Multi-sensor particle filtering with multi-step randomly delayed measurements

Yunqi Chen1 | Zhibin Yan2

1 Center for Control Theory and Guidance Technology, Harbin Institute of Technology, Harbin, People’s Republic of China
2 School of Science, Harbin Institute of Technology, Shenzhen, People’s Republic of China

Correspondence
Zhibin Yan, School of Science, Harbin Institute of Technology, Shenzhen 518055, People’s Republic of China.
Email: zbyan@hit.edu.cn

Abstract
This paper develops particle filtering for multi-sensor systems with randomly delayed measurements, where the general case that random delay can be multi-step rather than one-step or two-step is considered. Moreover, different sensors can have different delay steps and delay probabilities. Random delays are assumed to be mutually independent for different sensors and modelled by a separate sequence of random variables obeying discrete distributions. Since random delay leads to the actual measurements being dependent rather than independent given states, and this dependence becomes more complicated with the increase of random delay step, a new formula of the local likelihood density is proposed and then a new weighting scheme is adopted in particle filtering to deal with these difficulties. The proposed method is applied to two examples to testify its effectiveness and superiority.

1 | INTRODUCTION

Classical filtering methods are based on the assumption that measurements are available in real time or with a constant delay. However, in many actual situations, measurements may be randomly delayed [1–4]. A well-known example is the multi-sensor remote-sensing network system, where actual measurements received by data processing centre may suffer from random delay because of the network congestion [5, 6]. The random measurement delay (RMD) phenomenon can lead to actual measurements being disordered, repeated and missing, which imposes a great challenge to obtain a good filtering estimation.

Recently, several Gaussian approximation filtering methods are proposed to deal with RMD. In the one-step RMD case, the unscented Kalman filter, the cubature Kalman filter, and the quadrature Kalman filter are introduced in [7], [8] and [9], respectively. In the multi-step RMD case, the unscented Kalman filter is proposed in [10]. Subsequently, the cubature Kalman filter and the cubature quadrature Kalman filter are presented in [11] and [12], respectively. The main idea of Gaussian approximation filtering methods to handle RMD is state augmentation. Consequently, the dimension of the augmented state increases linearly with the delay step. This indicates that these methods suffer from a computational burden and diverge especially for high-dimensional state space. Moreover, due to the inherent Gaussian assumption, they will yield a poor filtering estimate in strongly non-linear or non-Gaussian cases.

Particle filtering (PF) is a promising alternative to Gaussian approximation filters since it is not subject to the Gaussian assumption and can provide a decent estimate with sufficient samples [13–15]. Thus it is attractive to utilise PF to deal with RMD. PFs are proposed to deal with one-step RMD and multi-step RMD in [16, 17] and [18], respectively. For the multi-sensor system with multi-step RMD, if randomly delays of all sensors occur in a synchronous manner, the PF proposed in [18] can be directly used by augmenting the measurement equation. However, different sensors generally have different delay steps and delay probabilities, that is to say, random delays are asynchronous for different sensors. To tackle this case, a PF is proposed in [19].

All aforementioned PFs assume that the current actual measurement is independent of the previous actual measurements conditioned on the state sequence, which is impractical in the presence of measurement delay and brings a theoretical problem to these PF algorithms (See Remark 1 in Section 4 for details). To solve this problem, an improved PF is developed for the one-step or two-step delay in [20] and [21], respectively, where a new local likelihood density is designed. We find that the calculation
of the local likelihood density requires to take all possible combinations of a segment of actual measurements into consideration, and the number of all possible combinations is a power exponential function of delay step (See Remark 2 in Section 4 for details). When delay step is small, one can enumerate all possible combinations as has been adopted in [20] and [21]. But for multi-step RMD, which is more likely to occur in practical applications, combinations become extremely intricate, and enumeration method is obviously impractical. Hence, the motivation of this paper is to design a particle filtering which can deal with the measurement dependence for multi-step asynchronous RMD in multi-sensor systems.

The remainder of the paper is organised as follows. In Section 2, we present the multi-sensor measurement model with asynchronous random delays. In Section 3, we review the core of the generic PF and the standard multi-sensor PF. In Section 4, we study the dependence of multi-step randomly delayed measurements and then propose a new PF to deal with multi-step RMD. Numerical simulations are shown in Section 5. Section 6 concludes the paper.

2 | PROBLEM FORMULATION

This paper considers the following non-linear dynamic system with multiple independent sensors:

\[
x_{k+1} = f_k(x_k) + v_k
\]

\[
\zeta_{k,m} = h_{k,m}(x_k) + e_{k,m}, \quad m = 1, 2, \ldots, M
\]

where \( x_k \) is the latent state with known initial probability density \( p(x_0) \), \( \zeta_{k,m} \) is the ideal (undelayed) measurement of the \( m \)-th sensor at time step \( k \) \( \in \mathbb{N} \), and \( M \) the number of sensors. The system noise \( \{e_k, k \geq 0\} \) and the measurement noise \( \{e_{k,m}, k \geq 0\}_{m=1}^M \) are white noise processes with arbitrary probability densities \( p(e_k(\cdot)) \) and \( p(e_{k,m}(\cdot)) \), respectively.

