Abstract: This paper addresses the reliable time propagation algorithms for Point Mass Filter (PMF) and Rao–Blackwellized PMF (RBPMF) for the nonlinear estimation problem. The conventional PMF and RBPMF process the probability diffusion for the time propagation with the direct sampled-values of the process noise. However, if the grid interval is not dense enough, it fails to represent the statistical characteristics of the noise accurately so the performance might deteriorate. To overcome that problem, we propose time propagation convolution algorithms adopting Moment Matched Gaussian Kernel (MMGK) on regular grids through mass linear interpolation. To extend the dimension of the MMGK that can accurately describe the noise moments up to the kernel length, we propose the extended MMGK based on the outer tensor product. The proposed time propagation algorithms using one common kernel through the mass linear interpolation not only improve the performance of the filter but also significantly reduce the computational load. The performance improvement and the computational load reduction of the proposed algorithms are verified through numerical simulations for various nonlinear models.

Keywords: point mass filter; Rao–Blackwellized point mass filter; mass redefinition; Moment Matched Gaussian Kernel

1. Introduction

Recursive Bayesian filtering recursively predicts and corrects an unknown Probability Density Function (PDF) using a mathematical model and incoming measurements. However, it is almost impossible to obtain a closed-form solution for all estimation problems by applying Bayesian filtering technique. As a very limited case, if the model is linear and all the random variables included in the model follow normal distributions, there exists an explicit solution for Bayesian filtering, and that solution is well-known Kalman Filter (KF) [1,2]. For other general problems, two groups of techniques are applied to obtain approximate solutions.

The first group is based on the assumption that, despite the nonlinearity of the model, PDFs to be estimated follow normal distributions approximately. Among them, the most representative nonlinear filter widely applied in various fields is Extended Kalman Filter (EKF) [1,2], which applies KF after linearizing the model. EKF has the advantage of showing excellent asymptotic convergence characteristics despite linearization errors [3,4]. Methods which do not apply model linearization include Unscented Kalman Filter (UKF) and Ensemble Kalman Filter (EnKF) which approximate PDFs to normal distributions [5,6]. Gaussian Filters (GFs) apply Gaussian quadrature integration rules to calculate expectations of nonlinear functions for normal distributions [7].

The second group directly estimate PDFs by applying numerical approximation techniques to solve Bayesian filtering. Particle Filter (PF) is the most representative method. The concept of PF was first proposed in the 1950s, but it was not popular until Gordon proposed the bootstrap algorithm that the PF established itself as a representative nonlinear/non-gaussian estimation technique with high accuracy [1,2,5,7–9]. PF directly estimates
non-gaussian PDFs through the combination of positions and weights of random sampling points. It is known that the estimation error of PF is independent of the dimension of the state variable [10,11]. In the practical application of PF, the increase of the computational load with dimension is still a problem to be tackled [2,9]. Rao–Blackwellized PF (RBPF), which divides the state variable into a nonlinear part and a linear part, and applies PF to the nonlinear part and KF to the linear part, is a state-of-the-art computational load reduction technique [12,13].

PMF is another method obtaining an approximate Bayesian filtering solution [2,5,14]. PMF was introduced in the early 1970s because of its conceptual simplicity [15], and the advantage of the PMF over the PF is its deterministic nature of the algorithm. However, it has not received as much attention as the PF due to the lack of efficient grid design methods or a problem of excessive computation. However, developments of computing technologies led to the introduction of a PMF-based TRN algorithm in the 1990s. Since then, there have been active researches for performance improvements of PMF [16,17]. Most of the studies have been conducted for the purpose of improving TRN performance, but the results are applicable to general estimation problems. Numerous research results for efficient grid design or reselection have been presented to improve PMF performance, and there are the Anticipative Grid Design (AGD) algorithm and the Boundary-based Grid Design (BGD) algorithm [18], the grid resolution and support design algorithm considering a noise level in TRN [19], the grid support design algorithm using mutual information [20], the density specific grid design algorithm assuming two different grids [21], and the density difference grid design algorithm based on the differentiation of the PDF in a sparse grid [22].

In implementing the filter, it is important to process probability diffusion through time propagation for the filter performance. In the case of PMF, time propagation is conceptually a convolution operation between masses of grids and the kernel of the process noise. The kernel is a set of the directly sampled values of the process noise PDF with arguments of differences between irregular grids passing a system model and newly defined regular grids. However, if the grid interval is not dense enough, the conventional kernel cannot accurately represent the statistical characteristics of the noise. This causes a problem in which the probability diffusion is not properly handled so the filter performance might deteriorate [23]. To resolve this problem, Variance Adjusted Gaussian Kernel (VAGK) or Moment Matched Gaussian Kernel (MMGK) have been proposed [23,24]. As another countermeasure, the Density-Weighted Convolution (DWC) algorithm, dealing with a model that only the measurement model is nonlinear, has been proposed recently [25]. Among them, MMGK exactly matches the moments of the noise up to the effective kernel length. However, those are kernel generation techniques applicable only for the model whose system model is linear. That is, it cannot be directly applied to irregular grid intervals due to the nonlinearity of the system model. In this paper, as a first result, we propose the PMF algorithm with indirect time propagation using MMGK through a mass redefinition process for general nonlinear estimation problems. The proposed algorithm has the advantage of not only improving the performance but also reducing the burden of calculating the direct-sampled values of the process noise. Furthermore, we propose the dimension extended MMGK by applying outer tensor product.

Like PF, PMF cannot be free from the problem of the extensive computational burden. In particular, PMF is limited to low-dimensional problems due to the high-dimensional convolution operation. In order to reduce the computational load of PMF, RBPMF, analogous to RBPF, has been proposed relatively recently [26]. RBPMF algorithm for easy implementation of measurement validity check logic, which is essential for maintaining the filter stability in practical application, and RBPMF for TRN estimation problems were proposed [27,28]. Recently, Rao–Blackwellized Particle-Point Mass Fusion Filter (RBPPFF), which combines RBPF and RBPMF, has been introduced for robust TRN [29]. However, the proposed algorithms have the same abnormal probability diffusion problem in the time
propagation operation as the aforementioned PMF. In this paper, as a second result, we propose the RBPMF algorithm applying MMGK without a mass redefinition.

In RBPMF, the weight of each grid for the nonlinear part is paired with the normal distribution for the linear part. Since the linear part acts as an artifact noise to the nonlinear part, it is impossible to apply a common kernel to the time propagation of the nonlinear part. However, if model terms related to the linear part are not functions of the nonlinear part like TRN, all covariances of the linear part approximately have the same value. So, for these specific models, we can apply the time propagation operation to RBPMF through grid and mass redefinition, like the proposed PMF. To complete this scheme, a linear part redefinition procedure following the nonlinear part mass redefinition is required. However, until now, only the index-based adaption algorithm of simply copying the state of the neighboring linear part for the TRN problem has been proposed [30]. Therefore, for the constant linear model case, we propose the RBPMF algorithm with indirect time propagation as the aforementioned PMF. In this paper, as a second result, we propose the RBPMF algorithm applying MMGK without a mass redefinition.

The composition of this paper is as follows. First, we introduce Bayesian filtering in Section 2, then describe the conventional PMF and the proposed PMF algorithm with the indirect time propagation algorithm, and then show the simulation results comparing their performances. Section 4 describes the conventional RBPMF and two proposed RBPMF algorithms, simulation results for them, and concludes in Section 5.

2. Bayesian Filtering

Let us consider the following nonlinear discrete-time stochastic dynamic system.

\[
x_k = f_{k-1}(x_{k-1}) + w_{k-1}
\]

\[
y_k = h_k(x_k) + v_k
\]

where \( x \in \mathbb{R}^n \) is the state variable to be estimated and \( y_k \in \mathbb{R}^m \) is the measurement. It’s assumed that the process noise \( w_k \) and the measurement noise \( v_k \) are white noise and mutually independent of each other, and follow known normal distributions, \( p_w(w_k) \) and \( p_v(v_k) \), respectively. The nonlinear mappings \( f_k : \mathbb{R}^n \rightarrow \mathbb{R}^n \) and \( h_k : \mathbb{R}^n \rightarrow \mathbb{R}^m \) represent the system model and the measurement model. The model in Equation (1) is an input-free model, but can be easily extended to an input-driven one.

The conditional PDFs of the state variable given measurements, which is the estimation target of Bayesian filtering, is \( p(x_k|Y_k) \). Here, \( Y_k = \{y_0, y_1, \ldots , y_k\} \) is a set of all measurements up to time \( t_k \). Two PDFs to be estimated are the priori PDF \( p(x_k|Y_{k-1}) \) which is one step ahead prediction and the posteriori PDF \( p(x_k|Y_k) \) which is filtering.

To solve a recursive Bayesian filtering problem, PDF models are required. First, the transition PDF of the state variable for the system model in Equation (1) is as follows.

\[
x_k \sim p(x_k|x_{k-1}) = p_{w_{k-1}}(x_{k} - f_{k-1}(x_{k-1}))
\]

The transition PDF satisfies \( p(x_k|x_{k-1}, Y_{k-1}) = p(x_k|x_{k-1}) \) from the Markovian characteristic. The PDF model for the measurement equation is as follows.

\[
y_k \sim p(y_k|x_k) = p_v(v_k - h_k(x_k))
\]

\[
p(y_k|x_k, Y_{k-1}) = p(y_k|x_k) \]

is satisfied because \( y_k \) is dependent only on \( x_k \) from the measurement model.

Bayesian filtering is a process of recursively obtaining the posteriori PDF \( p(x_k|Y_k) \) for the current time under the assumption that the posteriori PDF \( p(x_{k-1}|Y_{k-1}) \) for the previous time is known. To solve this, first, apply Bayes’ theorem to the posteriori PDF, then \( p(x_k|Y_k) \) becomes as in Equation (5).

\[
p(x_k|Y_k) = \frac{p(y_k|x_k, Y_{k-1})p(x_k|Y_{k-1})}{p(y_k|Y_{k-1})}
\]
If \( p(y_k|x_k, y_{k-1}) = p(y_k|x_k) \) is applied to Equation (5), \( p(x_k|y_k) \) can be rewritten as in Equation (6).

\[
p(x_k|y_k) = \frac{p(y_k|x_k)p(x_k|y_{k-1})}{p(y_k|y_{k-1})}
\]

(6)

The first term of the nominator represents the likelihood as the measurement PDF of Equation (4), and the second term is the priori PDF, which can be obtained by the Chapman–Kolmogorov equation.

\[
p(x_k|y_{k-1}) = \int p(x_k|x_{k-1}, y_{k-1})p(x_{k-1}|y_{k-1})dx_{k-1}
\]

(7)

The denominator \( p(y_k|y_{k-1}) \) of Equation (6) is called the evidence and can be calculated by integrating the nominator.

\[
p(y_k|y_{k-1}) = \int p(y_k|x_k)p(x_k|y_{k-1})dx_k
\]

(8)

That is, \( p(x_k|y_{k}) \) can be obtained by calculating Equations (6)–(8) for \( p(x_{k-1}|y_{k-1}) \).

