times 100 \text{ m}^2$ region. One target runs through a random path with a constant speed. The target periodically broadcasts the request signals to the sensors, and the sensors respond to the request. Then, the ranging values are calculated according to the time-of-arrival (TOA) of the responses. Each sensor node j is assigned coordinations $[s_x^j, s_y^j]^T$, and the target state at time t is $x_t = [p_x^t, p_y^t, v_x^t, v_y^t]^T$, where $[p_x^t, p_y^t]$ is the position vector, $[v_x^t, v_y^t]$ is the velocity vector and T is the transpose operator. In the simulation, we assume that the measurement noise consists of the EPD and the NPD values, and the measurement noise for each sensor node is identical independent distributed (i.i.d). The EPD noise follows the zero mean Gaussian distribution, where $w_t^j \sim \mathcal{N}(0, \sigma_w^2)$ and $\sigma_w = 1 \text{ m}$. Additionally, the NPD noise also follows a Gaussian distribution, where $n_t^j \sim \mathcal{N}(1, \sigma_n^2)$ and $\sigma_n = 3 \text{ m}$. Then, the DGM can be illustrated as shown in Figure 6.

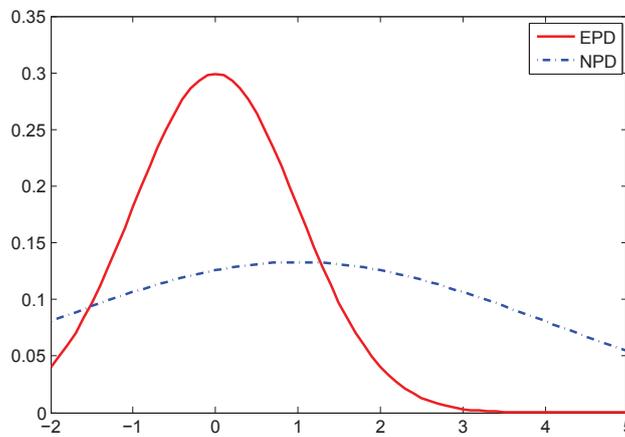


Figure 6. The dynamic Gaussian model (DGM) pattern in the the simulation.

7.2. Particle Filter Modeling

We employ a linear model as the prediction function, and Equation (1) can be expressed as:

$$x_t = \mathbf{F}_t x_{t-1} + q_t \quad (32)$$

where \mathbf{F}_t is the linear transition matrix:

$$\mathbf{F}_t = \begin{pmatrix} 1 & 0 & \Delta T & 0 \\ 0 & 1 & 0 & \Delta T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (33)$$

where ΔT indicates the time interval.

In the particle sampling stage, the samples are initially randomly generated and propagated according to the linear transition model in BPF and A-BPF. In GPF and A-GPF, the particles are sampled based on the updated PDF. In CPF, the particles are constrained to some conditions. For range-based localization systems, the constraint region can be drawn by the bounding box algorithm, which is demonstrated to be robust to measurement noise [37]. Using z_t^j to denote the j -th range measurement for node j with the position $[s_x^j, s_y^j]^T$ in the two-dimensional playing field, then the geometric constraint

region is: $\max\{s_x^j - z_t^j\}_{j=1}^N \leq p_x^t \leq \min\{s_x^j + z_t^j\}_{j=1}^N$ and $\max\{s_y^j - z_t^j\}_{j=1}^N \leq p_y^t \leq \min\{s_y^j + z_t^j\}_{j=1}^N$, where N is the number of sensor nodes. Then, the particles are sampled within this region.

7.3. Optimal Belief Factor for A-BPF

First, we compare the estimation accuracy of BPF to our algorithm, A-BPF, to analyze the relationship between θ and the estimation accuracy. We implement A-BPF and vary θ from 0 to 1 to verify whether θ has an optimal value with minimum estimation error. The algorithms are tested in the different measurement noise scenarios, in which the NPD noise variance σ_n^2 is tuned from 0.5^2 to 5^2 . The results of our simulation are averaged by over 1000 Monte-Carlo trials. We randomly deploy sensor nodes in every trial. In this simulation, all algorithms generate 1000 particles at each time step t . Figure 7 illustrates the root mean square error (RMSE) of each algorithm with some different measurement noise scenarios.

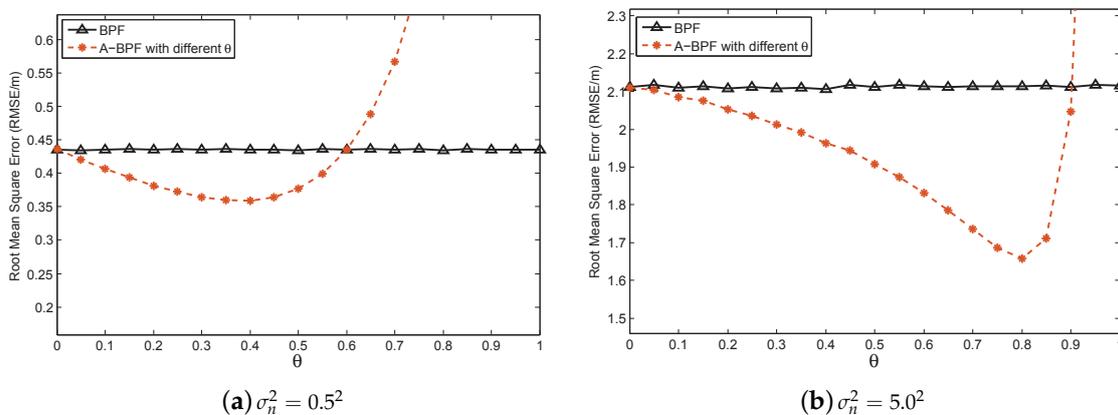


Figure 7. Estimation error comparison for different algorithms with different measurement error variances.