Now assume that the ideal measurements \( \{\zeta_{0,m}\}_{m=1}^M \) are always available for use but that for \( k \geq 1 \), the ideal measurements \( \{\zeta_{k,m}\}_{m=1}^M \) may be randomly unavailable. Then the multi-sensor measurement model with asynchronously multi-step random delays can be modelled as

\[
y_{k,m} = \begin{cases} 
\zeta_{k,m}, & k = 0 \\
\zeta_{k-\tau_{k,m},m}, & k \geq 1
\end{cases}, \quad m = 1, 2, \ldots, M
\]

where \( y_{k,m} \) is the actual measurement of the \( m \)-th sensor at time step \( k \), and \( \tau_{k,m} \) denotes the delay step of the \( m \)-th sensor at time step \( k \). Let \( d_m \) be the user-defined maximum possible measurement delay for the \( m \)-th sensor. Then \( \tau_{k,m} \) obeys a \( d_m \)-valued discrete distribution

\[
P(\tau_{k,m} = s) = p_{k,m}(s), \quad s = 0, 1, \ldots, d_m,
\]

where the random event \( \tau_{k,m} = s \) represents that an \( s \)-step measurement delay occurs for the \( m \)-th sensor at time step \( k \), and \( p_{k,m}(\cdot) \) is the \( s \)-step delay probability for the \( m \)-th sensor at time step \( k \). It is assumed that \( \tau_{k,m} \) is independent of \( \tau_{k_j,m_j} \) when \( k_l \neq k_j \), which means that delays occur at different time steps are mutually independent for the same sensor \( m \). In addition, we assume that \( \tau_{k+1,m} \) is independent of \( \tau_{k,m} \) when \( m_1 \neq m_2 \). This indicates that delays between different sensors are mutually independent. We further assume the mutual independence of \( \{\tau_{k,m}, k \geq 0\}_{m=1}^M \) and \( \{\tau_{k,m}, k \geq 1\}_{m=1}^M \).

For notational simplicity, we denote by \( \zeta_k = \{\zeta_{k,1}, \ldots, \zeta_{k,M}\} \) the ideal measurement set of \( M \) sensors, \( y_k = \{y_{k,1}, \ldots, y_{k,M}\} \) the actual measurement set of \( M \) sensors, and \( a_{i,j} \) the set \( \{a_1, a_1+1, \ldots, a_i\} \) of any general sequence \( a_k \).

For the non-linear system (1) with randomly delayed measurements (2), our objective is to design particle filtering to approximate the filtering density \( p(x_k|y_{0:k}) \). To fulfil this goal, we quote two widely used properties in the Bayesian filtering framework as follows [22].

**Property 1.** (Markov property of states)

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

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

where \( A \) and \( B \) are arbitrary subsets of \( \{x_{0:k-2}; \zeta_{0:k-1}\} \) and \( \{x_{k+1:T}; \zeta_{k+1:T}\} \), respectively.

**Property 2.** (Conditional independence of ideal measurements)

\[
p(\zeta_k|x_{0:k}, C) = p(\zeta_k|x_k),
\]

where \( C \) is an arbitrary subset of \( \{x_{0:k-1}; \zeta_{0:k-1}\} \).

3 | MULTI-SENSOR PARTICLE FILTERING WITHOUT RMD

3.1 | Generic particle filtering

In particle filtering context, the joint density \( p(\xi_{0:k}|y_{0:k}) \) is usually approximated by a set of weighted random samples as

\[
p(\xi_{0:k}|y_{0:k}) \approx \sum_{i=1}^N w_k^{(i)} \delta_{\xi_{0:k}}^{(i)}(\xi_{0:k}),
\]

\[
w_k^{(i)} = \frac{w_k^{(i)}}{\sum_{j=1}^N w_k^{(j)}},
\]

where \( \delta(\cdot) \) is the Dirac delta function. The importance weight \( w_k^{(i)} \) is computed recursively by

\[
w_k^{(i)} \propto w_{k-1}^{(i)} \frac{p(y_k^{(i)}|\xi_{0:k-1}^{(i)}; y_{0:k-1}) p(x_k^{(i)}|\xi_{0:k-1}^{(i)}; y_{0:k-1})}{p(y_k^{(i)}|\xi_{0:k-1}^{(i)}; y_{0:k-1})}.
\]
where \( q(x_k | x_{0:k-1}, y_{0:k}) \) is known as the importance density function.

### 3.2 Standard multi-sensor particle filtering

In the multi-sensor case, since sensors are assumed to be independent of each other, the joint likelihood density \( p(y_k | x_{0:k}, y_{0:k-1}) \) can be computed as

\[
p(y_k | x_{0:k}, y_{0:k-1}) = \prod_{m=1}^{M} p(y_{k,m} | x_{0:k}, y_{0:k-1,m}),
\]

where \( p(y_{k,m} | x_{0:k}, y_{0:k-1,m}) \) is called the local likelihood density. In the undelayed case, the actual measurement sequence \( y_{0:k} \) is exactly the ideal measurement sequence \( z_{0:k} \). Then by using Properties 1 and 2, the importance weight (5) is simplified into

\[
w_k = \mathbb{w}_k \propto \prod_{i=1}^{K} \frac{p(y_k | x_k^{(i)}) p(x_k^{(i)} | x_{k-1}^{(i)})}{q(x_k^{(i)} | x_{k-1}^{(i)}, y_k)},
\]

where the joint likelihood density (6) is simplified into

\[
p(y_k | x_k) = \prod_{m=1}^{M} p(y_{k,m} | x_{k}).
\]

Moreover, if we choose \( q(x_k | x_{0:k-1}, y_k) = p(x_k | x_{0:k-1}) \) and conduct resampling at each time step, the standard multi-sensor particle filtering, also known as the central particle filter [23] can be obtained.