3. PMF with Reliable Time Propagation

3.1. Conventional PMF with Direct Time Propagation

To obtain the posteriori PDF in closed-form from Bayesian filtering, the integrals of Equations (7) and (8) must be explicitly calculated. However, it is almost impossible to solve such integrals for general nonlinear models. Therefore, numerical techniques are usually used to obtain approximate solutions, and PMF is one of those approximation methods. The basic concept of PMF is to discretize a state space into equally spaced grids and to calculate a mass (or a weight) in each grid to discretely express the PDF. That is, suppose that the mass \( \omega^i_{k-1|k-1} \) of the each grid \( \zeta^i_{k-1} \) from the grid set \( \Xi_{k-1}(N_{k-1}) = \{ \zeta^i_{k-1}; i = 1, \cdots, N_{k-1} \} \) defined at time \( t_{k-1} \) is as follows.

\[
\omega^i_{k-1|k-1} \propto p(x_{k-1} = \zeta^i_{k-1}|y_{k-1})
\]

(9)

Then, the discretely approximated posteriori PDF at time \( t_{k-1} \) can be expressed as Equation (10).

\[
\hat{p}(x_{k-1}|y_{k-1}) \simeq \sum_{i=1}^{N_{k-1}} \omega^i_{k-1|k-1}\delta(x_{k-1} - \zeta^i_{k-1})
\]

(10)

where \( \delta(\cdot) \) is dirac delta function. Previous PMF-related literatures considered the mass \( \omega^i_{k-1|k-1} \) as the value of the pdf at \( \zeta^i_{k-1} \). Therefore the rectangular volume term \( \Delta \zeta^i \) around \( \zeta^i_{k-1} \) should be included in Equation (10). However, if assuming equal grid intervals and \( \sum_i \omega^i_{k-1|k-1} = 1 \) for the masses, then the volume is a common term for all masses so the volume can be viewed as included in the mass. This can reduce unnecessary multiplication operations when implementing the algorithm.

When the PDF of Equation (10) is applied to recursive Bayesian filtering, the integral equations of Equations (7) and (8) are converted to discrete summation equations, so that the discretized posteriori PDF at time \( t_k \) can be obtained in a similar manner. To apply PMF to general nonlinear models, an adaption procedure for irregular grid intervals, grid support, and grid resolution due to nonlinearity of the model must be included. The conventional PMF algorithm including that procedure is as follows [18].
Algorithm 1 Conventional PMF

1. **Initialization** Define the initial grid set and the masses for the initial priori PDF \( p(x_0|y_{-1}) \); \( \Xi_0(N_0) = \{ \xi_i^0; i = 1, \cdots, N_0 \} \), \( \omega_{0|-1}^i = p(x_0 = \xi_i^0|y_{-1}) \). Set \( k = 0 \).

2. **Measurement Update** Calculate the measurement updated masses for all \( i = 1, \cdots, N_k \)
   \[
   \omega_{k|k}^i = c_k^{-1} \omega_{k|k-1}^i p_{V_k}(y_k - h_k(\xi_i^k))
   \]
   \[c_k = \sum_{i=1}^{N_k} \omega_{k|k-1}^i p_{V_k}(y_k - h_k(\xi_i^k)) \text{ (Normalization Constant)}\]

3. **Grid Propagation** Calculate the nonlinear mapped grid set \( H_k+1(N_k) = \{ \eta_{k+1}^i; i = 1, \cdots, N_k \} \)
   \[
   \eta_{k+1}^i = f_k(\xi_i^k)
   \]

4. **Grid Redefinition** Redefine the grid set \( \Xi_{k+1}(N_{k+1}) = \{ \xi_{k+1}^j; j = 1, \cdots, N_{k+1} \} \) with regular grid spacing from \( H_k+1(N_k) \)

5. **Time Propagation** Calculate the predicted masses for all \( j = 1, \cdots, N_{k+1} \)
   \[
   \omega_{k+1|k}^j = \sum_{i=1}^{N_k} \omega_{k|k}^i p_{W_k}(\xi_{k+1}^j - \eta_{k+1}^i)
   \]

6. Update \( k := k + 1 \) and repeat (2)-(5)

Various algorithms can be applied to the grid redefinition of Step (4). Among proposed algorithms, the algorithms suitable for general estimation problems are AGD and BGD proposed by Šimandl [18]. AGD is an algorithm that selects grid support and grid resolution considering the performance of time propagation calculation. AGD assumes Gaussian distribution when selecting grid support, whereas BGD is a grid support selection algorithm that considers non-Gaussian distribution. Suitable algorithms for TRN include the grid resolution/support adaption algorithm considering noise magnitude [19], and the grid support adaption algorithm using mutual information [20]. The density specific grid design algorithm that assumes two different grids [21] and the density difference grid design algorithm based on the differentiation of the PDF in a sparse grid [22] have been presented recently for general estimation problems.

PMF is a global approximation nonlinear filter, so its application range is very wide. However, to apply PMF, we need to know exactly the time evolution model of Equations (1) and (2) or the probabilistic model of Equations (3) and (4). If there are uncertainties in the model or disturbances that cannot be modeled, such terms can be considered as process noise, but the estimation performance may be degraded. The Takagi–Sugeno (T-S) fuzzy affine model is known to be very effective in dealing with such uncertainties and disturbances. As the state-of-the-art result in that field, a sampled-data filtering design technique for l_f stochastic T-S fuzzy affine system has been proposed recently [31].

3.2. PMF with Indirect Time Propagation

If the probability diffusion of the mass through time propagation is not properly processed, only the contraction of the probability distribution by the measurement update is repeated over time. This eventually causes the filter to diverge because the measurement can be no longer reflected in the error correction. Conversely, the excessive diffusion of the probability is the same as losing estimation information so far, which might degrade the estimation performance. That is, in the implementation of filtering, the proper diffusion process is very important for the stability and the performance of the filter.

If the interval between grids is greater than about 1.5 times the standard deviation of the process noise, the kernel for probability diffusion by sampling does not adequately reflect the statistical characteristics of the original noise [23]. VAGK, MMGK, and DWC
have been proposed to resolve this problem [23–25]. VAGK generation method creates the kernel according to the conventional method and scales up it to match only the variance. So it cannot deal with the process noise whose mean is not zero. MMGK is the structured kernel obtained through linear equations where kernel $K_t$ of length $I$ matches moments up to $I - 1$. The DWC method uses the difference between the Cumulative Distribution Functions (CDFs) at the upper and lower limits of the volume near the grid. In order to find a kernel generation technique that accurately expresses the variance in the sense of probability diffusion, we compared the variances of the various kernels while varying the variance of the noise. Figure 1 shows the comparison result. The x-axis and y-axis of the graph represent the ratio of the standard deviation of the process noise to the grid interval and the ratio of the variance of the generated kernel to the variance of the noise, respectively. The variance ratio of the conventional kernel gradually decreases as the standard deviation ratio becomes smaller than 0.67. The DWC kernel has a slightly larger value than the original variance when the standard deviation ratio is greater than 0.29 but gradually decreases in the interval less than 0.29. On the other hand, unlike the previous two kernels, MMGK has a variance that exactly matches the original variance regardless of the standard deviation ratio. So we adopt the MMGK as the time propagation kernel.

Adopting MMGK significantly improves the PMF’s inconsistency problem especially when the standard deviation of the process noise is quite small. However, MMGK was proposed to apply to the TRN problem that the system model is an identity matrix. In this paper, to resolve the inadequate probability diffusion problem, the general PMF algorithm adopting MMGK is proposed.

MMGK can be applied in the case of regular grid interval, but the kernel in Step (5) of Algorithm 1 is the sampled values of the process noise with the difference between element $\eta_{i+1}$ of the irregular grid set $H_{k+1}(N_i)$ and element $\xi_{k+1}$ of the newly defined regular grid set $\Xi_{k+1}(N_{k+1})$. Therefore, it is impossible to directly apply MMGK to the conventional PMF algorithm described in the previous section. However, if the convolution operation is performed only on the new grids set $\Xi_{k+1}(N_{k+1})$, MMGK can be applied immediately. That is, after redefining mass $\gamma_{ik|k}$ for $\xi_{k+1}$ with $\omega_{ik|k}$, the posteriori mass for grid $\eta_{k+1}$, the convolution operation is performed indirectly for $\gamma_{ik|k}$. $\gamma_{ik|k}$ calculation can be easily implemented through various well-known multivariate interpolation algorithms for irregular grids [32]. Figure 2 shows the comparison of the probability diffusion process concepts of the previous method and the proposed one (in order to explain the probability diffusion in the 2D state space, the grid indeces are expressed in 2D).

![Figure 1](image1.png)

**Figure 1.** Comparison of the variances of the various kernels, the conventional kernel, MMGK, and DWC.

In the conventional method, the probability values adjusted in proportion to the sampled value of the process noise for all $\eta_{k+1}$ within the effective distance of process noise near $\xi_{k+1}$ (for example, 3 times the standard deviation of the process noise) are diffused and accumulated in $\xi_{k+1}$. On the other hand, the proposed method first calculates the new mass $\gamma_{ik|k}$ for each $\xi_{k+1}$. A variety of interpolation algorithms can be applied, but in this
paper, mass $\gamma_{k+1}^i$ is calculated as the linear combination of $\eta_{k+1}^i$ around $\zeta_{k+1}^i$ as follows by applying linear interpolation.

$$
\gamma_{k+1}^i = \sum_{s \in S_i} \lambda_{s,i} \omega_{s,k}^i,
$$

where $\lambda_{s,i}$ is the linear combination coefficient of $\eta_{k+1}^i = f_k(\zeta_{k+1}^i)$ concerning for $\zeta_{k+1}^i$, which satisfies $\zeta_{k+1}^i = \sum_{s \in S} \lambda_{s,i} \omega_{s,k}^i$ and $\omega_{s,k}^i$ represents the nonlinear transformation of the probability $\omega_{s,k}^i$ from $\zeta_{k}^i$ to $\eta_{k}^i$. Equation (12) stems from the relationship $f_k'(y)dy = f_k(x)dx$ between two random variables $x$ and $y$ and the determinant of the Jacobian $\partial f_k(x_k)/\partial x_k$ for the system model $f_k(x_k)$ represents $d\eta_{k+1}^i/d\zeta_{k+1}^i[33]$. The condition of $\sum_{s \in S} \lambda_{s,i} = 1$ for the coefficient $\lambda_{s,i}$ means that $\zeta_{k+1}^i$ is a value generated by interpolation. If the interpolation is unavailable, the corresponding mass is assigned as 0. The mass transformation in Equation (12) is only applicable when $f_k(x_k)$ is an invertible function such that the solution of $\zeta_{k+1}^i = f_k(x_k)$ is one. If there are multiple solutions, it is necessary to calculate the mass for each solution and add up every transformed mass.

The indirect time propagation operation using the new mass $\gamma_{k+1}^i$ by the linear interpolation is shown in Equation (13).

$$
\omega_{k+1}^i = \sum_i \gamma_{k+1}^i P_{s,k}^i (\zeta_{k+1}^i - \zeta_{k+1}^i)
$$

The time propagation equation in step (5) of Algorithm 1 and the equation in Equation (13) are mathematically the same. The only difference is that the sampling in Equation (13) is processed on equally spaced grids while the sampling interval of the process noise in step (5) is irregular. Therefore, it is possible to design a structured kernel to accurately reflect the statistics of the process noise. In this paper, MMGK is adopted as the kernel.