In Figure 7, the solid line represents the RMSE of BPF, and the dashed line indicates the RMSE of A-BPF with different θ , in which θ changes from 0 to 1 with an interval of 0.05. When $\theta = 0$, A-BPF totally believes the noisy measurement and has the same accuracy as BPF. The RMSE of A-BPF is decreasing when θ increases from 0, which indicates that our adaptation method can improve the measurement likelihood and estimation accuracy. As indicated in each figure in Figure 7, A-BPF has a minimum RMSE with optimal θ in different measurement noise scenarios. When θ is larger than the optimal value, the RMSE of A-BPF begins to increase. Additionally, when θ approaches 1, it is over tuned and causes high estimation error.

When the measurement noise is low, e.g., in Figure 7a, A-BPF does not improve much, even with the optimal θ . However, when the measurement noise is quite large, e.g., in Figure 7b, A-BPF can effectively reduce the RMSE with the optimal θ . Additionally, the gap between A-BPF and BPF is much larger in Figure 7b than in Figure 7a. Besides, the optimal θ increases when the NPD noise rises, e.g., in Figure 7a, θ is 0.3, and it becomes 0.8 in Figure 7b.

The comparison of θ obtained in the simulation and derived in A-BPF is depicted in Figure 8. The solid curve is the optimal θ in the simulations, and the dashed curve is the optimal θ based on Equation (24) with different measurement noise. As shown in Figure 8, the optimal θ based on our calculation in Equation (24) is slightly different from the simulation results, because the optimal θ is derived according to the approximation in Equation (20). Since the optimal θ is approaching the simulation results, it is still suitable for implementation. Figure 8 also depicts that the optimal θ increases with the rise of measurement noise. It testifies for Equation (24) that θ rises when the measurement noise becomes larger. In this case, A-BPF assigns more belief for the predicted measurement for adaptation.

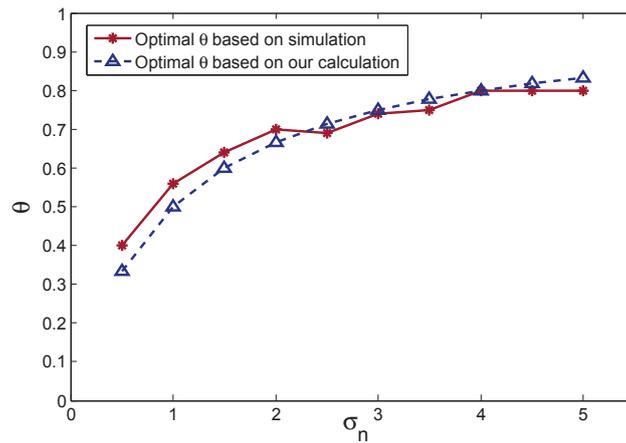


Figure 8. Optimal θ comparison between the simulation results and our calculation based on Equation (24).

7.4. Root Mean Square Error Comparison

Figure 9 illustrates the RMSE comparison for BPF, A-BPF, GPF, A-GPF, CPF and A-CPF, and the CRLB for the adaptive filters is also depicted as the optimal indicator. The measurement noise covariance varies from 0.5^2 to 5.5^2 , and we use the optimal θ based on Equation (24) as the belief factor in the adaptive PFs. When the measurement error is small, the performances of the PFs are similar. Furthermore, we can find that the adaptive PFs have slightly higher error than the original PFs when the noise is too low. This is due to the inherent sampling method, which generate samples with higher statistical covariance than the measurement noise. In this case, our adaptation method relies on such covariance and causes extra error in the final estimation, which is as illustrated in the previous sections. When the error increases, the estimation errors of BPF, GPF and CPF rise accordingly. Especially, CPF has the highest RMSE due to the imprecise likelihood calculation, although it has constraints. This indicates that the measurement noise is the major impact that influences the estimation performance of PFs. The adaptive PFs have better performance. A-BPF and A-GPF have similar estimation accuracy. The RMSE of A-CPF is the lowest and mostly approaching the CRLB.

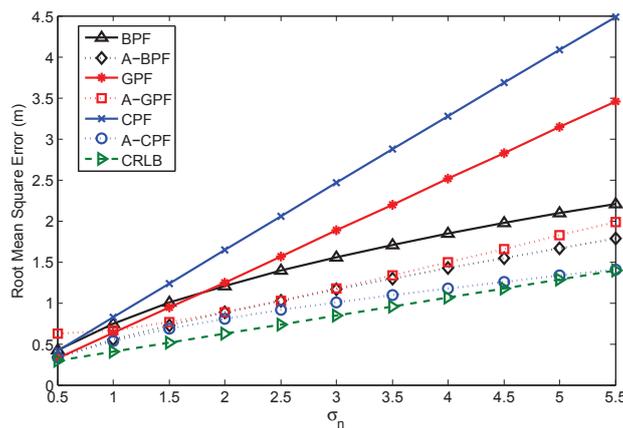


Figure 9. Comparing the performances of different algorithms.

