International Journal of Control, Automation, and Systems 22(6) (2024) 1893-1901
http://dx.doi.org/10.1007/s12555-023-0399-2
ISSN:1598-6446 eISSN:2005-4092
http://www.springer.com/12555

Modified Kalman and Maximum Correntropy Kalman Filters for Systems With Bernoulli Distribution k-step Random Delay and Packet Loss

Zheng Liu, Xinmin Song*, and Min Zhang

Abstract: The simultaneous presence of uncertain data delays and data loss in a network control system complicates the state estimation problem and its solution. This paper redesigns the Kalman filter (KF) algorithm for systems with k-step random delayed data and data loss to improve estimation accuracy. A binary Bernoulli distribution is employed in the modified KF algorithm to model the received data with the knowledge of data delay and loss probabilities. Besides, the distribution of the non-Gaussian noise in the measurement system will degrade the performance of the conventional KF algorithm based on the minimum mean square error. Therefore, the modified KF algorithm is extended to the maximum correntropy Kalman filter (MCKF) algorithm to overcome the effect of non-Gaussian noise. The estimation accuracy of the modified KF and MCKF algorithms are experimentally compared under Gaussian and non-Gaussian noises, respectively. The simulation results demonstrate the excellent estimation performance of the proposed modified KF and MCKF algorithms under Gaussian and non-Gaussian noises, respectively.

Keywords: Bernoulli distribution, data loss, Kalman filter, k-step random data delay, maximum correntropy Kalman filter.

1. INTRODUCTION

Recently, owing to its low cost, simple installation and maintenance, high flexibility and reliability, the network control system has become a popular research subject and has been widely utilized in aerospace [1], unmanned aircraft [2,3], unmanned vehicle driving [4,5], and other fields. However, the data in the mentioned network control systems are typically transmitted in a limited capacity common channel, which may be subject to some uncertain disturbances such as measurement random step data delays and packet loss. The presence of data delays and data loss complicates the state estimation problem and introduces significant estimation bias.

In order to solve the above issues, scholars have proposed some estimation algorithms with timestamp technique and known distribution of packet loss variables, respectively [6-15]. Uribe-Murcia *et al*. developed UFIR, Kalman, and H_∞ filters to solve the state estimation problem in systems containing random delays and packet loss, in which the random delays obey a binary Bernoulli distribution [16]. Hermoso-Carazo *et al*. employed the state enhancement to construct an unscented Kalman filter (KF) for the two-step random delay data [17]. In [18], a generalized filter algorithm with a maximum one-step random delay was designed. These nonlinear estimators have assumed that the system remains Gaussian even under randomly delayed measurements. Moreover, they have established models that can receive measurements multiple times to correlate the received measurements with past measurements. These phenomena restrict the estimator from using the standard Bayesian estimation structure.

Although the above systems assume Gaussian noises, the real noises are usually non-Gaussian, seriously degrading the algorithm's performance. Many extended algorithms for KF have been proposed to reduce the influence of non-Gaussian noise [19-21]. These algorithms improve the estimation accuracy of KF under non-Gaussian noise, but none of them consider the situation where random delays and packet loss occur in the system. In order to effectively deal with data delay and packet loss in non-Gaussian noise, Zhang *et al*. designed a one-step random delayed particle filter algorithm by modifying the importance weights [22]. Subsequently, they extended the one-step random delay to the multi-step random delay [23]. Xu *et al*. developed a delayed linear recursive model and combined it with a maximum correntropy criterion to update the posterior estimates and covariance matrix [24]. They then proposed a robust delayed filtering algorithm for the co-location of autonomous underwater vehicles

Manuscript received June 22, 2023; revised October 2, 2023; accepted December 5, 2023. Recommended by Associate Editor Jun Moon under the direction of Senior Editor Hyun Myung. This work was supported in part by the National Natural Science Foundation of China (61873152 and 61876101).

Zheng Liu, Xinmin Song, and Min Zhang are with the Department of Information Science and Engineering, Shandong Normal University, Jinan 250358, China (e-mails: {lz13256707253, xinminsong}@sina.com, minnie_mzhang@163.com).
* Corresponding author.

[25]. An improved KF to reduce the influence of delayed measurements and non-Gaussian noise has been proposed by Nanda *et al*. [26]. Although the above algorithms only consider random delays or packet loss, both generally coexist in real systems. In [27], a particle filter method was presented that employs a different measurement model, including packet loss and random delay measurements. These works have utilized the sum of the product of likelihood densities employing every particle repeatedly at successive time steps for the entire length of the maximum delay. Furthermore, Chen *et al*. presented a maximum correntropy Kalman filter (MCKF) algorithm to process non-Gaussian noise, and the results demonstrated its excellent performance [28]. Therefore, this paper extends the one-step random delay and packet loss [16] to a random k-step delay and packet loss to derive the modified KF and MCKF.