The MMGK generation method is as follows [24]. First, let $M_{m,k}$ be the $k$-th moment of the process noise whose mean and variance are $\mu_m$ and $\sigma_m^2$, respectively. For the grids set $\xi_{k+1} (N_{k+1})$ where the grid spacing of the $m$-th state variable is $\Delta_m$, considering the effective support of the noise only up to the $\pm 3\sigma$ range, it is sufficient to generate a discrete kernel for $x_s = s \cdot \Delta_m + x_{s,0}, s \in [L_m, U_m], L_m = \lfloor (\mu_m - 3\sigma_m - x_{s,0}) / \Delta_m \rfloor, U_m = \lceil (\mu_m + 3\sigma_m + x_{s,0}) / \Delta_m \rceil, s \in [L_m, U_m]$. Using the mean and variance of the process noise, the probability density function (PDF) of $x_s$ is calculated as

$$
\rho_s(x) = \frac{1}{\sqrt{2\pi}\Delta_m} \exp\left(-\frac{(x - (s \cdot \Delta_m + x_{s,0}))^2}{2\sigma_m^2}\right)
$$

If the interpolation is unavailable, the corresponding mass is assigned as 0. The mass $\omega_{k+1}^i$ is calculated as the linear combination of $\eta_{k+1}^i$ around $\zeta_{k+1}^i$ as follows by applying linear interpolation.
where \( I_m = U_m - L_m + 1 \) is the length of the kernel, and \( K_m(s) \) is the element of the kernel. It is the 1-dimension kernel for the process noise. If two dimensional problems are dealt, vector outer product is enough to adopt the MMGK. In this paper, we propose the dimension extended MMGK by outer tensor product. The extended MMGK is a combination of \( K_m \) generated for each process noise, as shown in Equation (15).

\[
Ker = K_1 \circ K_2 \circ \cdots \circ K_n
\]

where \( \circ \) represents the tensor product (in the sensor of the outer product) [34]. When the generated kernel \( Ker \) is applied to the time propagation, the time propagation or the probability diffusion of step (5) for newly defined mass \( \gamma_{k|k}^{(i)} \) as shown in Equation (16).

\[
\omega_{k+1|k}^{(i)} = \sum_{s \in Ker} \gamma_{k|k}^{(i)} p_{w_3}(s_{k+1}^{(i)} - \xi_{k+1}^{(i)}) = \sum_{s \in Ker} \gamma_{k|k}^{(s)} Ker(s)
\]

where \( s = j - i \), and \( s \in Ker \) means that the index \( s \) is within the valid range of \( Ker \). Here the mass is represented by the one-dimensional index, but the kernel \( Ker \) is the \( n \)-dimensional tensor. Therefore, to implement the above equation, an appropriate transformation must be included between the 1-dimension mass index and the \( n \)-dimensional kernel index. If the mass is expressed and processed with a \( n \)-dimensional index like the kernel, Equation (16) can be rewritten as Equation (17).

\[
\omega_{k+1|k}^{(i_1, i_2, \ldots, i_n)} = \sum_{s_1 = L_1}^{U_1} \sum_{s_2 = L_2}^{U_2} \cdots \sum_{s_n = L_n}^{U_n} \gamma_{k|k}^{(s_1, s_2, \ldots, s_n)} K_1(s_1) K_2(s_2) \cdots K_n(s_n)
\]

The proposed time propagation algorithm has the following advantages and disadvantages. In the proposed method, the time propagation performs the convolution through MMGK on grids of equal spacing, so the probability diffusion is more accurate than the conventional one. The conventional method has to calculate the distance between one new grid and all previous grids and examine whether it is within the effective range for the noise probability distribution. On the other hand, the proposed method uses the kernel which considers the effective length of the noise for a new set of grids, so such process is unnecessary. However, the proposed method must additionally perform the interpolation operation for the masses. Bergman’s grid adaption algorithm for TRN is a special case of indirect time propagation that performs only interpolation and decimation of two times intervals [17]. The PMF algorithm including the new indirect time propagation algorithm proposed in this section is as follows.
Algorithm 2  PMF with Indirect Time Propagation

1: **Initialization**  Same as (1) of Algorithm 1
2: **Measurement Update**  Same as (2) of Algorithm 1
3: **Grid Propagation**  Same as (3) of Algorithm 1
4: **Grid and Mass Redefinition**  Redefine the grid set $\Xi_{k+1}$ as $\{\xi_{j}^{+1}; j = 1, \cdots, N_{k+1}\}$ from $H_{k+1}(N_{k})$ and calculate the interpolated masses for all $\xi_{j}^{+1}$:

$$\omega_{j}^{s} = p(x_{k+1} = \eta_{j}^{s} + \nu_{k}) = \omega_{j}^{s}/|\partial f_{k}(x_{k})/\partial x_{k}|_{x_{k} = \xi_{j}^{s}}$$

$$\gamma_{j}^{s} = \sum_{s \in S_{i}} \lambda_{s} \omega_{j}^{s}$$

If $\eta_{j}^{s}$ has two or more solutions, repeat the linear interpolation for each solution and calculate their total sum. Calculate the total kernel $Ker$ as a tensor product after finding MMGK $K_{m}$ for each process noise.

5: **Time Propagation**  Calculate the predicted masses with $\gamma_{j}^{s}$ and $Ker$ for all $j = 1, \cdots, N_{k+1}$

$$\omega_{j}^{s} = \sum_{s \in Ker} \gamma_{j}^{s} Ker(s)$$

6: **Update**  $k := k + 1$ and repeat (2)–(5)

The consistency between the estimation error and the covariance of the filter is very important to ensure the reliable operation of the filter. Specifically, PMF that estimates PDFs by mass on a discrete grid requires consistent handling of the process of probability diffusion through time propagation. However, if the grid interval compared to the variance of the process noise is not dense, the variance of the conventional kernel is treated less than the original design. So, over time, the filter behaves as if it only does measurement updates without time propagation. Eventually, the covariance of the filter gets smaller and smaller, so the measurement update no longer works. However, the proposed PMF adopted MMGK for probability diffusion. MMGK can accurately handle at least the noise variance, so the filter’s consistency can be reliably maintained. Therefore, the performance of the proposed PMF can be improved.

The time propagation of the conventional PMF (step (5) of Algorithm 1) uses the value obtained by directly sampling the process noise for the probability diffusion, and this sampling must be performed $N_{k+1} \times N_{k}$ times. However, the proposed PMF uses MMGK that is independent of the number of grids, and the size of the MMGK is much smaller than the total number of grids, so the computation time of time propagation can be drastically reduced. However, the proposed PMF needs to perform mass interpolation on the new grid set before performing probability diffusion. If the nonlinearity of the system model is too large, the interpolation has to be repeated several times, which can increase the computation time.

3.3. Numerical Examples
3.3.1. One Dimensional Growth Model

To verify the performance improvement of the proposed PMF algorithm, a simulation was performed on the following non-stationary growth model used by Gordon et al. in a paper that proposed a bootstrap filter [2,8,26].

$$x_{k+1} = a \cdot x_{k} + b \cdot \frac{x_{k}}{1 + x_{k}^2} + c \cdot \cos(k) + w_{k}$$

$$y_{k} = d \cdot x_{k}^2 + v_{k}$$

(18)
where each parameter of the model is \( a = 0.5, b = 25, c = 8, d = 0.05 \), respectively. The measurement noise is \( v_k \sim N(0, 1^2) \), and the initial error and the process noise are considered in several cases for performance comparisons. The state variable of the model with the given parameters does not exceed the range \([-25, +25]\) in spite of the process noise so the grid adaption process is not mandatory in this problem. Therefore, the grid set may be fixedly determined according to a predetermined grid interval. In this paper, simulations were performed for a total of four grid intervals. Besides, the same simulation was performed with the bootstrap filter of 1000 particles regardless of the performance change with the grid interval.

First, we compared the PDF outputs estimated by the PMFs and the PF (here, bootstrap filter) for several cases, and Figure 3 shows the results. The pdf of the PMF is just a sampled-function at the grid set. In the case of PF, PDF can be obtained by applying a window function in the form of a normal distribution of small variance to each particle and then summing up all the window functions. The result of \( k = 12 \) for the case where the initial error and the variance of the process noise are \( x_0 \sim N(1, 5^2) \) and \( Q_k = 2^2 \), respectively, and the grid interval \( \Delta \xi \) is 0.1 such that the grid interval is sufficiently small compared to the process noise, is shown in (a). As you see, all three algorithms show similar PDFs. In this case, the conventional PMF showed more similar results to the PF. However, in the case of (b) where the process noise variance is reduced to \( 0.3^2 \), the conventional PMF shows the completely different aspect of the PDF from the proposed PMF or the PF. In particular, in the range of \([-5, +15]\), the probability distribution of the PF and the proposed PMF shows a moderate decrease trend, whereas the conventional PMF tends to fluctuate. When \( \Delta \xi \) is set to 0.5 for \( Q_k = 0.3^2 \), the PDF estimates for \( k = 11, 12 \) are shown in (c) and (d).

In the case of \( x_0 \sim N(5, 5^2) \), the 100 times Monte Carlo simulation results varying the grid interval \( \Delta \xi \) and process noise variance \( Q_k \) are shown in Figure 4. The \( x \)-axis of each graph is the ratio of process noise standard deviation to grid resolution, which means that the smaller the value, the greater the grid interval than the process noise effective range. The \( y \)-axis represents the Root Mean Square (RMS) of the estimation error of the simulations. From the results, if the grid interval is sufficiently small (\( \Delta \xi = 0.1 \)), there are no differences in performance regardless of the ratio. On the other hand, in other grid

![Figure 3](image-url)
interval cases, if the grid interval is not sufficiently dense ($\sqrt{Q_k}/\Delta \xi < 1$), the performance of the proposed PMF is superior to that of the conventional PMF. In particular, the estimation error of the new PMF tends to decrease like the PF as the ratio decreases, whereas the estimation error of the conventional PMF increases again when $\sqrt{Q_k}/\Delta \xi < 0.15$. Table 1 shows the numerical results of the simulations. The number of grids of the PMFs for grid intervals 1.0, 0.5, 0.3, and 0.1 were 51, 101, 167, and 501, respectively.

![Figure 4. Comparison of the estimation errors between the conventional PMF, the proposed PMF, and the PF for various grid intervals and process noise variances](image)

To verify the effect of the computational load reduction of the proposed algorithm, the computation time was measured every epoch. We performed the simulation on the Matlab 2020b single thread environment on Windows 10 operating system with Intel i7-10750H 2.6GHz CPU and 32GB DDR4 Memory. Furthermore, we used vectorized operations and functions to shorten the execution time as much as possible.

**Table 1.** Estimation errors of the conventional PMF, the proposed PMF, and PF for various grid intervals and process noise variances.

| Grid Resolutions | Algorithms       | Process Noise Standard Deviation Ratio vs. Grid Resolution |
|------------------|------------------|-----------------------------------------------------------|
|                  |                  | 3.0  | 2.0  | 1.5  | 1.0  | 0.75 | 0.5  | 0.2  | 0.15 | 0.1  |
| 1.0              | Conventional PMF | 3.998| 3.113| 2.924| 3.018| 2.789| 2.773| 2.512| 2.399| 2.334|
|                  | Proposed PMF     | 4.153| 3.391| 3.052| 2.918| 2.458| 2.469| 1.689| 1.684| 1.618|
|                  | Bootstrap PF     | 4.059| 3.077| 2.760| 2.644| 2.259| 2.125| 1.379| 1.233| 1.187|
| 0.5              | Conventional PMF | 2.815| 2.744| 2.598| 2.543| 2.250| 2.122| 1.990| 2.022| 2.345|
|                  | Proposed PMF     | 2.715| 2.557| 2.254| 2.091| 1.810| 1.611| 1.431| 1.311| 1.311|
|                  | Bootstrap PF     | 2.652| 2.493| 2.222| 2.089| 1.821| 1.525| 1.107| 1.049| 0.968|
| 0.3              | Conventional PMF | 2.535| 2.387| 2.242| 1.955| 1.975| 1.783| 1.969| 1.959| 2.567|
|                  | Proposed PMF     | 2.431| 2.231| 2.048| 1.728| 1.634| 1.441| 1.299| 1.144| 1.265|
|                  | Bootstrap PF     | 2.446| 2.221| 2.058| 1.852| 1.557| 1.307| 1.013| 0.826| 0.852|
| 0.1              | Conventional PMF | 1.709| 1.369| 1.381| 1.111| 1.230| 1.144| 0.858| 0.835| 0.744|
|                  | Proposed PMF     | 1.678| 1.336| 1.379| 1.078| 1.128| 1.229| 0.963| 0.890| 0.805|
|                  | Bootstrap PF     | 1.713| 1.334| 1.352| 1.029| 1.083| 1.087| 0.598| 0.530| 0.399|
Figure 5 shows the results of measuring the calculation time for various grid intervals and process noises. The result of Figure 5a is the average value of the calculation time for various process noises. In both algorithms, it can be seen that the computational load increases as the grid interval becomes narrower. However, while the time of the proposed algorithm did not change significantly, the time of the conventional one increases exponentially. In particular, when the grid interval is 0.1, the proposed one is about 12.5 times faster than the conventional one. On the other hand, when the grid spacing is 1, the proposed one is about 1.42 times slower. This is because a total of five linear interpolations were performed for mass redefinition considering the high nonlinearity of the growth model. Since mass redefinition process occupies about 70% of the computation time of the proposed algorithm, reducing the number of mass redefinitions can significantly reduce the overall execution time.