We also compare the estimation performance of PFs with varying the particle numbers. As Crisan *et al.* indicate, the estimation error should converge to zero with the increasing number of particles [38]. However, simulation results show that the estimation accuracy is corrupt with high measurement noise. When the error is small, as illustrated in Figure 10a, the RMSEs of PFs can

converge to a very low value with the increased particle number, except BPF and CPF, which illustrates that the measurement noise can influence the convergence of PFs, although not much. In this case, our adaptive method does not improve much, and A-GPF even has a higher RMSE than GPF. When the measurement noise begins to rise, increasing the particle number of the original PFs cannot improve the estimation accuracy. In Figure 10b, the RMSEs of BPF and CPF begin to rise when the number of particles exceed 100, and GPF does not improve either. This implies that high measurement error leads to a significantly inaccurate likelihood calculation, which degrades the estimation. In addition, CPF outperforming BPF, which is mentioned in [9], only occurs in the low noise case. However, our adaptive method can improve the estimation accuracy and make PFs converge to a low RMSE. In Figure 10, RMSEs of three adaptive PFs decrease with the rising particle number. RMSEs can converge to a very low value even when the measurement error is high. Therefore, our method can reduce the imprecise measurement effect and achieve a better performance.

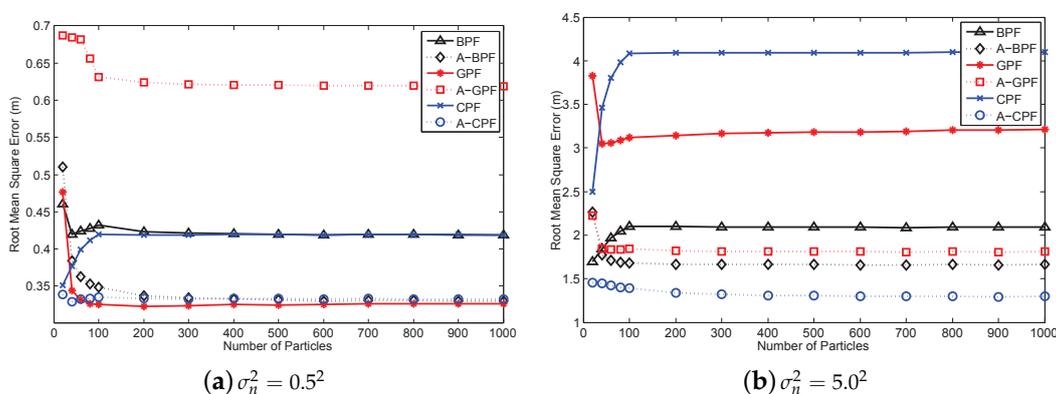


Figure 10. RMSE comparison for different algorithms with different numbers of particles.

7.5. The Real Indoor Experiment Test-Bed

We also employ a reference system for indoor localization test-beds to examine our proposed algorithms. In this system, we deployed 17 wireless sensor nodes, either along the corridor or in the offices of the research building. A robot carrying a sensor node as a target moved along the corridor of the building with constant speed while recording its own positions [30]. In the previous simulations, we assume that the noise follows an independent identical distribution. However, in the real indoor environment, the wireless signals are propagated through complicated channels. The LOS and NLOS range measurements are mixed for a single target. Thus, a specific distribution for each sensor or anchor is necessary to derive accurate estimation. We still use the DGM to model the measurement noise. All sensors are integrated with the nanoPAN 5375 RF module with a 2.4-GHz transceiver and a 1-Mbps data rate for range measurement, LPC 2387 as the micro-controller and a CC1101 900-MHz transceiver as the radio transceiver for communication. The data collected from sensor nodes are also range measurement values, which are based on TOA. There is an inherent error when the 3D localization is projected into the 2D map, due to the height difference between the sensors and target [39]. We adapt the heights of the sensors to be the same as the robot to reduce the impact of the height difference in the 3D world. In this case, we can draw an exact 2D trajectory within the map. Figure 11 depicts the map of our experimental building. The triangles, which are randomly deployed, mark the sensor node positions.

We also implement the six algorithms mentioned above to compare their performance. Each algorithm maintains 200 particles, and several trajectories are examined. In Figure 12, we present one estimated trajectory. We compare the estimation performance by evaluating the RMSE. The RMSE results of the six algorithms are listed in Table 1. As shown in Table 1, the mean average errors (MAE), which are the average performance, for adaptive PF are less than the original PFs. Additionally,

the adaptive PFs have about 1 m less than the original PFs in the RMSE. The max error, which is the worst case for estimation, in the original PFs is more than 6 m and even 8 m in CPF. However, adaptive PFs have about less than 6.91 m, and the max error can even reach 4.0649 m in A-BPF in Table 1, which is much more precise than the original PFs. The overall cumulative distribution functions (CDF) for both trajectories are plotted in Figure 13. Based on the comparison, our adaptive PFs generally outperform the original PFs and show robust performance in the dynamic indoor environment.

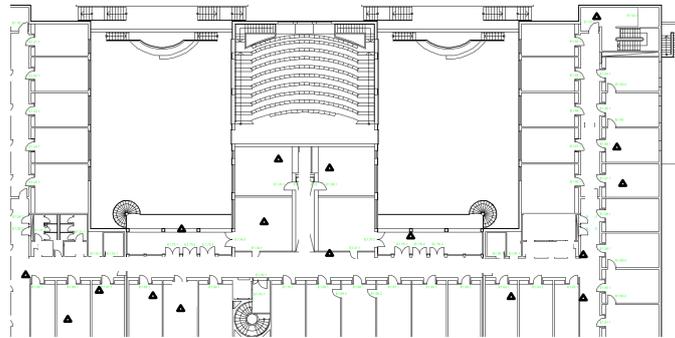


Figure 11. Building layout for indoor localization experiment and the robot trajectory. The triangles mark the positions of sensor nodes, which are placed either in the offices or along the corridor.