### 4 MULTI-SENSOR PARTICLE FILTERING WITH RMD

In the case of RMD, the actual measurements are disordered, repeated and missing in general. This fact leads to that Properties 1 and 2 cannot be directly applied to actual measurements, and resultantly the importance weight (5) cannot be simplified into (7). Hence, when the standard multi-sensor particle filtering is used by roughly ignoring the RMD, it will inevitably yield a poor state estimate.

To develop a new particle filtering which can capture the information of actual measurements correctly, the importance weight (5) is required to be reformulated. More specifically, we need to figure out expressions of \( p(x_k | x_{0:k-1}, y_{0:k}) \) and \( p(y_k | x_{0:k}, y_{0:k-1,m}) \). To this end, we shall give the following two lemmas.

**Lemma 1.** Conditioned on \( x_{k-1} \), \( x_k \) is independent of \( x_{0:k-2} \) and \( y_{0:k-1} \), that is,

\[
p(x_k | x_{0:k-1}, y_{0:k-1}) = p(x_k | x_{k-1}).
\]

**Proof.** Since the actual measurement set \( \{y_0, y_1, \ldots, y_{k-1}\} \) is a subset of the ideal measurement set \( \{z_{0}, z_{1}, \ldots, z_{k-1}\} \), (9) can then be obtained by using Property 1.

**Lemma 2.** The local likelihood density is simplified into

\[
p(y_{k,m} | x_{0:k}, y_{0:k-1,m}) = p(y_{k,m} | x_{k-d_{m,k}}, y_{k-d_{m,k}}).
\]

**Proof.** Since \( y_{k,m} \in \{z_{k-d_{m,k}}, z_{k-d_{m,k}+1}, \ldots, z_{k}\} \), \( y_{k,m} \) does not depend on \( x_{0:k-d_{m,k}} \) and \( y_{0:k-d_{m,k}} \) conditioned on \( x_{k-d_{m,k}} \) by using Properties 1 and 2, respectively.

**Remark 1.** It is assumed in [16–19] that \( y_{k,m} \) does not depend on \( y_{k-d_{m,k}, k-1,m} \). This results in that (10) can be further simplified into \( p(y_{k,m} | x_{k-d_{m,k}}) \). But in the case of RMD, this assumption is impractical because \( y_{k,m} \) and one of \( y_{k-d_{m,k}, k-1,m} \) have a positive probability to be the same ideal measurement. Thus \( y_{k,m} \) is indeed not independent of \( y_{k-d_{m,k}, k-1,m} \).

For the simplified form (10), according to the assumption that elements of \( \{y_{k,m}\}_{m=1}^{M} \) for the fixed \( m \) are mutually independent and also independent of the state sequence and measurement sequence, we have

\[
p(y_{k,m} | x_{k-d_{m,k}}) \propto \int \int \cdots \int \cdots \int \prod_{i=0}^{d_{m}} \mathcal{P}(\tau_{k-l,m}) d\tau_{k-l,m} \times \\
\varphi(j_{k-d_{m,k}}, \ldots, j_{k,m}),
\]

where

\[
\varphi(j_{k-d_{m,k}}, \ldots, j_{k,m}) = p(y_{k,m} | \tau_{k-d_{m},w} = j_{k-d_{m,k}}, \tau_{k-d_{m,k}+1,m} = j_{k-d_{m,k}+1,m}, \ldots, \tau_{k,m} = j_{k,m}, x_{k-d_{m,k}}, y_{k-d_{m,k}}).
\]

**Remark 2.** For different combinations of \( j_{k-d_{m,k}}, \ldots, j_{k,m} \), the function \( \varphi(j_{k-d_{m,k}}, \ldots, j_{k,m}) \) may take different forms. As a result, we have to specify concrete expressions of \( (d_{m}+1)^{d_{m}+1} \) components \( \varphi(j_{k-d_{m,k}}, \ldots, j_{k,m}) \) in order to compute the local likelihood density.

By dividing all possible combinations of \( j_{k-d_{m,k}}, \ldots, j_{k,m} \) into the following \( d_{m}+2 \) cases, we can obtain explicit expressions of \( \varphi(j_{k-d_{m,k}}, \ldots, j_{k,m}) \).
CHEN AND YAN

Hence

\[ \varphi(j_{k,m}, j_{k-1, m}, \ldots, j_{k,m}) = p(y_{k,m} \mid x_{k-1}). \]  

(13)

**Case 2:** \( j_{k,m} = 1 \) and \( j_{k-1,m} \neq 0 \).

In this case, the number of combinations is \( d_m (d_m + 1)^{d_m - 1} \), and the probability of these combinations is \( \beta_{j_{k,m}}^i (1 - \beta_{j_{k,m}}^{i-1}) \). Since \( y_{j_{k,m}} \) takes the quantity \( \mathcal{Z}_{k-1,m} \) and none of \( y_{k-d_k, k-1,m} \) can take the quantity \( \mathcal{Z}_{k,m} \), by using Property 2, we have

\[
p(y_{k,m} \mid x_{k-d_k}) = \frac{p(x_k \mid x_{k-d_k}) p(y_{k,m} \mid x_{k-d_k-1})}{p(x_k \mid x_{k-d_k-1})} = p(y_{k,m} \mid x_{k-1}).
\]

(14)

Hence

\[
\varphi(j_{k-d_m}, j_{k-d_m+1, m}, \ldots, j_{k,m}) = p(y_{k,m} \mid x_{k-1}).
\]