Figure 5. Comparison of the computation time between the conventional PMF and the proposed PMF for various grid intervals and process noise variances (a) computation times v.s. grid intervals (b) computation time v.s. process noise variances for two grid intervals.

Figure 5b shows the calculation time result versus process noise for the case where 0.3 and 0.5 grid intervals. As mentioned earlier, the proposed algorithm did not show a large variation along the grid interval, while the time of the conventional algorithm increased by 2.2 times as the grid interval increased 1.67 times. The time of the proposed algorithm also hardly changes even when the process noise changes, but in the conventional algorithm, the calculation time decreases as the process noise decreases. This is because exponential function calculation is required when performing the probability diffusion due to process noise, and in general numerical calculations, it is treated as 0 if the exponent of the exponential function is less than a certain value.

3.3.2. Two Dimensional Body Fall Problem

As another numerical example for PMF, we performed a simulation for the two dimensional body fall problem. The mathematical model of the body fall problem is as follows [1].

\[
\begin{align*}
\dot{x}_1 &= x_2 + w_1 \\
\dot{x}_2 &= \rho_0 \exp\left(-x_1/k\right) x_2^2 b_c / 2 - g + w_2 \\
y(t_k) &= \sqrt{M^2 + (x_1(t_k) - a)^2} + v_k
\end{align*}
\]

As usual, \(w_i\) is the process noise and \(v_k\) is the measurement noise, respectively. The two state variables \(x_1\) and \(x_2\) represents the altitude and the velocity of the body, respectively. \(\rho_0\) is the air density at sea level, \(k\) is a constant for the relationship between air density and altitude, \(g\) is the gravity, and \(b_c\) is the ballistic coefficient. We use a discretized system model with a step size of 100 ms and the measurement is obtained at every 0.5 s.
A range measuring device is located at an altitude \(a\) and the horizontal range between the device and the body is \(M\). The constants that we use are given as

\[
\begin{align*}
\rho_0 &= 105 \text{ kg s}^{-2} \text{ m}^4 \\
g &= 9.8 \text{ m s}^{-2} \\
k &= 5100 \text{ m} \\
b_c &= 6.24 \times 10^{-5} \text{ m}^3 \text{ kg}^{-1} \text{ s}^{-2} \\
M &= 10,000 \text{ m} \\
a &= 10,000 \text{ m}
\end{align*}
\]

The initial conditions of the system and the filter are given as \(x_0 = [40,000 -3000]^T\), \(P_0 = \text{diag}(100^2, 5^2)\), \(Q_k = \text{diag}(1^2, 0.1^2)\), and \(R_k = 10^2\). The grid size of the PMF is 101 \times 21 and the simulation time is 30 s. For the first few seconds, the velocity is slowly decreased. However, then the air density increases and drag slows the falling object. Toward the end of the simulation, the body reaches a constant terminal velocity.

Figure 6 shows the altitude and velocity RMS errors of 100 times Monte Carlo simulation. The proposed algorithm shows excellent performance in the entire time domain. The altitude error of the proposed algorithm tends to increase slightly between 5 to 20 s interval which the velocity changes rapidly, but after 20 s, the error stably decreases again. On the other hand, the altitude estimation error of the conventional one tends to increase gradually even after 20 s. Likewise, the velocity error increases slightly from 5 s when the velocity starts to change rapidly but stabilizes again after about 12 s. The calculation time was measured under the computing conditions mentioned in 3.3.1. The conventional one took about 30.45 ms for time propagation, but the proposed algorithm took only 5.32 ms. The measurement update of both algorithms took about 0.05 ms. In other words, the proposed algorithm performed about 5.67 times faster than the conventional algorithm while showing superior performance.

Figure 6. Comparison of the estimation errors between the conventional PMF, the proposed PMF for two dimensional body fall problem (a) altitude estimation error (b) velocity estimation error.

It is enough to apply the conventional PMF algorithm if the grid interval is sufficiently small. However, the realtime application of the nonlinear estimation filter might be very limited in computational power depending on the system payload. In that case, the performance is likely to deteriorate when the conventional PMF is applied. On the other hand, in particular, the process noise is small, but if the relative grid interval is not dense sufficiently, it is confirmed through the simulations that the proposed algorithm improves the performance and decreases the computation load over the conventional method. Recently, many studies for the accurate position determination of the small unmanned vehicles such as drones have been conducted, and the estimation models of the localization technique...
using radio wave, vision, and distance information are generally nonlinear. Therefore, in the sense of the estimation performance and computational load, the proposed algorithm is particularly effective when applied to small vehicles with limited payload in relation to power consumption. Regardless, according to the simulation results, the PF has better performance than the PMFs. However, due to the characteristics of the algorithm, it is known that PMF of the deterministic nature has a superior robustness to PF [35].

4. Rao–Blackwellized PMF with Reliable Time Propagation

The algorithmic complexity of PMF is $O(N^2)$ due to the convolution operation of the time propagation step where $N$ is the number of grids. Furthermore, generally, $N$ is an exponential function of the dimension $n$ of the state variable. So the calculation amount of PMF increases exponentially as $n$ increases. The Rao–Blackwellization technique is a representative method to reduce the computational complexity for high-dimensional nonlinear estimation problems based on PF [12,13]. In this technique, when the estimation model can be separated into a nonlinear part and a linear part, nonlinear filtering is applied only to the nonlinear part, and Kalman filtering is applied to the linear part to make the dimension of the nonlinear filter as small as possible. It has been developed and applied to PF earlier, and Šmídl was the first to propose the RBPMF relatively recently [26]. However, since all previous RBPMFs are based on the conventional PMF, there is a problem of degrading the filter stability due to the abnormal probability diffusion for the nonlinear part.

Therefore, in this paper, like the PMF proposed in the previous section, we propose an RBPMF algorithm using MMGK that can more accurately process the probability diffusion. The simulation results to verify the effectiveness of the proposed RBPMF are described. Finally, an RBPMF algorithm with the same indirect time propagation as the PMF proposed in the previous section is proposed for a special case.

4.1. Conventional Rao–Blackwellized PMF

Let us consider the following model where the state variable can be separated into a nonlinear part and a linear part [12].

\[
\begin{align*}
x^n_k &= f^n_{k-1}(x^n_{k-1}) + F^n_{k-1}(x^n_{k-1})x^l_{k-1} + w^n_{k-1} \\
x^l_k &= f^l_{k-1}(x^l_{k-1}) + F^l_{k-1}(x^l_{k-1})x^l_{k-1} + w^l_{k-1} \\
y_k &= h_k(x^n_k) + H_k(x^n_k)x^l_k + v_k
\end{align*}
\]  

(20)

where the state variable, the process noise, and the measurement noise are as follows.

\[
x_k = \begin{bmatrix} x^n_k \\ x^l_k \end{bmatrix}, \quad w_k = \begin{bmatrix} w^n_k \\ w^l_k \end{bmatrix} \sim \mathcal{N}(0, Q_k), \quad Q_k = \begin{bmatrix} Q^n_k & 0 \\ Q^l_k & Q^l_k \end{bmatrix}, \quad v_k \sim \mathcal{N}(0, R_k)
\]  

(21)

Then, the posteriori PDF for the decomposed state variable can be divided into two conditional PDFs as shown in Equation (22).

\[
p(x_k|Y_k) = p(x^n_k, x^l_k|Y_k) = p(x^n_k|Y_k)p(x^l_k|Y_k)
\]  

(22)

Here, an important assumption is that the conditional PDF $p(x^n_k|Y_k)$ for the linear part $x^l_k$ given the nonlinear part $x^n_k$ follows approximately a normal distribution. Therefore, KF is applied to the linear part and PMF is applied only to the PDF estimation of the nonlinear part.

There are two important aspects from the model in constructing the RBPMF algorithm. First, when processing the nonlinear part, the two terms $F^n_{k-1}(x^n_{k-1})x^l_{k-1}$ and $H_k(x^n_k)x^l_k$ due to the linear part are regarded as additional noises. This makes the variances of the effective process and measurement noises be larger when estimating the nonlinear part. Furthermore, in general, the two noises have been considered as zero-mean noise. However, in the case of RBPMF, the effective process and measurement noises are changed
to nonzero normal distribution due to the influences of the linear part (see Equations (A4) and (A8)). The second aspect is that the system model for the nonlinear part is processed as an artifact measurement for the linear part and the artifact measurement model must be processed before the system model for the linear part. To understand how these two perspectives are triggered, we described the PDF update procedure for RBPMF divided into six steps in the Appendix A. For the time propagation step for the linear part, if \( w_{k}^l \) and \( w_{k}^l \) are correlated with each other, the artifact measurement should be handled carefully, and Schönhoff proposed the well-established algorithm for this [12]. Figure 7 shows an example of a PDF estimated by PMF that treats a state variable as a two-dimensional nonlinear part for a two-dimensional estimation problem, and a PDF estimated by RBPMF composed of a one-dimensional nonlinear part and a one-dimensional linear part.

![Illustrations of PMF and RBPMF concepts for two dimensional estimation problem](image)

Figure 7. Illustrations of PMF and RBPMF concepts for two dimensional estimation problem (a) PMF (b) RBPMF.

The conventional RBPMF algorithm based on the Bayesian framework summarized in the Appendix A is as follows.

**Algorithm 3 Conventional RBPMF**

1. **Initialization** Define initial grids, masses for the nonlinear part PDF \( p(x_{0}^{n}|y_{-1}) \), and initial normal distributions for linear part PDF \( p(x_{0}^{n}|x_{0}^{n}, y_{-1}) \); \( \Xi_{0}(N_{0}) = \{ \xi_{0}^{[i]}, i = 1, \cdots, N_{0} \}, \omega_{0}^{[i]} = p(x_{0}^{n}|y_{-1}), p(x_{0}^{n}|y_{-1}) = \mathcal{N}(x_{0}^{n}|\bar{y}_{0}, \tilde{y}_{0}, y_{-1}) \). Set \( k = 0 \).

2. **Measurement Update for Nonlinear State** Calculate measurement updated masses.

\[
\omega_{k|k}^{[i]} = c_{k}^{-1} \omega_{k|k-1}^{[i]} p_{v_{k}}^{T}(y_{k} - h_{k}(\bar{\xi}_{k}^{[i]}))
\]

\[
c_{k} = \sum_{i=1}^{N_{k}} \omega_{k|k-1}^{[i]} p_{v_{k}}^{T}(y_{k} - h_{k}(\bar{\xi}_{k}^{[i]})) \quad \text{(Normalization Constant)}
\]

where \( v_{k}^{T} \sim \mathcal{N}(H_{k}(\bar{\xi}_{k}^{[i]}), H_{k}(\bar{\xi}_{k}^{[i]}) P_{k|k-1}^{[i]} H_{k}(\bar{\xi}_{k}^{[i]})^{T} + R_{k}) \).