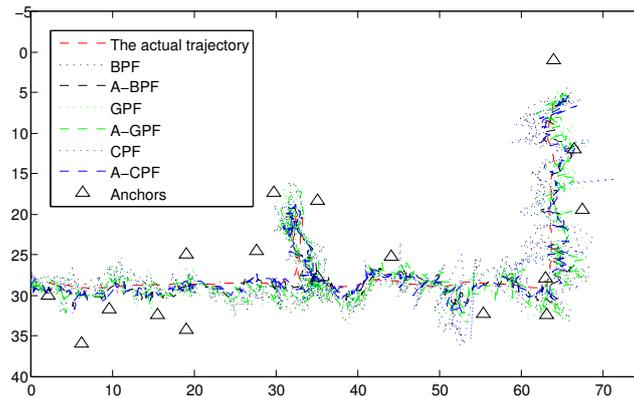


Figure 12. Trajectories for the indoor target tracking.

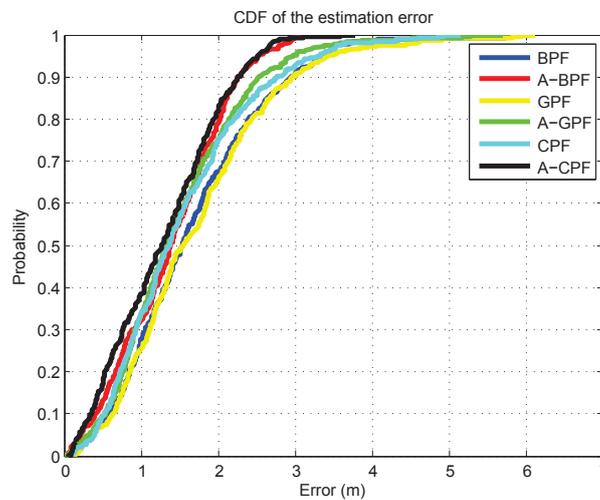


Figure 13. Overall cumulative distribution functions of particle filters (PFs) for both trajectories.

Table 1. Trajectory I: Performance comparison. BPF, bootstrap particle filter; A-BPF, adaptive BPF; GPF, Gaussian particle filter; CPF, constraint particle filter.

Algorithm	MAE (m)	RMSE (m)	Min Error (m)	Max Error (m)
BPF	0.4375	2.3637	0.0649	6.9100
A-BPF	0.2092	1.2907	0.0340	4.0649
GPF	0.4078	2.1231	0.0723	6.2308
A-GPF	0.1612	1.6453	0.0443	5.0614
CPF	0.2455	2.0439	0.0397	8.4929
A-CPF	0.1517	1.3395	0.0506	4.6814

8. Conclusions

In this paper, we use a DGM, which combines the EPD and NPD, to describe the hybrid LOS/NLOS ranging noise in the indoor environment. Based on this model, we propose an adaptation method by introducing the predicted measurement and its belief factor θ . The optimal θ is derived and implemented into our proposed adaptive algorithms, A-BPF, A-GPF and A-CPF. The estimation performance of the adaptive PFs is proven by the CRLB analysis. In the simulation, we observed that some analytical conclusion for PFs are not suitable for the high measurement noise scenarios, and we verify that the adaptive PFs improve the estimation accuracy with different measurement noise environments and that the optimal θ derived in our method approaches the actual value. The real indoor localization experimental results indicate that, comparing to the original PFs, our algorithms can effectively reduce the estimation error, especially in a high measurement noise environment. Besides, A-CPF is more accurate for target tracking than the other filters.

Acknowledgments: This work is partially supported by Major Project of Chinese National Programs for Fundamental Research and Development (973 Program, No. 2015CB352400), National Nature Science Foundation of China (Grand No. 61501443, U1401258, 61202416), Science and Technology Planning Project of Guangdong Province (2015B010129011), Shenzhen Overseas High-Caliber Personnel Innovation Funds (KQCX20140521115045446), Shenzhen-Hong Kong Innovative Technology Cooperation Funding (Grant No. SGLH20131009154139588), Shenzhen Fundamental Research Project (Grant No. JCYJ20150630151329298, JCYJ20150401145529029).

Author Contributions: Xiaofan Li and Yubin Zhao proposed the likelihood adaptation method for particle filters. Sha Zhang derived the optimal belief factor and designed the adaptive particle filters. Xiaofan Li and Yubin Zhao constructed the simulation and experiments. Sha Zhang analyzed the performance, both based on the CRLB and experimental results.

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

Appendix A. CRLB Derivation

The CRLB, which is given by the inverse of the Fisher information matrix (FIM), sets the lower limit for the variance (or covariance matrix) of any unbiased estimators of an unknown parameter (or unknown parameters) [35].

If $p(x_t, z_t)$ denotes the joint PDF of observations z_t and the state x_t , then the score function is defined as the gradient of its log-likelihood:

$$\mathbf{U}(x_t) = \nabla \ln p(x_t, z_t) = \frac{\partial \ln p(x_t, z_t)}{\partial x_t} \quad (\text{A1})$$

The FIM, $\mathbf{I}(x_t)$, is the variance of the score function:

$$\mathbf{I}(x_t) = \mathbb{E} \left\{ \left[\frac{\partial \ln p(x_t, z_t)}{\partial x_t} \right]^2 \right\} \quad (\text{A2})$$

and CRLB is just the inverse of FIM; the estimation covariance cannot be lower than it:

$$\mathbf{Cov}_{\mathbf{x}_t}(\hat{\mathbf{x}}_t) \succeq \{\mathbf{I}(\mathbf{x}_t)\}^{-1} \quad (\text{A3})$$

Note that the CRLB relies on the system model and the noise distribution. In general, using different information can lead to different results. For instance, if the system only applies the measurement model, the CRLB is higher than the recursive CRLB, which applies the previous information iteratively. In our work, we combine the previous information into the measurement model. When the model and related error distribution are changed, a new CRLB is derived accordingly.