(15)

**Case s (3 ≤ s ≤ d_m + 1):** \( j_{k,m} = s - 1 \), \( j_{k-1,m} \neq s - 2 \), \( j_{k-1,m} \neq 0 \).

In this case, the number of combinations is \( d_m (d_m + 1)^{d_m - 1} \), and the probability of these combinations is \( \beta_{j_{k,m}}^i (1 - \beta_{j_{k,m}}^{i-1}) \). Since \( y_{j_{k,m}} \) takes the quantity \( \mathcal{Z}_{k-1,m} \) and none of \( y_{k-d_k, k-1,m} \) can take the quantity \( \mathcal{Z}_{k-1,m} \), \( j_{k,m} \) does not depend on \( y_{k-d_k, k-1,m} \) conditioned on \( x_{k-d_k} \). By using the Bayes formula and Property 2, we have

\[
p(y_{k,m} \mid x_{k-d_k}) = \frac{p(x_k \mid x_{k-d_k}) p(y_{k,m} \mid x_{k-d_k-1})}{p(x_k \mid x_{k-d_k-1})} = p(y_{k,m} \mid x_{k-1}).
\]

(16)

Now, consider the case that some of \( y_{k-d_k, k-1,m} \) can take the value in \( \mathcal{Z}_{k-1,m} \). For simplicity, we denote by \( \mathcal{Y}_{k,m}^+ \) the part of \( y_{k-d_k, k-1,m} \) which takes values in \( \mathcal{Z}_{k-1,m} \) and \( \mathcal{Y}_{k,m}^- \) the other part of \( y_{k-d_k, k-1,m} \), which does not take values in \( \mathcal{Z}_{k-1,m} \). Then conditional on \( x_{k-d_k}, y_{k,m} \) does not depend on \( y_{k-d_k, k-1,m} \) and \( \mathcal{Y}_{k,m}^- \). By using the Bayes formula, Properties 1 and 2, we have

\[
p(y_{k,m} \mid x_{k-d_k}, \mathcal{Y}_{k,m}^+) \]

\[
= \frac{p(x_k \mid x_{k-d_k}, \mathcal{Y}_{k,m}^+) p(y_{k,m} \mid x_{k-d_k-1})}{p(x_k \mid x_{k-d_k-1})} = p(y_{k,m} \mid x_{k-1}).
\]

(17)

Hence

\[
\varphi(j_{k-d_m}, j_{k-d_m+1, m}, \ldots, j_{k,m}) = p(y_{k,m} \mid x_{k-1}).
\]

(18)

**Case \( d_m + 2 \):** Other combinations of \( j_{k-d_m, m}, \ldots, j_{k,m} \).

In this case, the number of combinations is \( d_m + 1 \), and the probability of these combinations is \( 1 - \sum_{j_{k,m} = 0}^{d_m} C_{j_{k,m}}^i \). Since \( y_{j_{k,m}} \) and at least one of \( y_{k-d_k, k-1} \) take the same value, by using the fact

\[
\bigcup_{j=1}^{d_m} \{ y_{j_{k,m}} = y_{j_{k,m}} \} = \sum_{j=1}^{d_m} C_{j_{k,m}}^i
\]

(19)

we have

\[
\varphi(j_{k-d_m}, j_{k-d_m+1, m}, \ldots, j_{k,m}) = \sum_{j=1}^{d_m} \chi_{C_{j_{k,m}}^i} \delta_{j_{k,m}}(y_{j_{k,m}}),
\]

(20)

where \( \chi_{C_{j_{k,m}}^i} \) denotes the indicator function of set \( C_{j_{k,m}}^i \).

In summary, we obtain the following theorem.

**Theorem 1.** The local likelihood density can be computed as

\[
p(y_{k,m} \mid x_{k-d_k}, y_{k-d_k, k-1,m})
\]

\[
= \sum_{j=1}^{d_m} d_{j_{k,m}}^i p(y_{k,m} \mid x_k - j) + b_{k,m} \sum_{j=1}^{d_m} \chi_{C_{j_{k,m}}^i} \delta_{j_{k,m}}(y_{j_{k,m}}),
\]

(22)

where

\[
d_{j_{k,m}}^i = \begin{cases} p_{k,m}^i, & j = 0, \\ p_{k,m}^i \prod_{j=1}^{j_{k,m}} (1 - p_{k,m}^{j-1}), & 1 \leq j \leq d_m. \end{cases}
\]
where

\[ y_{k,m} = 1 - \sum_{j=0}^{d_{k,m}} a_{k,m}^j \]

\[ p(y_{k,m}|x_{k-j}) = p(y_{k,m} - b_{k,j} | x_{k-j}) \] \(1 \leq j \leq d_{k,m}\).

In fact, the measurement \( y_{k,m} \) can only occur a \( k \)-step delay at most when \( k < d_{k,m} \), that is, \( p_{k,m}^j = 0 \) for \( k < j \leq d_{k,m} \). Thus Equation (22) in Theorem 1 can be simplified by discarding the combinations of \( j \geq d_{k,m}, j < d_{k,m} + 1, \ldots, j < d_{k,m} \) with zero probability.