3. **Measurement Update for Linear State** KF measurement update for the linear part with measurement \( y_{k} - h_{k}(x_{k}^{n}) = H_{k}(x_{k}^{n}) x_{k}^{n} + v_{k} \)

\[
k_{k}^{[i]} = P_{k|k-1}^{[i]} H_{k}(\bar{\xi}_{k}^{[i]})^{T}(H_{k}(\bar{\xi}_{k}^{[i]}) P_{k|k-1}^{[i]} H_{k}(\bar{\xi}_{k}^{[i]})^{T} + R_{k})^{-1}
\]

\[
P_{k|k}^{[i]}(I - K_{k}^{[i]} H_{k}(\bar{\xi}_{k}^{[i]})) P_{k|k-1}^{[i]}
\]

\[
x_{k|k}^{[i]} = x_{k|k-1}^{[i]} + K_{k}^{[i]}(y_{k} - h_{k}(\bar{\xi}_{k}^{[i]})) - H_{k}(\bar{\xi}_{k}^{[i]}) x_{k|k-1}^{[i]})
\]
Algorithm 3 Cont.

4. **Grid Time Propagation for Nonlinear State** Calculate the nonlinear mapped grid set \( H_{k+1}(N_k) = \{ \eta_{k+1}^i; i = 1, \cdots, N_k \} \)

\[ \eta_{k+1}^i = f_k^P(\xi_k^i) \]

5. **Grid Redefinition** Redefine the grid set \( \Xi_{k+1}(N_{k+1}) = \{ \xi_{k+1}^j; j = 1, \cdots, N_{k+1} \} \) from \( H_{k+1}(N_k) \)

6. **Time Propagation for Nonlinear State** Calculate priori masses \( \omega_{k+1}^i[k] \)

\[ \omega_{k+1}^i[k] = \sum_{i=1}^{N_k} \omega_{k+1}^i[k+1] \]

where \( w_i^j \sim \mathcal{N}(F_{k+1}(\xi_k^i), P_{k+1}(\xi_k^i)) \)

7. **Time Propagation for Linear State** KF time propagate for the linear part with artifact measurement \( x_k^p - f_k^P(x_k^p) = z_k \)

\[ z_k = F_k^P(x_k^p) + w_k^p \]

\[ \hat{x}_{k+1}^i[k] = (F_k^P[k] - L_k^T[k]P_{k+1}^l[k])\hat{x}_k^i[k] + f_k^P(Q_k^p) + (Q_{k+1}^l Q_k^p) \]

where \( F_k^P[k] = F_k^P Q_k^p F_k^P[k+1] \) and \( P_{k+1}^l[k] = P_{k+1}^l[k+1] \)

\[ P_{k+1}^l[k] = F_k^P Q_k^p F_k^P[k+1] + Q_{k+1}^l \]

\[ N_{k+1}^l[k] = N_{k+1}^l[k+1] \]

8. **Marginalization for Linear State** Marginalize the linear part PDF for \( x_k^p = \hat{x}_k^i[k] \) by applying moment matching

\[ \alpha_{k+1}^i[k] = \alpha_{k+1}^i[k+1] \]

\[ \hat{x}_{k+1}^i[k] = \sum_{i=1}^{N_k} \alpha_{k+1}^i[k] \hat{x}_{k+1}^i[k] \]

\[ P_{k+1}^l[k] = \sum_{i=1}^{N_k} \alpha_{k+1}^i[k] (\hat{x}_{k+1}^i[k] - \hat{x}_{k+1}^i[k]) (\hat{x}_{k+1}^i[k] - \hat{x}_{k+1}^i[k])^T \]

9. Update \( k := k + 1 \) and repeat (2)–(8)

After performing Step (3) or Step (8), the total mean and variance for the linear part can be acquired as shown in Equations (23) and (24).

\[ \hat{x}_{k}^l[k] = \sum_{i=1}^{N_k} \alpha_{k}^i[k] \hat{x}_{k}^i[k] \]

\[ P_{k}^l[k] = \sum_{i=1}^{N_k} \alpha_{k}^i[k] (\hat{x}_{k}^i[k] - \hat{x}_{k}^i[k]) (\hat{x}_{k}^i[k] - \hat{x}_{k}^i[k])^T \]

4.2. Rao–Blackwellized PMF with MMGK

In the time propagation of the RBPMF, the probability diffusion is calculated by sampling the normal distributions with mean \( F_k^P(z_k^i[k]) \) and variance \( F_k^P(z_k^i[k]) P_{k+1}^l[k] F_k^P(z_k^i[k])^T \).
Algorithm 4: RBPMF with MMGK

1. **Initialization** Same as (1) of Algorithm 3

2. **Measurement Update for Nonlinear State** Same as (2) of Algorithm 3

3. **Measurement Update for Linear State** Same as (3) of Algorithm 3

4. **Grid Propagation for Nonlinear State** Calculate the nonlinear mapped new grid set $H_{k+1}(N_k) = \{\eta_{k+1|i}; i = 1, \cdots, N_k\}$

   $$\eta_{k+1|i} = f^\eta_k(\xi_{k|i}) + F^\eta_k(\xi_{k|i}) \hat{x}_{k|i}$$

5. **Grid Redefinition** Same as (5) of Algorithm 3

6. **Time Propagation for Nonlinear State** Generate the MMGK $\text{Ker}^{[i]}$ with mean $\eta_{k+1|i}$ and variance $F^\eta_k(\xi_{k|i}) F^\eta_k(\xi_{k|i})^T + Q^\eta_k$, and calculate the priori masses as follows

   $$\omega_{k+1|i}^{[j]} = \omega_{k|i}^{[j]} \text{Ker}^{[i]}(j), \text{only for } j \in \text{Ker}^{[i]} \text{ (otherwise 0)}$$

   $$\omega_{k+1|i}^{[j]} = \sum_{j=1}^{N_k} \omega_{k+1|i}^{[j]}$$

   where $j \in \text{Ker}^{[i]}$ means that $\xi_{k+1|i}$ lies within the effective support range of $\text{Ker}^{[i]}$.

7. **Time Propagation for Linear State** Same as (7) of Algorithm 3

8. **Marginalization for Linear State** Same as (8) of Algorithm 3 except that the summations are conducted only for $\omega_{k+1|i}^{[j]} \neq 0$

9. Update $k := k + 1$ and repeat (2)–(8)
4.3. Rao–Blackwelkized PMF with Indirect Time Propagation for Constant Linear Model Case

Since the RBPMF algorithm for the general nonlinear model has its own covariance matrix for each linear part, it was impossible to apply the indirect time propagation algorithm of Algorithm 2 using one common kernel. However, if \( F_k^l(x_k^p) \), \( F_k^j(x_k^p) \), and \( H_k(x_k^p) \) of the model for the linear part are not functions of the non-linear part but constant, then the linear part covariances are approximately equal as follows. Suppose \( P_{k|k-1}^{i,j} \) are equal for all \( i = 1, \cdots , N_k \). Then, the calculation results for \( P_{k|k-1}^{i,j} \) in Step (3) of Algorithm 4 give the same value by \( H_k(x_k^p) = H_k \). Again, the calculation results for \( P_{k+1|k}^{i,j} \) in Step (7) give the same value by \( F_k^l(x_k^p) = F_k^l \), \( F_k^j(x_k^p) = F_k^j \). The last covariance operations of the linear part is the adjustment by the moment matching in Step (8), as shown in Equation (25).

\[
\begin{align*}
P_{k+1|k}^{i,j} &= \sum_{i=1}^{N_k} a_k^{i,j}[i] (P_{k+1|k}^{i,j} + \langle x_k^{i,j}[i] \rangle - \langle x_k^{i,j}[i] \rangle)T \\
&= P_{k+1|k}^{i,j} + \sum_{i=1}^{N_k} a_k^{i,j}[i] (\langle x_k^{i,j}[i] \rangle - \langle x_k^{i,j}[i] \rangle)T \\ &\quad (25)
\end{align*}
\]

Equation (25) has the same covariance \( P_{k+1|k}^{i,j} \) term and the covariance adjustment terms by \( N_k \) normal distributions scattered from mean \( \xi_k^{i,j} \). Since the covariance adjustment terms act in the direction of making the covariance larger, there is a positive diagonal matrix \( \Lambda_k^{i,j} \) that satisfies \( P_{k+1|k}^{i,j} \leq P_{k+1|k}^{i,j} + \Lambda_k^{i,j} \Lambda_k^{i,j} \) (in the sense of positive semi-definite). Then, it can be set as \( P_{k+1|k}^{i,j} = P_{k+1|k}^{i,j} + \Lambda_k^{i,j} \Lambda_k^{i,j} \). \( \Lambda_k = \max \Lambda_k^{i,j} \) by applying a maximum covariance adjustment so that the linear part covariances have the same value for all \( j = 1, \cdots , N_k+1 \) (in the sense of element-wise). Alternatively, \( \Lambda_k \) can be considered as the tuning parameter of \( P_{k+1|k} \). Therefore, by applying the maximum covariance adjustment in moment matching together with the \( F_k^i(x_k^p) = F_k^i \), \( F_k^j(x_k^p) = F_k^j \), and \( H_k(x_k^p) = H_k \) conditions, the covariance matrices of the linear part are equal. In other words, only one operation for the covariance matrix is enough, which not only can significantly reduce the computational burden but also makes it possible to apply the indirect time propagation algorithm of Algorithm 2 for the nonlinear parts which was not applicable because the covariance matrices are different for each linear part.

The PMF’s indirect time propagation algorithm includes the mass redefinition procedure for the nonlinear state variables. Therefore, to apply it to RBPMF, the linear part redefinition for the new grid must also be performed (the covariance matrix is common, and only the mean corresponding to the state estimate of the linear part is redefined). Linear part redefinition is to redefine \((\xi_{k+1}^{i,j}, \gamma_{k+1}^{i,j})_k\) \( p(x_k^p | x_{k+1}^p = \xi_{k+1}^{i,j}, Y_k) \) pairs from \((\xi_k^{i,j}, \omega_{k|k})_k \) \( p(x_k^p | x_{k+1}^p = \xi_k^{i,j}, Y_k) \) pairs according to the nonlinear part redefinition procedure. It has been previously described that \( \gamma_{k|k}^{i,j} \) can be calculated following the same equation as the linear combination of \( \xi_{k+1}^{i,j} \), and the same linear combination can be applied to the linear part redefinition. To show this, examining the PDF of the linear part conditioned on \( \xi_{k+1}^{i,j} \), it is as shown in Equation (26).

\[
p(x_k^p | x_{k+1}^p = \xi_{k+1}^{i,j}, Y_k) = \frac{p(x_k^p, x_{k+1}^p = \xi_{k+1}^{i,j}, Y_k)}{p(x_{k+1}^p = \xi_{k+1}^{i,j}, Y_k)} (26)
\]

The denominator of Equation (26) is \( p(x_{k+1}^p = \xi_{k+1}^{i,j}, Y_k) = p(x_{k+1}^p = \xi_{k+1}^{i,j}|Y_k) p(Y_k) = \gamma_{k|k}^{i,j} p(Y_k) \). The numerator can be calculated through an approximation as follows.
\begin{align*}
p(x^l_k, x^n_{k+1} = \hat{s}^{[i]}_{k+1}, Y_k) &= p(x^l_k, x^n_{k+1} = \xi^{[i]}_{k+1} | Y_k) p(Y_k) \\
&= p(x^l_k, x^n_{k+1} = \sum_s \lambda_s \eta^{[s]}_{k+1} | Y_k) p(Y_k) \\
&= \sum_s \lambda_s \eta^{[s]}_{k+1} \sum_s \lambda_s \eta^{[s]}_{k+1} | Y_k) p(Y_k) \\
&= p(x^l_k, x^n_{k+1} = \sum_s \lambda_s \eta^{[s]}_{k+1} | Y_k) p(Y_k) \\
&= \sum_s \lambda_s \eta^{[s]}_{k+1} \sum_s \lambda_s \eta^{[s]}_{k+1} | Y_k) p(Y_k)
\end{align*} \tag{27}

If we assume a fixed \( x^l_k \), then \( x^l_k \) does not affect an approximation of \( x^n_{k+1} \). Therefore, using the linear interpolation coefficients of \( \xi^{[i]}_{k+1} \), Equation (27) can be approximated in the form of the linear interpolation as shown in Equation (28). Figure 8 illustrates the concept of the approximation of the joint probability of the nonlinear part and the linear part by the linear interpolation.

\begin{align*}
p(x^l_k, x^n_{k+1} = \hat{s}^{[i]}_{k+1}, Y_k) &\simeq \sum_s \lambda_s \eta^{[s]}_{k+1} | Y_k) p(Y_k) \\
&= \sum_s \lambda_s \eta^{[s]}_{k+1} \sum_s \lambda_s \eta^{[s]}_{k+1} | Y_k) p(Y_k) \\
&= \sum_s \lambda_s \eta^{[s]}_{k+1} \sum_s \lambda_s \eta^{[s]}_{k+1} | Y_k) p(Y_k) \\
&= \sum_s \lambda_s \eta^{[s]}_{k+1} \sum_s \lambda_s \eta^{[s]}_{k+1} | Y_k) p(Y_k)
\end{align*} \tag{28}

Figure 8. Concept of the approximation of the joint probability of the nonlinear part and the linear part by the linear interpolation in RBPMF.