This paper establish the two-step and the k-step random delay models, respectively. Subsequently, the state space model is modified to remove its delay. Meanwhile, the proposed algorithm replaces the lost data with the predicted measurements. Then, the correlation between the system and measurement noises in the modified model is removed using the Lagrange multiplier method. Finally, the KF and MCKF algorithms are designed for systems with k-step random delay and packet loss. The contributions of this paper are twofold: First, we construct a model with k-step random delay and packet loss, which guarantees that the receiver can receive only one measurement at the current moment, thus avoiding data redundancy. Second, we design the MCKF algorithm with k-step random delay and packet loss for the non-Gaussian noise system. The estimation effect of the MCKF algorithm outperforms that of KF for the non-Gaussian noise system.

The remainder of the current paper is arranged as follows: Section 2 presents one-step, two-step, three-step, and multi-step random delay and packet loss models. Section 3 modifies the state space model to remove the delay and the correlation between the system and measurement noises. The KF and MCKF algorithms for systems with the k-step random delay and packet loss are designed in Section 4. Section 5 presents the simulation results under different noisy systems. The conclusion is given in Section 6.

The following notations are employed throughout this paper: $\mathbb{R}^v$ is the v-dimensional Euclidean space; A^T and A^{-1} denote the transpose and inverse of matrix A; $E\{\cdot\}$ describes the expectation operator; I is the unit matrix of the appropriate dimension; $diag(a_1 \cdots a_n)$ represents the diagonal matrix with diagonal elements $a_1 \cdots a_n$; $|A|$ denotes the absolute value of A; $P\{A\}$ indicates the occurrence probability of A.

2. PROLIMINARIES AND PROBLEM FORMULATION

This paper employs the following state space model

$$x_n = Bx_{n-1} + \omega_n, \tag{1}$$

$$y_n = Cx_n + \nu_n, \tag{2}$$

where the subscript n represents the discrete time index, $x_n \in \mathbb{R}^l$ and $y_n \in \mathbb{R}^M$ describe the state and measurement vectors, respectively. $B \in \mathbb{R}^{l\times l}$ and $C \in \mathbb{R}^{M\times l}$ represent the time-varying state and measurement transfer matrices, respectively. The process noise $\omega_n \in \mathbb{R}^l$ and measurement noise $\nu_n \in \mathbb{R}^M$ are zero-mean and their covariance matrices are $Q_n = E\{\omega_n\omega_n^T\}$ and $R_n = E\{\nu_n\nu_n^T\}$, satisfying $E\{\omega_n\nu_k^T\} = 0$ for all n and k.

Fig. 1 models the k-step random delay and packet loss, where x_n and y_n denote real and the received measurements at the moment t_n , respectively. If x_n arrives between the t_n and t_{n+1} moments, then $y_n = x_n$ and the receiver no longer receives data at the current moment. One can see that $x_{1,0}$ indicates that x_1 can normally arrive, while $x_{1,k}$ indicates that the measurement data x_1 occurs with a k-step data delay. Assume that the receiver simultaneously receives $x_{1,1}$ and $x_{2,0}$, indicating that x_2 arrives normally and x_1 occurs with a one-step delay. Since $x_{1,1}$ and the received $x_{2,0}$ overlap, $x_{1,1}$ is lost at t moment, and y_1 does not receive data.

It is assumed that the k-step random delay and packet loss may occur when transmitting the data over an unreliable communication channel. Besides, the delay and packet loss probabilities are given for each step. The proposed algorithm utilizes the predictive measurement of the current moment to replace the lost data when packet loss occurs. This paper presents the following packet loss model with k-step delay ($k > 2$),

$$\begin{aligned} Y_n = {} & \Lambda_{0,n} y_n \\ & + (1-\Lambda_{0,n})(1-\Lambda_{0,n-1})\Lambda_{1,n}(1-\Lambda_{2,n})y_{n-1} \\ & + (1-\Lambda_{0,n})\sum_{r=2}^{k-1}\Bigg[(1-\Lambda_{0,n-r})\prod_{i=1}^{r}\Lambda_{i,n-r+1} \\ & \times (1-\Lambda_{r+1,n-r+1})\prod_{t=1}^{r-1}[1-(1-\Lambda_{0,n-t}) \end{aligned}$$

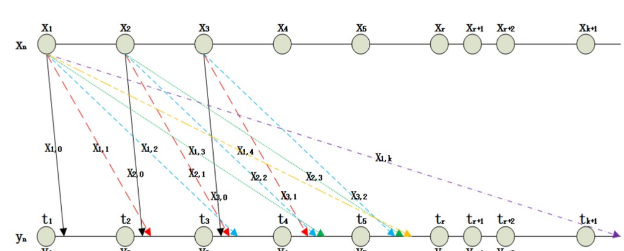


Fig. 1. Modelling of k-step random delay and packet loss in networked control systems.