**Corollary 1.** The local likelihood density is \( p(y_{0,m}|x_0) \) for \( k = 0 \) and for \( 1 \leq k \leq d_{k,m} \).

\[ p(y_{k,m}|x_{k-j}) = \sum_{j=0}^{k-1} a_{k,m}^j p(y_{k,m} | x_{k-j}) + b_{k,m} \sum_{j=1}^{k} \chi_{C_{k,m}}^j \delta_{j,k,j} y_{k,m}, \quad (24) \]

where

\[ b_{k,m} = 1 - \sum_{j=0}^{k-1} a_{k,m}^j \quad (25) \]

In Theorem 1, the dependence of the actual measurements \( y_{k-d_{k,m},k-d_{k,m}} \) is indeed considered. But the local likelihood density given by Theorem 1 has an unusual form because of the presence of Dirac delta functions and indicator functions, which brings a trouble for practical calculation. However, when \( y_k \) and at least one of \( y_{k-d_{k,m},k-d_{k,m}} \) take the same quantity, the second term in (22) is \( \infty \), and resuitantly the local likelihood density is \( \infty \). This means that \( y_{k,m} \) is uninformative about \( x_{k-d_{k,m},k} \) and should be discarded. The above analysis can provide us with a solution for this trouble. Let \( \lambda_{k,m} \) be the index satisfying

\[ \lambda_{k,m} = \begin{cases} 0, & y_{k,m} \notin \{ y_{k-d_{k,m},k-d_{k,m}+1, \ldots, k-1, m} \} \\ 1, & y_{k,m} \in \{ y_{k-d_{k,m},k-d_{k,m}+1, \ldots, k-1, m} \} \end{cases} \quad (26) \]

where \( \tilde{d}_{m} = \min\{k-1, d_{k,m}\} \). Then the local likelihood density can be assigned as

\[ p(y_{k,m}|x_{k-\tilde{d}_{m},k}) = \begin{cases} 1/N, & \lambda_{k,m} = 1 \\ \sum_{j=0}^{\tilde{d}_{m}} a_{k,m}^j p(y_{k,m} | x_{k-j}), & \lambda_{k,m} = 0 \end{cases} \quad (27) \]

**Remark 3.** \( \lambda_{k,m} \) has the ability to distinguish the actual measurements which are repeated. Moreover, the probability that \( y_{k,m} \) is repeated is \( p(\lambda_{k,m} = 1) = a_{k,m} \). Repeated measurements are regarded as valueless and will be discarded in (27).

**TABLE 1** Multi-sensor particle filtering with RMD

\[ \Phi_i, \text{ for } i = 1, 2, \ldots, N \]

- **Sampling:** Draw \( x_0^{(i)} \) from \( p(x_0) \).
- **Weighting:** Compute \( x_0^{(i)} \propto \prod_{m=1}^{M} p(y_{0,m}|x_0^{(i)}) \), and normalise
  \[ \tilde{x}_0^{(i)} = \frac{x_0^{(i)}}{\sum_{j=1}^{N} x_0^{(j)}} \]
- **Resampling:** Draw \( N_0^{(i)} \) from \( \{ x_0^{(j)}, \tilde{x}_0^{(j)} \}_{j=1}^{N} \).

\[ \Phi_i, \text{ for } i = 1, 2, \ldots, N \]

- **Sampling:** Draw \( x_k^{(i)} \) from \( p(x_k|x_{k-1}) \), and augment \( x_k^{(i)} \leftarrow \{ x_k^{(i)}, x_k^{(i)} \} \).
- **Weighting:**
  - Compute \( \tilde{x}_k^{(i)} \) and \( p(y_{k,m}|x_{k-\tilde{d}_{m},k}, y_{k-d_{k,m},k-1,m}) \) by (26) and (27), respectively, where \( \tilde{d}_{m} = \min\{k, d_{k,m}\} \).
  - Compute \( x_k^{(i)} \) by (28), and normalise \( \tilde{x}_k^{(i)} = \frac{x_k^{(i)}}{\sum_{j=1}^{N} x_k^{(j)}} \)
- **Resampling:** Draw \( N_k^{(i)} \) from \( \{ N_0^{(i)}, \tilde{x}_k^{(i)} \}_{j=1}^{N} \).

In this paper, we use the transition density as the proposal density and conduct resampling at each time step. The recursive formula of importance weight (5) is thus simplified into

\[ w_k^{(i)} \propto \prod_{m=1}^{M} p(y_{k,m}|x_{k-\tilde{d}_{m},k}, y_{k-d_{k,m},k-1,m}), \quad k \geq 1. \quad (28) \]

The resulting particle filtering for multiple sensors with RMD is presented in Table 1.

5 | NUMERICAL SIMULATIONS

In this section, the efficiency of the proposed particle filtering is shown by two typical non-linear examples. One is the non-stationary growth model, and the other is the two-dimensional radar tracking model. The following filtering algorithms will be used.

- **CQKFRMD:** Cubature quadratic Kalman filter for RMD proposed in [12].
- **SMSPF ignored:** Standard multi-sensor particle filtering in [23] with RMD being ignored.
- **SMSPF ideal:** Standard multi-sensor particle filtering in [23] using ideal measurements.
- **MSPFRMD:** Multi-sensor particle filtering for RMD in [16, 18].
- **newMSPFRMD:** New multi-sensor particle filtering for RMD in Table 1.