Substituting the resulting equation of the numerator and denominator into Equation (26), \( p(x^l_k | x^n_{k+1} = \hat{s}^{[i]}_{k+1}, Y_k) \) becomes Equation (29).

\begin{align*}
p(x^l_k | x^n_{k+1} = \hat{s}^{[i]}_{k+1}, Y_k) &= \frac{1}{\sum_s \lambda_s \eta^{[s]}_{k+1} \sum_s \lambda_s \eta^{[s]}_{k+1} \sum_s \lambda_s \eta^{[s]}_{k+1} | Y_k) p(Y_k)} \\
&= \frac{1}{\sum_s \lambda_s \eta^{[s]}_{k+1} \sum_s \lambda_s \eta^{[s]}_{k+1} \sum_s \lambda_s \eta^{[s]}_{k+1} | Y_k) p(Y_k)} \\
&= \frac{1}{\sum_s \lambda_s \eta^{[s]}_{k+1} \sum_s \lambda_s \eta^{[s]}_{k+1} | Y_k) p(Y_k)}
\end{align*} \tag{29}

Therefore, mean \( \hat{x}^{[i]}_{k|k} \) of \( p(x^l_k | x^n_{k+1} = \hat{s}^{[i]}_{k+1}, Y_k) \) can be obtained as Equation (30).

\begin{align*}
\hat{x}^{[i]}_{k|k} &= \frac{1}{\sum_s \lambda_s \eta^{[s]}_{k+1} \sum_s \lambda_s \eta^{[s]}_{k+1} \sum_s \lambda_s \eta^{[s]}_{k+1} | Y_k) p(Y_k)}
\end{align*} \tag{30}
That is, the redefined mean of the linear part can also be calculated as the linear combination of the previous state estimate with the grid redefinition coefficients. If $\omega_{k|k}^{[s]}$ has approximately the same value, Equation (30) can be more simplified as Equation (31) because it is $\sum_{s} \lambda_{s,i} = 1$.

\[ \hat{x}_{k|k}^{(p)} \simeq \sum_{s} \lambda_{s,i} \tilde{x}_{k|k}^{[s]} \]  \hspace{1cm} \text{(31)}

If the errors by the redefinition have to be treated, we can model it as an additional process noise and reflect it in the time propagation of the linear part. Although the estimation performance due to the redefinition is slightly deteriorated, the robustness of the linear part filter against model errors can be improved. Figure 8 shows the concept of the state estimate redefinition of the linear part.

The RBPMF algorithm applying the indirect time propagation scheme to the model where $p_{k}^{l}(x_{k}^{T}) = F_{k}^{l}(x_{k}^{T}) = F_{k}^{l}$ and $H_{k}(x_{k}^{T}) = H_{k}$ is satisfied is as follows.

**Algorithm 5** RBPMF with Indirect Time Propagation for Constant Linear Model

1. Initialization Same as (1) of Algorithm 3 except for the constant linear state covariance $p_{0|0}^{l} = p_{0}^{l}$

2. Measurement Update for Nonlinear State Calculate the measurement updated masses.

\[ \omega_{k|k}^{[s]} = c_{k}^{-1} \omega_{k|k-1}^{[s]} p_{k}^{l} (y_{k} - h_{k} (\tilde{x}_{k}^{[s]})) \]

\[ c_{k} = \sum_{i=1}^{N_{k}} \omega_{k|k-1}^{[i]} p_{k}^{l} (y_{k} - h_{k} (\tilde{x}_{k}^{[i]})) \] (Normalization Constant)

where $v_{k}^{l} \sim \mathcal{N}(H_{k}(\tilde{x}_{k}^{[i]})) \hat{x}_{k|k-1}^{l}, H_{k}(\tilde{x}_{k}^{[i]})) P_{k|k-1}^{l} H_{k}(\tilde{x}_{k}^{[i]})^{T} + R_{k}$

3. Measurement Update for Linear State KF measurement updates for the linear part with measurement $y_{k} - h_{k}(x_{k}^{T}) = H_{k} x_{k}^{T} + v_{k}$

\[ K_{k} = P_{k|k-1}^{l} H_{k}^{T} (H_{k} P_{k|k-1}^{l} H_{k}^{T} + R_{k})^{-1} \]

\[ p_{k|k}^{l} = (I - K_{k} H_{k}) P_{k|k-1}^{l} \]

\[ \hat{x}_{k|k}^{l} = x_{k|k-1}^{l} + K_{k} (y_{k} - h_{k}(\tilde{x}_{k}^{[i]})) - H_{k} x_{k|k-1}^{l} \]

4. Grid Time Propagation for Nonlinear State Calculate the nonlinear mapped grid set $H_{k+1}(N_{k}) = \{ \eta_{k+1}^{[i]}; i = 1, \cdots, N_{k} \}$

\[ \eta_{k+1}^{[i]} = f_{k}^{p}(\tilde{x}_{k}) + F_{k}^{l} \hat{x}_{k|k}^{l} = f_{k}^{p}(\tilde{x}_{k}) \]

5. Grid, Mass, and Linear State Distribution Redefinition Redefine the grid set $\Xi_{k+1}(N_{k+1}) = \{ \xi_{k+1}^{[i]}; j = 1, \cdots, N_{k+1} \}$ from $H_{k+1}(N_{k})$. Calculate the interpolated masses $\gamma_{k|k}^{[s]}$ for $\xi_{k+1}^{[j]}$ and redefine the linear state mean

\[ \omega_{k|k}^{[s]} = p(x_{k+1}^{T}) = \omega_{k+1|k}^{[s]} | y_{k}^{T} \rangle = \omega_{k|k}^{[s]} | \partial f_{k}^{p}(x_{k}^{T}) / \partial x_{k}^{T} \rangle_{x_{k}^{T} = \tilde{x}_{k}^{[s]}} \]

\[ \gamma_{k|k}^{[s]} = \sum_{s} \lambda_{s,i} \omega_{k|k}^{[s]} \]

\[ \hat{x}_{k|k}^{l} = \frac{1}{\sum_{s} \lambda_{s,j} \omega_{k|k}^{[s]} \hat{x}_{k|k}^{l}} \sum_{s} \lambda_{s,j} \omega_{k|k}^{[s]} \hat{x}_{k|k}^{l} \]
Algorithm 5 Cont.

6. **Time Propagation for Nonlinear State** Generate the MMGK $K_{l,m}$ for each axis of the process noise whose variance is $F_{k}^{nl} = F_{k}^{nl} F_{k}^{nlT} + Q_{k}^{nl}$. Calculate the priori masses by applying kernel $K_{l}$, the tensor product of $K_{l,m}$

$$\omega_{k+1|k}^{[j]} = \gamma_{k}^{[j]} K_{l} \text{ for } j \neq 0\text{ (otherwise 0)}$$

where the index $i$ is limited to the range of $K_{l,m}$

7. **Time Propagation for Linear State** KF time propagate for the linear part with artifact measurement $x_{k+1}^{n} - f_{k}^{n}(x_{k}^{n}) = z_{k} = F_{k}^{l} x_{k}^{n} + w_{k}^{l}$

$$x_{k+1|k}^{[j]} = (F_{k}^{l} - L_{1}^{l} Q_{k}^{nlT} F_{k}^{nl}) x_{k+1|k}^{[j]l} + f_{k}^{l}(\hat{z}_{k+1}^{[j]l}) + (Q_{k}^{nlT} Q_{k}^{nl} + L_{1}^{l} N_{k}^{l} L_{1}^{lT}) z_{k}^{[j]}$$

where

$$F_{k}^{l} = F_{k}^{l} - Q_{k}^{nlT} Q_{k}^{nl} - F_{k}^{nl}$$

$$N_{k}^{l} = F_{k}^{nlT} F_{k}^{nl} + Q_{k}^{nl}$$

$$L_{1}^{l} = F_{k}^{nlT} N_{k}^{l}$$

8. **Marginalization for Linear State** Marginalize the linear part PDF for $x_{k}^{n} = \hat{z}_{k+1}^{[j]}$ by applying moment matching

$$\alpha_{k+1|k}^{[j]} = \frac{\omega_{k+1|k}^{[j]} \alpha_{k+1|k}^{[j]l}}{\omega_{k+1|k}^{[j]} \sum_{l=1}^{N_{k}} \alpha_{k+1|k}^{[j]l}}$$

$$x_{k+1|k}^{[j]} = \sum_{l=1}^{N_{k}} \alpha_{k+1|k}^{[j]l} x_{k+1|k}^{[j]l}$$

$$P_{k+1|k}^{[j]} = P_{k+1|k}^{[j]l} + \Gamma_{k} P_{k+1|k}^{[j]l}$$

where $\Gamma_{k}$ is a positive diagonal matrix and the summation is conducted only for $\omega_{k+1|k}^{[j]} \neq 0$

9. Update $k := k + 1$ and repeat (2)–(8)

As a representative example of such a model, there is Terrain Referenced Navigation (TRN) in which the horizontal position error of two dimensions and the altitude error of one dimension are state variables. For this problem, Peng proposed the interpolation technique of the height errors according to changing grid intervals. However, his method is an index-based adaptive method that simply copies neighbor height error intuitively [30], which is not theoretical and also has a disadvantage that it cannot be applied to changes in grid position. On the other hand, the proposed redefinition algorithm of the linear part is more systematic and can be applied to both the grid position and the interval change.

### 4.4. Numerical Examples

#### 4.4.1. Growth Model with Unknown Parameters

The RBPMF algorithms described in Sections 4.1 and 4.2 were applied to the simulation model performed in Section 3.3. Although the same model, parameters $b$ and $d$ are considered as unknown estimation targets. That is, there are three states to be estimated; $x_{k}$, $b$, and $d$. Furthermore, to apply RBPMF, the nonlinear state variable and the linear state
variable are set to $x^a_n = x_k$ and $x^l_k = [b, d]^T$, respectively. Then the model can be rewritten as follows.

$$
x^{n+1}_k = a \cdot x^n_k + \left[ \begin{array}{c} \frac{x^n_k}{1 + x^n_k^2} \\ c \cdot \cos(k) \end{array} \right] + w^n_k
$$

$$
x^{l+1}_k = x^l_k + w^l_k
$$

where the true parameter values are $a = 0.5$, $b = 5$, $c = 8$, and $d = 0.05$ and the initial error, the process noise, and the measurement noise are $x_0 \sim N(5, 2^2)$, $w^n_k \sim N(0, 0.1^2)$, $w^l_k \sim N(0, \text{diag}(0.001^2, 0.001^2))$, and $v_k \sim N(0, 0.1^2)$, respectively. The grid spacing is $\Delta \xi = 0.5$.