Since $p(\mathbf{x}_t, \mathbf{z}_t) = p(\mathbf{z}_t|\mathbf{x}_t)p(\mathbf{x}_t)$ based on the Bayesian theorem, it is easily seen that $\mathbf{I}(\mathbf{x}_t)$ can be decomposed into two parts:

$$\mathbf{I}(\mathbf{x}_t) = \mathbf{I}_D(\mathbf{x}_t) + \mathbf{I}_P(\mathbf{x}_t) \quad (\text{A4})$$

where $\mathbf{I}_D(\mathbf{x}_t)$ represents the information obtained from measurement data and $\mathbf{I}_P(\mathbf{x}_t)$ represents the prior information.

Firstly, we decompose $\mathbf{I}_D(\mathbf{x}_t)$ using the chain rule as:

$$\mathbf{I}_D(\mathbf{x}_t) = \mathbf{H} \cdot \mathbf{I}_H \cdot \mathbf{H}^T \quad (\text{A5})$$

where $\mathbf{H} = [\nabla_{\mathbf{x}_t} \mathbf{h}_t(\mathbf{x}_t)]$ is the Jacobian matrix of \mathbf{x}_t :

$$\mathbf{H} = \begin{pmatrix} \frac{\partial h_1^1(\mathbf{x}_t)}{\partial p_x^t} & \cdots & \frac{\partial h_1^N(\mathbf{x}_t)}{\partial p_x^t} \\ \frac{\partial h_1^1(\mathbf{x}_t)}{\partial p_y^t} & \cdots & \frac{\partial h_1^N(\mathbf{x}_t)}{\partial p_y^t} \end{pmatrix} \quad (\text{A6})$$

where N indicates the dimension of vector \mathbf{z}_t , $\mathbf{x}_t = [p_x^t \ p_y^t]$. Note that we employ the measurement function according to Equation (4), then we have:

$$\begin{cases} \frac{\partial h_1^j(\mathbf{x}_t)}{\partial p_x^t} = \frac{p_x^t - a_x^j}{\sqrt{(p_x^t - a_x^j)^2 + (p_y^t - a_y^j)^2}} \\ \frac{\partial h_1^j(\mathbf{x}_t)}{\partial p_y^t} = \frac{p_y^t - a_y^j}{\sqrt{(p_x^t - a_x^j)^2 + (p_y^t - a_y^j)^2}} \end{cases} \quad (\text{A7})$$

and \mathbf{I}_H is the FIM conditioned on $\mathbf{h}_t(\mathbf{x}_t)$:

$$\mathbf{I}_H = \mathbb{E}\{\nabla_{\mathbf{h}_t} \ln p(\mathbf{z}_t|\mathbf{x}_t) [\nabla_{\mathbf{h}_t} \ln p(\mathbf{z}_t|\mathbf{x}_t)]^T\} \quad (\text{A8})$$

which is the measurement covariance \mathbf{R}_t^{-1} :

$$\mathbf{I}_H = \mathbf{R}_t^{-1} = \begin{pmatrix} R_t^1 & & \\ & \ddots & \\ & & R_t^N \end{pmatrix} \quad (\text{A9})$$

then the $\mathbf{I}_D(\mathbf{x}_t)$ for conventional CRLB is:

$$\mathbf{I}_D(\mathbf{x}_t) = \begin{pmatrix} \sum_{j=1}^N R_j^{-1} \left(\frac{\partial h_1^j(\mathbf{x}_t)}{\partial p_x^t}\right)^2 & \sum_{j=1}^N R_j^{-1} \frac{\partial h_1^j(\mathbf{x}_t)}{\partial p_x^t} \frac{\partial h_1^j(\mathbf{x}_t)}{\partial p_y^t} \\ \sum_{j=1}^N R_j^{-1} \frac{\partial h_1^j(\mathbf{x}_t)}{\partial p_x^t} \frac{\partial h_1^j(\mathbf{x}_t)}{\partial p_y^t} & \sum_{j=1}^N R_j^{-1} \left(\frac{\partial h_1^j(\mathbf{x}_t)}{\partial p_y^t}\right)^2 \end{pmatrix} \quad (\text{A10})$$

then, we derive the FIM of the adaptive likelihood and recursive formulation based on Equation (A10).

Appendix A.1. FIM of the Adapted Likelihood

For the adaptive PFs, the prior information is integrated into the likelihood function. Since the measurement is adapted by the prior information, \mathbf{I}_D is changed by using the adapted measurement \bar{z}_t , which is the combination of predicted measurement and observed data. Then, the error covariance does not follow the original error distribution, but changes accordingly. If the measurement noise also follows a Gaussian distribution, we can easily derive the covariance of adapted measurement:

$$\begin{aligned} \text{Cov}(\bar{z}_t) &= \mathbf{E}\{(\mathbf{h}_t(\mathbf{x}_t) - \bar{z}_t)(\mathbf{h}_t(\mathbf{x}_t) - \bar{z}_t)^T\} \\ &= \mathbf{E}\{|\theta \frac{\partial \mathbf{h}_t(\mathbf{x}_t)}{\partial \mathbf{x}_t} \mathbf{q}_t + (\mathbf{1} - \theta) \mathbf{n}_t + \mathbf{w}_t|^2\} \\ &= \theta \frac{\partial \mathbf{h}_t(\mathbf{x}_t)}{\partial \mathbf{x}_t} \mathbf{Q}_t \left[\frac{\partial \mathbf{h}_t(\mathbf{x}_t)}{\partial \mathbf{x}_t} \right]^T \theta^T + (\mathbf{1} - \theta) \mathbf{R}_t (\mathbf{1} - \theta)^T + \mathbf{W}_t \\ &= \bar{\mathbf{R}}_t \end{aligned} \quad (\text{A11})$$