Here, newMSPFRMD is our proposed method, while others are used for comparison. It is worth noting that
SMSPFideal is used only as the benchmark and cannot be used for RMD, because ideal measurements are unavailable actually. We use the following two quantities as performance measures, the root-mean square error (RMSE)

$$\text{RMSE} \triangleq \sqrt{\frac{1}{K} \sum_{k=1}^{K} (x_{t|t}^{k} - x_{t}^{k})^2},$$

and the time-averaged RMSE (TARMSE)

$$\text{TARMSE} \triangleq \frac{1}{T} \sum_{t=0}^{T} \frac{1}{K} \sum_{k=1}^{K} (x_{t|t}^{k} - x_{t}^{k})^2,$$

where $K$ denotes the number of Monte Carlo simulations, $T$ is the simulated time step, $x_{t|t}^{k}$ and $x_{t}^{k}$ represent the true state and the estimated state at time $t$ of the $k$th Monte Carlo run, respectively.

### 5.1 Example 1: Non-stationary growth model

The non-stationary growth model has been widely used as a benchmark problem to testify the performance of non-linear filtering. It is described as

$$x_{k+1} = 0.5x_{k} + 25 \frac{x_{k}}{1 + x_{k}^2} + 8 \cos(1.2k) + v_{k}$$  \hspace{1cm} (29a)

$$z_{k,m} = \frac{x_{k}^2}{20} + \epsilon_{k,m}, \quad m = 1, 2, 3$$  \hspace{1cm} (29b)

where the initial state follows $x_{0} \sim \mathcal{N}(0, 25)$, and noise processes $\{v_{k}, k \geq 0\}$ and $\{\epsilon_{k,m}, k \geq 0\}_{m=1}^{3}$ are mutually independent, white and Gaussian distributed according to $v_{k} \sim \mathcal{N}(0, 10)$ and $\epsilon_{k,m} \sim \mathcal{N}(0, 1)$ for $m = 1, 2, 3$, respectively.

Suppose that the actual measurement $y_{k,1}$ suffers from a two-step random delay, and $y_{k,2}$ and $y_{k,3}$ suffer from a three-step random delay. The concrete delay probabilities are given in Table 2. To make a fair comparison, we conduct $K = 1000$ independent Monte Carlo runs with $T = 500$ time steps. In each Monte Carlo run, $N = 200$ particles are employed for all particle filtering methods, and the initial state are randomly chosen from its initial probability distribution.

We firstly use the single sensor 2 to investigate the performance of all filtering methods. For clarity, we only present the RMSE curves of the first 100 and last 50 time steps for all filtering methods in Figures 1 and 2, respectively. Table 3 displays the TARMSEs and the total running time. Clearly, SMSPFideal performs best in terms of RMSE as it uses ideal measurements. By contrast, SMSPFignored has a very low estimation accuracy. The remaining three methods, CQKFRMD, MSPFRMD and newMSPFRMD, have mechanisms to handle the multi-step RMD. But they show different performances. Both MSPFRMD

---

**FIGURE 1** RMSEs of the first 50 time steps for sensor 2

**FIGURE 2** RMSEs of the last 50 time steps for sensor 2

---

**TABLE 2** Delay probabilities in Example 1

| Time step | Sensor 1 | Sensor 2 | Sensor 3 |
|-----------|----------|----------|----------|
| $k = 1$   | $p_{k,1}^1 = 0.5$, $p_{k,2}^1 = 0.5$, $p_{k,3}^1 = 0.2$ | $p_{k,1}^2 = 0.5$, $p_{k,2}^2 = 0.5$, $p_{k,3}^2 = 0.8$ | $p_{k,1}^3 = 0.3$, $p_{k,2}^3 = 0.32$, $p_{k,3}^3 = 0.2$ |
| $k = 2$   | $p_{k,1}^1 = 0.5$, $p_{k,2}^1 = 0.5$, $p_{k,3}^1 = 0.2$ | $p_{k,1}^2 = 0.3$, $p_{k,2}^2 = 0.25$, $p_{k,3}^2 = 0.32$ | $p_{k,1}^3 = 0.2$, $p_{k,2}^3 = 0.25$, $p_{k,3}^3 = 0.48$ |
| $k \geq 3$| $p_{k,1}^1 = 0.5$, $p_{k,2}^1 = 0.25$, $p_{k,3}^1 = 0.32$, $p_{k,2}^2 = 0.125$, $p_{k,3}^2 = 0.384$, $p_{k,3}^3 = 0.096$ | $p_{k,1}^3 = 0.2$, $p_{k,2}^3 = 0.125$, $p_{k,3}^3 = 0.096$ | $p_{k,1}^3 = 0.2$, $p_{k,2}^3 = 0.125$, $p_{k,3}^3 = 0.096$ |
and newMSPFMRD can deal with RMD well, but the latter performs slightly better and consumes less time than the former. This is because the dependence of measurements is considered, and repeated measurements are discarded. CQKFMRD gives the worst results due to its inherent Gaussian assumption, and it requires the most running time due to state augmentation.

Since the multi-sensor CQKFMRD has not been developed yet, we use four other methods to estimate the state in the multisensor case. The curves of RMSEs for the first 100 and last 50 time steps are plotted in Figures 3 and 4, respectively, and simulation results are summarised in Table 3. Similarly to the single sensor case, SMSPFideal gives the best result, followed by newMSPFRMD. By contrast, SMSPFignored has the worst performance. Even though MSPFRMD costs more running time, its estimation error is bigger than newMSPFRMD.