The RMS results of the estimation errors through 100 times Monte Carlo simulations up to 100 s are shown in Figure 9. Sub-figures (c) and (d) are enlarged graphs of the $y$-axis in (a) and (b), respectively. The initial transient characteristics of the conventional RBPMF (Algorithm 3) and the proposed RBPMF (Algorithm 4) are almost same. However, after the convergence, the estimation error of the nonlinear state variable of Algorithm 3 has a large fluctuation, whereas the result of Algorithm 4 shows a much smaller fluctuation. Regarding the parameter estimations, first, the estimation errors of the parameter $b$ were approximately 0.7126 and 0.3455 for Algorithm 3 and 4, respectively, and Algorithm 4 showed approximately two times better performance. For the parameter $d$, there is no significant difference in the estimated error magnitude, but the result of Algorithm 3 tends to oscillate at approximately 20 s intervals after 40 s, but no such trend is seen in the result of Algorithm 4.

Figure 9. Comparison of the simulation error RMS of the RBPMFs for the growth model with two unknown parameters (a) Nonlinear state estimation error (b) Parameter estimation errors (c) Magnification of (a) (d) Magnifications of (b).
Considering the four parameters as the estimation target, since \( F_{nlk}(x^n_k) \) in Equation (20) is \( [x^n_k \ x^n_k / (1 + x^n_k^2) \ \cos(k) \ 0] \), the covariance \( F_{nlk}(\xi^i_k)P_{\xi^i_k}F_{nlk}(\xi^i_k)^T \) of the process noise of the nonlinear part due to the linear part has a large value compared to the grid interval depending on \( x^n_k \) values. Therefore, in this paper, only the parameter \( b \) and \( d \) were estimated, and the simulation was performed only for the problem that the linear part influence on the process noise is small. Figure 10 shows the change in kernel length for MMGK creation over time for one of the previous simulations. Except for the initial transient region, the kernel length within the range of \( \xi < |15| \) maintains 3 to 4 for almost all regions, and in this case, the performance is improved by applying MMGK.

![Figure 10. Kernel length profile for the growth model with the two unknown parameters.](image)

4.4.2. Tightly-Coupled INS/TRN Integration

TRN is the most representative application field of PMF. The basic concept of TRN is to find the position by comparing the difference between the absolute altitude of the Inertial Navigation System (INS) output and the relative altitude of the Radio Altimeter (RA) with a terrain elevation database. TERCOM, the first TRN system, is based on a batch processing algorithm that intermittently estimates the position by accumulating measured values for a certain period of time, but in recent years it is gradually developing into a filter-based sequential processing method. Since the measurement model of TRN is the terrain itself and the terrain has very high nonlinearity, it is mandatory to apply a nonlinear filter to the sequential processing TRN algorithm. In general, TRN is integrated with INS and there are three ways to INS/TRN integration as follows.

- No Integration: Single TRN filter structure without any integration;
- Loosely-coupled: Cascaded structure of the INS aiding filter following TRN filter;
- Tightly-coupled: Single filter structure combining TRN filter and INS aiding filter.

In this paper, the proposed RBPMF algorithm is applied to the tightly-coupled method, which is known to have the best performance among the three methods. To apply RBPMF, first, the mathematical model of TRN must be constructed as shown in Equation (20). Here, a 15th order model including all of the position errors, altitude errors, velocity errors, attitude errors, accelerometer bias errors, and gyro bias errors is considered. Among the 15th state variables, only the two horizontal position errors, which are the independent variable of the terrain elevation function, is selected as the nonlinear part state variable,
and the rest is the linear part state variable. Based on this, the tightly-coupled INS/TRN system model is represented as follows.

\[
\begin{align*}
\mathbf{x}_n^k &= I_{2 \times 2} \mathbf{x}_{n-1}^k + I_{2 \times 2} \mathbf{x}_{l-1}^k + \mathbf{w}_{n-1}^k \\
\mathbf{x}_l^k &= F_{l-1} \mathbf{x}_{n-1}^k + \mathbf{w}_{l-1}^k \\
y_k &= h_{DEM}(\mathbf{x}_n^k) - I_{1 \times 2} \mathbf{x}_n^l + \mathbf{v}_k
\end{align*}
\] (33)

A detailed explanation of the INS error model \( F_{l-1} \) is provided in many other pieces of literature, so it is omitted in this paper. Generally, the horizontal position errors are generally expressed as the angular errors of the latitude and the longitude, but in that case, the system model for the nonlinear part is not a unit matrix because it is affected by the earth radii. Therefore, in this paper, the horizontal position errors are considered as a distance error instead of the angle.

The configuration of various conditions for simulation is as follows. First, the PMF grid size is 51 \times 51. Time propagation and measurement update are performed every 1 s. Due to the characteristics of RBPMF, the nonlinear part is changed by the velocity error of the linear part, so the grid redefinition is performed every time propagation. The initial grid interval is set to cover the 3-sigma region of the initial position error of the INS, and the mass is initialized in the form of a normal distribution. Various initial errors and sensor errors are summarized in Table 2. Terrain elevation data with about 30m resolutions is used for simulation. We conducted 50 times Monte Carlo simulation to obtain position error RMS every time. The flight trajectory is assumed to be straight at constant speed for 200 s. Figure 11 shows the ground trajectory on the terrain elevation. The flight altitude is 300m higher than the highest ground altitude below the flight trajectory, and it is assumed that there is no INS altitude errors and vertical velocity error.

![Figure 11. Ground trajectory and terrain elevation for INS/TRN simulation.](image)

**Table 2.** Initial navigation and sensor error configuration for tightly-coupled INS/TRN simulation.

| Error Types            | Error Magnitude (Standard Deviation) |
|------------------------|--------------------------------------|
| Initial Navigation Errors | Positions 50/50/5 m (\(\delta \lambda / \delta \phi / \delta h\)) |
|                        | Velocities 0.3/0.3/0.1 m/sec (\(\delta v_e / \delta v_n / \delta v_u\)) |
|                        | Attitudes 0.1/0.1/1 mrad (\(\phi_e / \phi_n / \phi_u\)) |
| Accelerometer Bias Error | 100 ug |
| Accelerometer White Noise | 10 ug |
| Gyro Bias Error         | 0.005 deg/hr |
| Gyro White Noise        | 0.005 deg/\sqrt{hr} |
| Radar Altimeter Error   | 10 m |
Figure 12 is a graph of the position error of INS/TRN based on three RBPMFs. Up to the first 60 s, the three algorithms show almost the same position error results. However, after 60 s, only the two proposed algorithms show a similar positional error, and after about 80 s, the stable positional errors of 10 m level are shown in a rough terrain region which is suitable for TRN operation. On the other hand, the error of the conventional RBPMF increased up to the 30 m level as the error was not bound and gradually diverged after 80 s.

![Figure 12](image-url)

**Figure 12.** Comparison of the position error RMS of the conventional RBPMF (Algorithm 3), RBPMF with MMGK (Algorithm 4), and RBPMF for constant linear model (Algorithm 5).

Table 3 summarizes the execution times of the three algorithms. It can be seen that the proposed algorithms operate several times faster than the conventional algorithm. In particular, Algorithm 5 is found to be about 5.67 times faster. Algorithm 4 computes the linear covariance as many as the number of grids, whereas Algorithm 5 computes only one covariance. Due to this effect, the execution time of Algorithm 5 decreased by 29.3 ms and 16.5 ms, respectively, in time propagation and measurement update time compared to Algorithm 4. In other words, Algorithm 5 is calculated approximately 1.35 times faster than Algorithm 4.

| Algorithm                             | Propagation | Update | Total | Ratio |
|---------------------------------------|-------------|--------|-------|-------|
| Conventional RBPMF (Algorithm 3)      | 723         | 28.5   | 751.5 | 1     |
| RBPMF with MMGK (Algorithm 4)         | 152.5       | 25.8   | 178.3 | 4.21  |
| RBPMF for Constant Linear Model (Algorithm 5) | 123.2       | 9.3    | 132.5 | 5.67  |

5. Conclusions

In this paper, we proposed the various algorithms that can improve the reliability of the time propagation of PMF. First, we proposed the PMF algorithm that indirectly performs the probability diffusion through the mass redefinition and the dimension extended MMGK, as opposed to the conventional PMF directly sampling the process noise to perform the probability diffusion. The proposed PMF outperforms the conventional one but requires less computation load. To verify the performance of the proposed algorithm, the simulation was performed on the Growth model and body fall problem. The simulation results show that the proposed PMF performance is improved under most conditions and the computational load is reduced by up to 12 times. RBPMF is one remedy to resolve the excessive computational burden of PMF, which increases exponentially as the dimension of the state variable increases. However, the RBPMF based on the conventional PMF has the
same problem in the probability diffusion process. So, as the second result of this paper, we proposed the RBPMF algorithm adopting MMGK but without mass redefinition. The third result is the proposal of the RBPMF algorithm including the redefinition step of the linear part for indirect time propagation in the case of a constant linear model such as TRN. Simulations results for the Growth model with two unknown parameters and the tightly-coupled INS/TRN integration of the 15th order state variable verify that the proposed algorithm shows better performance with less computation than the conventional RBPMF.

When generating the extended MMGK, we ignored correlations between process noises. However, to consider the correlations, higher moments for multivariate normal distribution have to be dealt with. The proposed algorithms perform mass linear interpolation and it requires the nonlinear transformation of the PDF. Its implementation might not be easy if the system dynamics model is complicated and it has multi-solutions for a given target value.

Author Contributions: Conceptualization, C.-K.S.; methodology, C.-K.S.; validation, C.-K.S. and S.J.L.; formal analysis, C.-K.S.; writing—original draft preparation, C.-K.S.; writing—review and editing, S.J.L.; visualization, C.-K.S.; supervision, S.J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

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

Appendix A

First, for the random vector whose PDF is \( p_w(w) = N_w(m_w, P_w) \), the following equation holds.

\[
 p_w(f(x) - Ay)N_y(m_y, P_y) = p_z(f(x))N_y(m, P)
\]  

(A1)

where \( m \) and \( P \) are \( m = P(A^T P_w^{-1} f(x) + P_y^{-1} m_y) \), \( P^{-1} = A^T P_w^{-1} A + P_y^{-1} \), respectively, and the new random variable \( z \) follows the normal distribution whose mean and covariance are \( m_y + Am \) and \( P_w + AP_yA^T \), respectively [36].

The Rao–Blackwellized Bayesian filtering algorithm consists of 6 steps to obtain \( p(x_{k+1}^l|x_k^n, \bar{Y}_k) \) from \( p(x_k^l|x_k^n, \bar{Y}_{k-1}) \).