If the measurement noise is independent of each anchor, the covariance matrix can be simplified as $\bar{\mathbf{R}}_t = \text{diag}_{N \times N}(\bar{R}_j)$, where \bar{R}_j can be derived independently in each anchor. Similar to Equation (A10), $\mathbf{I}_D(\mathbf{x}_t)$ is formulated as:

$$\mathbf{I}_D(\mathbf{x}_t) = \begin{pmatrix} \sum_{j=1}^N \bar{R}_j^{-1} \left(\frac{\partial h_t^j(\mathbf{x}_t)}{\partial p_x^t} \right)^2 & \sum_{j=1}^N \bar{R}_j^{-1} \frac{\partial h_t^j(\mathbf{x}_t)}{\partial p_x^t} \frac{\partial h_t^j(\mathbf{x}_t)}{\partial p_y^t} \\ \sum_{j=1}^N \bar{R}_j^{-1} \frac{\partial h_t^j(\mathbf{x}_t)}{\partial p_x^t} \frac{\partial h_t^j(\mathbf{x}_t)}{\partial p_y^t} & \sum_{j=1}^N \bar{R}_j^{-1} \left(\frac{\partial h_t^j(\mathbf{x}_t)}{\partial p_y^t} \right)^2 \end{pmatrix} \quad (\text{A12})$$

Appendix A.2. FIM of Recursive Estimation

The FIM of recursive Bayesian estimation has been initially formulated by Petr *et al.* [36]. The formulation of the FIM, which indicates the estimation procedure of the conventional PFs, is similar to Equation (A4). The FIM of measurement data can be attained according to Equation (A11). Taking the prior information in $\mathbf{I}_P(\mathbf{x}_t)$ into account, the prior probability is the transition probability $p(\mathbf{x}_t | \mathbf{x}_{t-1})$. However, both \mathbf{x}_t and \mathbf{x}_{t-1} should be estimated during the process. Thus, the information about \mathbf{x}_{t-1} is also not accurate. Then, the estimated parameters for Fisher information is $[\mathbf{x}_t, \mathbf{x}_{t-1}]^T$, and the related FIM including $\mathbf{I}_D(\mathbf{x}_t, \mathbf{x}_{t-1})$ and $\mathbf{I}_P(\mathbf{x}_t, \mathbf{x}_{t-1})$ should be reformulated.

For $\mathbf{I}_D(\mathbf{x}_t)$ computation, it is extended to be a 4×4 matrix, which is:

$$\mathbf{I}_D(\mathbf{x}_t, \mathbf{x}_{t-1}) = \mathbb{E} \left\{ \left[\frac{\partial \ln p(\mathbf{x}_t, \mathbf{z}_t)}{\partial \mathbf{x}_t} \quad \frac{\partial \ln p(\mathbf{x}_t, \mathbf{z}_t)}{\partial \mathbf{x}_{t-1}} \right] \left[\frac{\partial \ln p(\mathbf{x}_t, \mathbf{z}_t)}{\partial \mathbf{x}_t} \quad \frac{\partial \ln p(\mathbf{x}_t, \mathbf{z}_t)}{\partial \mathbf{x}_{t-1}} \right]^T \right\} \quad (\text{A13})$$

Since the current probability is independent of the previous state, $\frac{\partial \ln p(\mathbf{x}_t, \mathbf{z}_t)}{\partial \mathbf{x}_{t-1}} = 0$. Then, we obtain:

$$\mathbf{I}_D(\mathbf{x}_t, \mathbf{x}_{t-1}) = \begin{pmatrix} \mathbf{I}_D(\mathbf{x}_t) & \\ & \mathbf{0} \end{pmatrix} \quad (\text{A14})$$

where $\mathbf{0}$ is a 2×2 zero matrix.

For the recursive process, $\mathbf{I}(\mathbf{x}_t)$ is calculated recursively, which involves $\mathbf{I}(\mathbf{x}_{t-1})$ as the previous FIM. Then, $\mathbf{I}_P(\mathbf{x}_t, \mathbf{x}_{t-1})$ is expressed as:

$$\mathbf{I}_P(\mathbf{x}_t, \mathbf{x}_{t-1}) = \begin{pmatrix} \mathbf{Q}_t^{-1} & \frac{\partial \mathbf{f}_t(\mathbf{x}_t)}{\partial \mathbf{x}_{t-1}} \mathbf{Q}_t^{-1} \\ \frac{\partial \mathbf{f}_t(\mathbf{x}_t)}{\partial \mathbf{x}_{t-1}} \mathbf{Q}_t^{-1 T} & \frac{\partial \mathbf{f}_t(\mathbf{x}_t)}{\partial \mathbf{x}_{t-1}} \mathbf{Q}_t^{-1} \frac{\partial \mathbf{f}_t(\mathbf{x}_t)}{\partial \mathbf{x}_{t-1}}^T + \mathbf{I}(\mathbf{x}_{t-1}) \end{pmatrix} \quad (\text{A15})$$

then, $\mathbf{I}(\mathbf{x}_t, \mathbf{x}_{t-1})$ is given:

$$\mathbf{I}(\mathbf{x}_t, \mathbf{x}_{t-1}) = \mathbf{I}_P(\mathbf{x}_t, \mathbf{x}_{t-1}) + \mathbf{I}_D(\mathbf{x}_t, \mathbf{x}_{t-1}) = \begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{C} + \mathbf{I}(\mathbf{x}_{t-1}) \end{pmatrix} \quad (\text{A16})$$

where:

$$\begin{aligned} \mathbf{A} &= \mathbf{I}_P(\mathbf{x}_t) + \mathbf{I}_D(\mathbf{x}_t) = \mathbf{Q}_t^{-1} + \mathbf{I}_D(\mathbf{x}_t) \\ \mathbf{B} &= -\frac{\partial \mathbf{f}_t(\mathbf{x}_t)}{\partial \mathbf{x}_{t-1}} \mathbf{Q}_t^{-1} \\ \mathbf{C} &= \frac{\partial \mathbf{f}_t(\mathbf{x}_t)}{\partial \mathbf{x}_{t-1}} \mathbf{Q}_t^{-1} \frac{\partial \mathbf{f}_t(\mathbf{x}_t)}{\partial \mathbf{x}_{t-1}}^T \end{aligned} \quad (\text{A17})$$

where $\mathbf{I}(\mathbf{x}_t)$ is a part of $\mathbf{I}([\mathbf{x}_t, \mathbf{x}_{t-1}]^T)$; \mathbf{A} , \mathbf{B} and \mathbf{C} are parts of $\mathbf{I}([\mathbf{x}_t, \mathbf{x}_{t-1}]^T)$, which represents the previous and current information [36]. Then, $\mathbf{I}(\mathbf{x}_t)$ is given based on Schurcompletion:

$$\mathbf{I}(\mathbf{x}_t) = \mathbf{A} - \mathbf{B}(\mathbf{I}(\mathbf{x}_{t-1}) + \mathbf{C})^{-1} \mathbf{B}^T \quad (\text{A18})$$

References

1. Shen, Y.; Win, M. Fundamental limits of wideband localization part I: A general framework. *IEEE Trans. Inf. Theory* **2010**, *56*, 4956–4980.
2. Mazor, E.; Averbuch, A.; Bar-Shalom, Y.; Dayan, J. Interacting multiple model methods in target tracking: A survey. *IEEE Trans. Aerosp. Electron. Syst.* **1998**, *34*, 103–123.
3. Zhao, F.; Liu, J.; Liu, J.; Guibas, L.; Reich, J. Collaborative signal and information processing: An information-directed approach. *Proc. IEEE* **2003**, *91*, 1199–1209.
4. Gao, W.; Zhao, H.; Song, C.; Xu, J. A New Distributed Particle Filtering for WSN Target Tracking. In Proceedings of the 2009 IEEE International Conference on Signal Processing Systems, Singapore, 19–24 April 2009; pp. 334–337.
5. Gu, D. Distributed Particle Filter for Target Tracking. In Proceedings of the 2007 IEEE International Conference on Robotics and Automation, Rome, Italy, 10–14 April 2007; pp. 3856–3861.
6. Bergman, N. Recursive Bayesian Estimation: Navigation and Tracking Applications. Ph.D. Thesis, Department of Electrical Engineering, Linköping University, Linköping, Sweden, 1999.
7. Straka, O.; Simandl, M. A survey of sample size adaptation techniques for particle filters. *IFAC Proc. Vol.* **2009**, *42*, 1358–1363.
8. Farahmand, S.; Roumeliotis, S.; Giannakis, G. Particle Filter Adaptation for Distributed Sensors via Set Membership. In Proceedings of the 2010 IEEE International Conference on Acoustics, Speech and Signal Processing, Dallas, TX, USA, 14–19 March 2010; pp. 3374–3377.
9. Farahmand, S.; Roumeliotis, S.; Giannakis, G. Set-membership constrained particle filter: Distributed adaptation for sensor networks. *IEEE Trans. Signal Process.* **2011**, *59*, 4122–4138.
10. Kwok, C.; Fox, D.; Meila, M. Adaptive Real-Time Particle Filters for Robot Localization. In Proceedings of the 2003 IEEE International Conference on Robotics and Automation, Taipei, Taiwan, 14–19 September 2003; Volume 2, pp. 2836–2841.
11. Kotecha, J.; Djuric, P. Gaussian particle filtering. *IEEE Trans. Signal Process.* **2003**, *51*, 2592–2601.
12. Hlinka, O.; Sluciak, O.; Hlawatsch, F.; Djuric, P.; Rupp, M. Likelihood consensus and its application to distributed particle filtering. *IEEE Trans. Signal Process.* **2012**, *60*, 4334–4349.
13. Geng, L.; Bugallo, M.; Athalye, A.; Djuric, P. Indoor tracking with rfid systems. *IEEE J. Sel. Top. Signal Process.* **2014**, *8*, 96–105.
14. Ristic, B.; Clark, D.; Vo, B.N.; Vo, B.T. Adaptive target birth intensity for phd and cphd filters. *IEEE Trans. Aerosp. Electron. Syst.* **2012**, *48*, 1656–1668.
15. Jung, J.; Myung, H. Indoor User Localization Using Particle Filter and NLOS Ranging Model. In Proceedings of the 2010 International Conference on Control Automation and Systems (ICCAS), Gyeonggi-do, Korea, 27–30 October 2010.