5.2 | Example 2: Two-dimensional radar tracking

Assume that the target is moving in the $x - y$ plane according to the following dynamics

$$x_{k+1} = \begin{bmatrix} I_2 & \Delta t I_2 \\ 0_2 & I_2 \end{bmatrix} x_k + \begin{bmatrix} \Delta^2/2 \end{bmatrix} I_2 r^*_{k},$$

where $x_k = [p^*_k, v^*_k, r^*_{k}, v^*_{k}]^T$, and $0_2$ and $I_2$ denote the $2 \times 2$ zero matrix and identity matrix, respectively. $p^*_k$ and $v^*_k$ represent the Cartesian coordinates of the target, and $r^*_{k}$ and $v^*_{k}$ represent velocities in the $x$ and $y$ directions, respectively. The sampling interval is set to $\Delta t = 1$ s. The system noise $v_k$ is the Gaussian white noise and follows $v_k \sim N(0_2, I_2)$. The initial probability density of the state is $N(\mu_0, P_0)$, where $\mu_0 = [2000$ m, $2000$ m, $20$ m/s, $20$ m/s]$^T$ and $P_0 = \text{diag}(4$ m$^2$, $4$ m$^2$, $16$ m$^2$/s$^2$, $16$ m$^2$/s$^2$).

The target is observed by two radar stations, and the corresponding measurement equations are given by

$$z_{k,1} = \begin{bmatrix} r_{k,1} \\ \theta_{k,1} \end{bmatrix} = \begin{bmatrix} 0 \\ \theta \end{bmatrix} = \begin{bmatrix} \sqrt{(p^*_k)^2 + (v^*_k)^2} \\ \text{arctan}(p^*_k/v^*_k) \end{bmatrix} + \epsilon_{k,1},$$

$$z_{k,2} = \begin{bmatrix} r_{k,2} \\ \theta_{k,2} \end{bmatrix} = \begin{bmatrix} 0 \\ \theta \end{bmatrix} = \begin{bmatrix} \sqrt{(p^*_k - 200)^2 + (v^*_k)^2} \\ \text{arctan}(p^*_k/(p^*_k - 200)) \end{bmatrix} + \epsilon_{k,2},$$

where $z_{k,m}$ is the ideal measurement at time step $k$ for the $m$th radar, which contains the radial distance $r_{k,m}$ and the bearing $\theta_{k,m}$ for $m = 1, 2$. The measurement noises $\epsilon_{k,1}$ and $\epsilon_{k,2}$ are mutually independent white noise processes and obey $\epsilon_{k,1} \sim N(0_2, R_{k,1})$ and $\epsilon_{k,2} \sim N(0_2, R_{k,2})$, respectively, where $R_{k,1} = R_{k,2} = \text{diag}(100$ m$^2$, $10^{-6}$ rad$^2$/$s^2$).

It is assumed that the measurements from these two radars suffer from asynchronously random delays, and the actual measurement $y_k = [y_{k,1}, y_{k,2}]$ satisfy (2). The maximum possible measurement delays for two sensors are set to $d_1 = d_2 = 3$, and the corresponding delay probabilities are given in Table 4.

The following results are for $K = 500$ Monte Carlo simulations with $T = 500$ time steps and $N = 2000$ particles in each

---

**TABLE 3** TARMSEs and total running time in Example 1

| Methods      | Sensor 2 TARMSEs | Time (s) | Multiple sensors TARMSEs | Time (s) |
|--------------|------------------|----------|--------------------------|----------|
| CQKFMRD      | 11.57            | 510.73   | *                        | *        |
| SMSPFignored | 9.64             | 38.67    | 9.83                     | 53.24    |
| MSPFMRD      | 7.68             | 94.65    | 6.52                     | 180.84   |
| newMSPFMRD   | 7.45             | 82.95    | 6.21                     | 154.50   |
| SMSPFideal   | 4.62             | 38.77    | 4.58                     | 52.45    |
TABLE 4  Delay probabilities in Example 2

| Time step | Sensor 1               | Sensor 2               |
|----------|------------------------|------------------------|
| $k = 1$  | $p_{0,k,1} = 0.7$      | $p_{0,k,2} = 0.5$      |
|          | $p_{1,k,1} = 0.3$, $p_{1,k,2} = 0.5$ |                      |
| $k = 2$  | $p_{0,k,1} = 0.7$      | $p_{0,k,2} = 0.5$      |
|          | $p_{1,k,1} = 0.21$, $p_{1,k,2} = 0.25$, $p_{2,k,1} = 0.09$, $p_{2,k,2} = 0.25$ |                      |
| $k \geq 3$ | $p_{0,k,1} = 0.7$      | $p_{0,k,2} = 0.5$      |
|          | $p_{1,k,1} = 0.1$, $p_{1,k,2} = 0.25$, $p_{2,k,1} = 0.1$, $p_{2,k,2} = 0.125$, $p_{3,k,1} = 0.1$, $p_{3,k,2} = 0.125$ |                      |

FIGURE 5  RMSEs of $p^y_k$

simulation. Figures 5–8 present RMSEs for four state components, and Figures 9 and 10 show position and velocity RMSEs, respectively. Table 5 lists TARMSEs and the total running time. It is clear that SMSPFideal has the smallest estimation error, while SMSPFignored has the largest estimation error. Our proposed method, newMSPFRMD, outperforms MSPFRMD in terms of RMSEs for all four state components, position and velocity. Moreover, it can save almost 15% of the total running time compared with MSPFRMD.