1. \( p(x_k^l|\bar{Y}_{k-1}) \rightarrow p(x_k^l|\bar{Y}_k) \)

Applying the Bayes' rule to \( p(x_k^l|\bar{Y}_k) \) is as follows.

\[
 p(x_k^l|\bar{Y}_k) = p(x_k^l|y_k, \bar{Y}_{k-1}) = \frac{y_k|x_k^l, \bar{Y}_{k-1}p(x_k^l|\bar{Y}_{k-1})}{p(y_k|\bar{Y}_{k-1})}
\]  

(A2)

The first term of the numerator is likelihood, which can be obtained as follows.

\[
 p(y_k|x_k^n, \bar{Y}_{k-1}) = \int \int p(y_k|x_k^n, x_k^n, v_k, \bar{Y}_{k-1})p(x_k^n|x_k^n, \bar{Y}_{k-1})p(v_k)dv_kdx_k^n
\]

\[
 = \int \int \delta(y_k - h_k(x_k^n) - H_k(x_k^n)x_k^n - v_k)p(x_k^n|x_k^n, \bar{Y}_{k-1})p(v_k)dv_kdx_k^n
\]

\[
 = \int p_v(y_k - h_k(x_k^n) - H_k(x_k^n)x_k^n)p(x_k^n|x_k^n, \bar{Y}_{k-1})dx_k^n
\]  

(A3)

Since \( p_v(v_k) = N_v(0, R_k) \) and \( p(x_k^n|x_k^n, \bar{Y}_{k-1}) = N_{x_k^n}(x_{k|k-1}^n(x_k^n), p_{k|k-1}^n(x_k^n)) \) are also normal distribution, it can be rewritten as follows by Equation (A1).

\[
 p(y_k|x_k^n, \bar{Y}_{k-1}) = \int p_v(y_k - h_k(x_k^n))N_{x_k^n}(m, P)dx_k^n
\]

\[
 = p_v(y_k - h_k(x_k^n))
\]  

(A4)
where \( v' \sim \mathcal{N}(m_{v,k} + H_k(x^n_k)^T \delta_{[1:k-1]}^l, H_k(x^n_k)P_{[k:k-1]}^l H_k(x^n_k)^T + R_k) \) and \( m, P \) are \( m = P(H_k(x^n_k)^T R_k^{-1} (y^n_k - h_k(x^n_k)) + P_{[k:k-1]}^l \delta_{[1:k-1]}^l, P_{[k:k-1]}^{-1} = H_k(x^n_k)^T R_k^{-1} H_k(x^n_k) + P_{[k:k-1]}^{-1} \), respectively.

(2) \( p(x^n_k | x^n_{k-1}, y_{k-1}) \rightarrow p(x^n_k | x^n_{k}, y_{k}) \)

Applying the Bayes’ rule to \( p(x^n_k | x^n_{k}, y_{k}) \) is as follows.

\[
p(x^n_k | x^n_{k}, y_{k}, y_{k-1}) = \frac{p(y_{k} | x^n_k, x^n_{k-1}, y_{k-1}) p(x^n_k | x^n_{k-1}, y_{k-1})}{p(y_k | x^n_k, y_{k-1})}
\]

(A5)

The first term in the numerator is likelihood, which is as follows.

\[
p(y_k | x^n_k, x^n_{k-1}, y_{k-1}) = p_{y_k}(y_k - h_k(x^n_k) - H_k(x^n_k)x^n_k)
\]

(A6)

That is, since it is the likelihood for the linear measurement model \( H_k(x^n_k) \) and the measurement \( y_k - h_k(x^n_k) \) and \( p(x^n_k | x^n_{k-1}, y_{k-1}) \) is the normal distribution, the measurement update for the linear part, which gives \( p(x^n_k | x^n_{k}, y_{k}) = \mathcal{N}_{l_{ik}}(\delta_{[k:k-1]}^l(x^n_k), P_{[k:k-1]}^l(x^n_k)) \), can be performed by applying KF.

(3) \( p(x^n_{k+1} | y_k) \)

\( p(x^n_{k+1} | Y_k) \) is as follows from the Chapman–Kolmogorov equation.

\[
p(x^n_{k+1} | Y_k) = \iint \int p(x^n_{k+1} | x^n_{k}, w^n_k, Y_k) p(x^n_{k} | x^n_{k-1}, w^n_k, Y_{k-1}) \, dw^n_k \, dx^n_{k} \, dx^n_{k-1}
\]

\[
= \iint \int \delta(x^n_{k+1} - f^n_k(x^n_k)) - F^n_k(x^n_k)x^n_k - w^n_k) p(x^n_{k} | x^n_{k-1}, Y_{k-1}) p(w^n_k) \, dw^n_k \, dx^n_{k} \, dx^n_{k-1}
\]

\[
= \iint \int \delta(x^n_{k+1} - f^n_k(x^n_k)) - F^n_k(x^n_k)x^n_k - w^n_k) p(w^n_k) \, p(x^n_{k} | x^n_{k-1}, Y_{k-1}) p(x^n_{k+1} | Y_k) \, dx^n_{k} \, dx^n_{k-1}
\]  

(A7)

Since \( p(w^n_k) = \mathcal{N}_{w^n_k}(m_{w,k}, Q^n_k) \) and \( p(x^n_{k+1} | x^n_{k}, y_{k}) = \mathcal{N}_{l_{ik}}(\delta_{[k:k-1]}^l(x^n_{k}), P_{[k:k-1]}^l(x^n_{k})) \) are also normal distributions, it can be rewritten as follows by Equation (A1).

\[
p(x^n_{k+1} | Y_k) = \iint p_{w^n_k}(x^n_{k+1} - f^n_k(x^n_k)) \mathcal{N}_{l_{ik}}(m, P) p(x^n_{k} | Y_k) \, dx^n_{k} \, dx^n_{k-1}
\]

\[
= \iint p_{w^n_k}(x^n_{k+1} - f^n_k(x^n_k)) \, p(x^n_{k} | Y_k) \, dx^n_{k} \, dx^n_{k-1}
\]

(A8)

where \( w' \sim \mathcal{N}(m_{w,k} + F^n_k(x^n_k) \delta_{[1:k-1]}^l, Q^n_{w,k} + F^n_k(x^n_k)P_{[k:k-1]}^l F^n_k(x^n_k)^T) \) and \( m, P \) are \( m = P(F^n_k(x^n_k)^T Q^{-1}_{k-1} F^n_k(x^n_k)) + P_{[k:k-1]}^{-1} \), respectively.

(4) \( p(x^n_k | x^n_{k-1}, y_{k-1}) \)

Applying Bayes’ rule to \( p(x^n_k | x^n_{k-1}, y_{k-1}) \) is as follows.

\[
p(x^n_k | x^n_{k-1}, y_{k-1}) = \frac{p(x^n_{k+1} | x^n_k, x^n_{k-1}, y_{k-1}) p(x^n_k | x^n_{k-1}, y_{k-1})}{p(x^n_{k+1} | x^n_{k}, y_{k})}
\]  

(A9)
The first term in the numerator is likelihood, which is as follows.

\[
 p(x^n_{k+1}|x^n_k, Y_k) = \int p(x^n_{k+1}|x^n_k, x^n_{k+1}, Y_k) p(x^n_k|Y_k) dx^n_k
 = p_{w^n_k}(x^n_{k+1} - F^n_k(x^n_k))
\]  

(A10)

That is, since the above equation is the likelihood for linear measurement model \( F^n_k (x^n_k) \) and the measurement \( x^n_{k+1} \) and \( p(x^n_k|Y_k) \) is normal distribution, KF can be applied to perform the artifact measurement update for the linear part.

\[
 (5) \quad p(x^n_k|x^n_{k-1}, Y_k) \rightarrow p(x^n_k|x^n_{k-1}, Y_k) \\
 p(x^n_{k+1}|x^n_k, x^n_{k+1}, Y_k) \text{ is as follows from the Chapman–Kolmogorov equation.}
\]

\[
 p(x^n_{k+1}|x^n_k, x^n_{k+1}, Y_k) = \int p(x^n_{k+1}|x^n_k, x^n_{k+1}, u^n_k, Y_k) p(x^n_k|Y_k) dx^n_k
 = \int p(x^n_{k+1}|x^n_k, x^n_{k+1}, u^n_k) p(x^n_k|Y_k) dx^n_k \int dx^n_k
 = \int p(x^n_{k+1}|x^n_{k+1}, Y_k) \delta(x^n_{k+1} - F^n_k(x^n_k) - u^n_k) p(w^n_k) dx^n_k
 = \int p(x^n_{k+1}|x^n_{k+1}, Y_k) p_{w^n_k}(x^n_{k+1} - F^n_k(x^n_k)) dx^n_k
\]  

(A11)

Considering \( f^n_k(x^n_k) \) as the known input for model \( x^n_k \), since the update equation from \( x^n_k \) to \( x^n_{k+1} \) is the linear model and \( p(x^n_k|x^n_{k-1}, Y_k) \) is normal distribution, the time propagation for the linear state can be performed by applying KF.

\[
 (6) \quad p(x^n_{k+1}|x^n_k, x^n_{k+1}, Y_k) \rightarrow p(x^n_{k+1}|x^n_{k-1}, Y_k)
\]

Marginalization of \( p(x^n_{k+1}|x^n_k, x^n_{k+1}, Y_k) \) for \( x^n_k \) is as follows.

\[
 p(x^n_{k+1}|x^n_k, Y_k) = \int p(x^n_{k+1}|x^n_k, x^n_{k+1}, Y_k) p(x^n_k|Y_k) dx^n_k
 = \int p(x^n_{k+1}|x^n_k, x^n_{k+1}, Y_k) p(x^n_k|Y_k) \frac{p(x^n_{k+1}|x^n_k, Y_k)}{p(x^n_k|Y_k)} dx^n_k
\]  

(A12)

where \( p(x^n_{k+1}|x^n_k, Y_k) \) is the probability value at \( w' = x^n_{k+1} - f^n_k(x^n_k) \) for the new process noise \( w' \) in Step (3). That is, it is \( p(x^n_{k+1}|x^n_k, Y_k) = p_{w'}(x^n_{k+1} - f^n_k(x^n_k)) \), so the above equation can be rewritten as follows.

\[
 p(x^n_{k+1}|x^n_k, Y_k) = \frac{1}{p(x^n_{k+1}|Y_k)} \int p(x^n_{k+1}|x^n_k, x^n_{k+1}, Y_k) p_{w'}(x^n_{k+1} - f^n_k(x^n_k)) p(x^n_k|Y_k) dx^n_k
\]  

(A13)

References

1. Simon, D. Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches; Wiley-Interscience: New York, NY, USA, 2006; pp. 400–409, 451–452, 462–466, 466–469.
2. Gustafsson, F. Statistical Sensor Fusion; Studentlitteratur AB: Sweden, Switzerland, 2010; pp. 133–136, 197–198, 216–221, 233–236, 253–254.
3. Boutayeb, M.; Rafaralahy, H.; Darouach, M. Convergence Analysis of the Extended Kalman Filter Used as an Observer For Nonlinear Deterministic Discrete-time Systems. IEEE Trans. Autom. Control 1997, 42, 581–586. [CrossRef]
4. Barrau, A.; Bonnabel, S. The Invariant Extended Kalman Filter as a Stable Observer. IEEE Trans. Autom. Control 2017, 62, 1797–1812. [CrossRef]
5. Chen, Z. Bayesian Filtering: From Kalman Filters to Particle Filters, and Beyond. Stat. A J. Theor. Appl. Stat. 2003, 182, 1–69.
6. Fang, H.; Tian, N.; Wang, Y.; Zhou, M.; Haile, M.A. Nonlinear Bayesian Estimation: From Kalman Filtering to a Broader Horizon. IEEE/CAA J. Autom. Sin. 2018, 5, 401–417. [CrossRef]
7. Särkkä, S. Bayesian Filtering and Smoothing; Online Edition for Perfonal Use; Cambridge University Press: Cambridge, UK, 2013; pp. 97–99.
8. Gordon, N.J.; Salmond, D.J.; Smith, A.F. Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation. IEEE Proc. F-Radar Signal Process. 1993, 140, 107–113. [CrossRef]