16. Youssef, J.; Denis, B.; Godin, C.; Lesecq, S. Enhanced UWB Indoor Tracking through NLOS TOA Bias Estimation. In Proceedings of the 2008 IEEE Global Telecommunications Conference, New Orleans, LO, USA, 30 November–4 December 2008; pp. 4830–4834.
17. Jung, J.; Myung, H. Indoor Localization Using Particle Filter and Map-Based NLOS Ranging Model. In Proceedings of the IEEE International Conference on Robotics and Automation, Robotics and Automation (ICRA), Shanghai, China, 9–13 May 2011; pp. 5185–5190.
18. Nicoli, M.; Morelli, C.; Rampa, V. A jump markov particle filter for localization of moving terminals in multipath indoor scenarios. *IEEE Trans. Signal Process.* **2008**, *56*, 3801–3809.
19. Wang, J.; Gao, Q.; Yu, Y.; Wang, H. Toward robust indoor localization based on bayesian filter using chirp-spread-spectrum ranging. *IEEE Trans. Ind. Electron.* **2012**, *59*, 1622–1629.
20. Papa, G.; Braca, P.; Horn, S.; Marano, S. Adaptive Bayesian Tracking with Unknown Time-Varying Sensor Network Performance. In Proceedings of the 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), South Brisbane, Australia, 19–24 April 2015; pp. 2534–2538.
21. Song, Y.; Yu, H. A New Hybrid TOA/RSS Location Tracking Algorithm for Wireless Sensor Network. In Proceedings of the 2008 9th International Conference on Signal Processing Signal Processing, Beijing, China, 26–29 October 2008; pp. 2645–2648.
22. Prieto, J.; Mazuelas, S.; Bahillo, A.; Fernandez, P.; Lorenzo, R.M.; Abril, E.J. Adaptive data fusion for wireless localization in harsh environments. *IEEE Trans. Signal Process.* **2012**, *60*, 1585–1596.
23. Bargshady, N.; Pahlavan, K.; Alsindi, N.A. Hybrid WiFi/UWB, Cooperative Localization Using Particle Filter. In Proceedings of the 2015 International Conference on Computing, Networking and Communications (ICNC), Garden Grove, CA, USA, 16–19 February 2015; pp. 1055–1060.
24. Clark, D.; Ristic, B.; Vo, B.N.; Vo, B.T. Bayesian multi-object filtering with amplitude feature likelihood for unknown object SNR. *IEEE Trans. Signal Process.* **2010**, *58*, 26–37.
25. Zhao, Y.; Yang, Y.; Kyas, M. An Adaptive Likelihood Fusion Method Using Dynamic Gaussian Model for Indoor Target Tracking. In Proceedings of the 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Florence, Italy, 4–9 May 2014; pp. 1–4.
26. Hu, X.; Schon, T.; Ljung, L. A basic convergence result for particle filtering. *IEEE Trans. Signal Process.* **2008**, *56*, 1337–1348.
27. 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.
28. Yang, Y.; Zhao, Y.; Kyas, M. A Non-Parametric Modeling of Time-of-Flight Ranging Error for Indoor Network Localization. In Proceedings of the 2013 IEEE Global Communications Conference (GLOBECOM), Atlanta, GA, USA, 9–13 December 2013; pp. 189–194.
29. Prieto, J.; Bahillo, A.; Mazuelas, S.; Lorenzo, R.; Blas, J.; Fernandez, P. NLOS Mitigation Based on Range Estimation Error Characterization in an RTT-Based IEEE 802.11 Indoor Location System. In Proceedings of the IEEE International Symposium on Intelligent Signal Processing, Budapest, Hungary, 26–28 August 2009; pp. 61–66.
30. Schmitt, S.; Will, H.; Aschenbrenner, B.; Hillebrandt, T.; Kyas, M. A Reference System for Indoor Localization Testbeds. In Proceedings of the 2012 International Conference on Indoor Positioning and Indoor Navigation, Sydney, Australia, 13–15 November 2012; pp. 1–4.
31. Chou, R.; Boers, Y.; Podt, M.; Geist, M. Performance Evaluation for Particle Filters. In Proceedings of the 14th International Conference on Information Fusion (FUSION), Chicago, IL, USA, 5–8 July 2011; pp. 1–7.
32. Cover, T.M.; Thomas, J.A. *Elements of Information Theory*, 2nd ed.; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2006.
33. Hendeby, G.; Karlsson, R.; Gustafsson, F.; Gordon, N. Performance Issues in Non-Gaussian Filtering Problems. In Proceedings of the 2006 IEEE Nonlinear Statistical Signal Processing Workshop, Cambridge, UK, 13–15 September 2006; pp. 65–68.
34. Boyd, S.; Vandenberghe, L. *Convex Optimization*; Cambridge University Press: Cambridge, UK, 2004; pp. 291–293.
35. Van Trees, H.L. *Optimum Array Processing (Detection, Estimation, and Modulation Theory, Part IV)*; John Wiley & Sons: Hoboken, NJ, USA, 2002.
36. Tichavsky, P.; Muravchik, C.; Nehorai, A. Posterior cramer-rao bounds for discrete-time nonlinear filtering. *IEEE Trans. Signal Process.* **1998**, *46*, 1386–1396.

37. Yang, Y.; Zhao, Y.; Kyas, M. Weighted Least-Squares by Bounding-Box (B-WLS) for NLOS Mitigation of Indoor Localization. In Proceedings of the 2013 IEEE 77th Vehicular Technology Conference, Dresden, Germany, 2–5 June 2013; pp. 1550–2252.
38. Crisan, D.; Doucet, A. A survey of convergence results on particle filtering methods for practitioners. *IEEE Trans. Signal Process.* **2002**, *50*, 736–746.
39. Kushki, A.; Plataniotis, K.; Venetsanopoulos, A. Intelligent dynamic radio tracking in indoor wireless local area networks. *IEEE Trans. Mob. Comput.* **2010**, *9*, 405–419.



© 2016 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/>).