TABLE 5  TARMSEs and total running time in Example 2

| Methods     | $p^x_k$ | $p^y_k$ | $v^x_k$ | $v^y_k$ | $p_k$ | $v_k$ | Time (s) |
|-------------|---------|---------|---------|---------|-------|-------|----------|
| SMSPFignored | 23.31   | 23.63   | 3.25    | 3.27    | 33.20 | 4.61  | 325.83   |
| MSPFRMD     | 11.46   | 11.66   | 2.44    | 2.44    | 16.37 | 3.46  | 632.40   |
| newMSPFRMD  | 9.05    | 9.31    | 2.25    | 2.25    | 13.00 | 3.18  | 542.80   |
| SMSPFideal  | 4.65    | 4.66    | 1.81    | 1.81    | 6.59  | 2.56  | 325.91   |
6 | CONCLUSION

A new multi-sensor particle filtering is proposed to deal with the dependence of multi-step randomly delayed measurements. The simulation results validate the superiority of this method in terms of estimation accuracy and running time. It is very general and can be applied to various cases including: (a) The single sensor with multi-step random delay. (b) Multiple sensors with different delay steps and delay probabilities. (c) Delay probabilities can be time varying for the single sensor or multiple sensors.

ORCID

Yunqi Chen https://orcid.org/0000-0003-4343-862X  
Zhibin Yan https://orcid.org/0000-0001-5551-5202

REFERENCES

1. Ma, J., Sun, S.: Optimal linear estimators for systems with random sensor delays, multiple packet dropouts and uncertain observations. IEEE Trans. Signal Process. 59(11), 5181–5192 (2011)
2. Wang, X., et al.: Gaussian smoothers for nonlinear systems with one-step randomly delayed measurements. IEEE Trans. Automat. Control 58(7), 1828–1835 (2013)
3. Huang, Y., Ni, Y., Zhang, Y.: Latency probability estimation of non-linear systems with one-step randomly delayed measurements. IET Control Theory Appl. 10(7), 843–852 (2016)
4. Zhao, K., Song, S.: Alternative framework of the Gaussian filter for non-linear systems with randomly delayed measurements and correlated noises. IET Sci. Measurement Technol. 12(2), 161–168 (2018)
5. Moayedi, M., Foo, Y.K., Soh, Y.C.: Adaptive Kalman filtering in networked systems with random sensor delays, multiple packet dropouts and missing measurements. IEEE Trans. Signal Process. 58(3), 1577–1588 (2010)
6. Yang, Y., et al.: Gaussian-consensus filter for nonlinear systems with randomly delayed measurements in sensor networks. Inform. Fusion 30, 91–102 (2016)
7. Hermoso-Carazo, A., Linares-Pérez, J.: Extended and unscented filtering algorithms using one-step randomly delayed observations. Appl. Math. Comput. 190(2), 1375–1393 (2007)
8. Wang, X., et al.: Gaussian filter for nonlinear systems with one-step randomly delayed measurements. Automatica 49(4), 976–986 (2013)
9. Singh, A.K., Bhaumik, S., Kumar, K.: Quadrature filters for one-step randomly delayed measurements. Appl. Math. Model. 40(19), 8296–8308 (2016)
10. Hermoso-Carazo, A., Linares-Pérez, J.: Unscented filtering algorithm using two-step randomly delayed observations in nonlinear systems. Appl. Math. Model. 33, 3705–3717 (2009)
11. Zhang, Y., Huang, Y., Zhao, L.: A general framework solution of Gaussian filter with multiple step randomly delayed measurements. Acta Automatica Sinica 41(1), 122–135 (2015)
12. Singh, A.K., Date, P., Bhaumik, S.: A modified Bayesian filter for randomly delayed measurements. IEEE Trans. Automat. Control 62(1), 419–424 (2016)
13. Doucet, A., Godhil, S., Andrieu, C.: On sequential Monte Carlo sampling methods for Bayesian filtering. Stat. Comput. 10(3), 197–208 (2000)
14. Yan, Z., Yuan, Y.: Particle filter based on one-step smoothing with adaptive iteration. IET Signal Process. 11(5), 596–603 (2017)
15. Doucet, A., Johansen, A.M.: A tutorial on particle filtering and smoothing: fifteen years later. In: Crisan, D., Rozovskii, B. (eds.) Oxford Handbook of Nonlinear Filtering, 656–705. Oxford University Press, Oxford (2011)
16. Zhang, Y., et al.: Particle filter with one-step randomly delayed measurements and unknown latency probability. Int. J. Syst. Sci. 47(1), 1–13 (2015)
17. Xu, L., et al.: Particle filtering for networked nonlinear systems subject to random one-step sensor delay and missing measurements. Neurocomputing 275, 2162–2169 (2017)
18. Huang, Y., et al.: Particle filter for nonlinear systems with multiple step randomly delayed measurements. Electron. Lett. 51(23), 1859–1861 (2015)
19. Zuo, J., Zhong, X.: Particle filter for nonlinear systems with multi-sensor asynchronous random delays. J. Syst. Eng. Electron. 28(6), 1064–1071 (2017)
20. Zuo, J., Yan, B., Lian, W.: Forward–backward particle smoother for nonlinear systems with one-step randomly delayed measurement delay. IET Signal Process. 12(7), 836–843 (2018)
21. Zuo, J., Guo, Q., Ling, Z.: Particle filter for estimating multi-sensor systems using one- or two-step delayed measurements. AEU-Int. J. Electron. Commun. 82, 265–271 (2017)
22. Särkkä, S.: Bayesian Filtering and Smoothing (1st edn). Cambridge University Press, Cambridge (2013)
23. Savic, V., Wymeersch, H., Zazo, S.: Belief consensus algorithms for fast distributed target tracking in wireless sensor networks. Signal Process. 95(2), 149–160 (2014)