$$\times \prod_{j=1}^{t} \Lambda_{j,n-t+1} \left(1-\Lambda_{t+1,n-t+1}\right)\Big]\Big] y_{n-r}$$
$$+\left(1-\Lambda_{0,n}\right)\left(1-\Lambda_{0,n-k}\right) \times \prod_{i=1}^{k} \Lambda_{i,n-k+1}$$
$$\times \prod_{t=1}^{k-1}\Big[1-\left(1-\Lambda_{0,n-t}\right)$$
$$\times \prod_{j=1}^{t} \Lambda_{j,n-t+1}\left(1-\Lambda_{t+1,n-t+1}\right)\Big] y_{n-k}$$
$$+\left(1-\Lambda_{0,n}\right)\left[1-\left(1-\Lambda_{0,n-k}\right) \prod_{i=1}^{k} \Lambda_{i,n-k+1}\right]$$
$$\times \prod_{t=1}^{k-1}\Big[1-\left(1-\Lambda_{0,n-t}\right)$$
$$\times \prod_{j=1}^{t} \Lambda_{j,n-t+1}\left(1-\Lambda_{t+1,n-t+1}\right)\Big] \bar{y}_n. \tag{3}$$

Remark 1: Consider that $Y_n \epsilon R^M$ and $\bar{y}_n \epsilon R^M$ represent the received measurement and measurement prediction at the moment t_n. The packet loss model with one-step, two-step and three-step random delay is shown below.

$$\begin{aligned} Y_n = {} & \Lambda_{0,n} y_n + \left(1-\Lambda_{0,n}\right)\left[\left(1-\Lambda_{0,n-1}\right)\Lambda_{1,n} y_{n-1}\right. \\ & \left. +\left(1-\left(1-\Lambda_{0,n-1}\right)\Lambda_{1,n}\right)\bar{y}_n\right], \end{aligned} \tag{4}$$

$$\begin{aligned} Y_n = {} & \Lambda_{0,n} y_n + \left(1-\Lambda_{0,n}\right)\left(1-\Lambda_{0,n-1}\right)\Lambda_{1,n}\left(1-\Lambda_{2,n}\right) \\ & \times y_{n-1} + \left(1-\Lambda_{0,n}\right) \times \left(1-\Lambda_{0,n-2}\right)\Lambda_{2,n-1}\Lambda_{1,n-1} \\ & \times \left(1-\left(1-\Lambda_{0,n-1}\right)\Lambda_{1,n}\left(1-\Lambda_{2,n}\right)\right) y_{n-2} \\ & +\left(1-\Lambda_{0,n}\right) \times \left[1-\left(1-\Lambda_{0,n-2}\right)\Lambda_{2,n-1}\Lambda_{1,n-1}\right] \\ & \times \left(1-\left(1-\Lambda_{0,n-1}\right)\Lambda_{1,n}\left(1-\Lambda_{2,n}\right)\right)\bar{y}_n, \end{aligned} \tag{5}$$

$$\begin{aligned} Y_n = {} & \Lambda_{0,n} y_n + \left(1-\Lambda_{0,n}\right)\left(1-\Lambda_{0,n-1}\right)\Lambda_{1,n}\left(1-\Lambda_{2,n}\right) \\ & \times y_{n-1} + \left(1-\Lambda_{0,n}\right)\left(1-\Lambda_{0,n-2}\right)\Lambda_{2,n-1}\Lambda_{1,n-1} \\ & \times \left(1-\Lambda_{3,n-1}\right)\left(1-\left(1-\Lambda_{0,n-1}\right)\Lambda_{1,n}\left(1-\Lambda_{2,n}\right)\right) \\ & \times y_{n-2} + \left(1-\Lambda_{0,n}\right)\left(1-\Lambda_{0,n-3}\right)\Lambda_{1,n-2}\Lambda_{2,n-2}\Lambda_{3,n-2} \\ & \times \left(1-\left(1-\Lambda_{0,n-1}\right)\Lambda_{1,n}\left(1-\Lambda_{2,n}\right)\right) \\ & \times \left(1-\left(1-\Lambda_{0,n-2}\right)\Lambda_{1,n-1}\Lambda_{2,n-1}\left(1-\Lambda_{3,n-1}\right)\right) \\ & \times y_{n-3} + \left(1-\Lambda_{0,n}\right)\left(1-\left(1-\Lambda_{0,n-2}\right)\Lambda_{2,n-1}\right. \\ & \left. \times \Lambda_{1,n-1}\left(1-\Lambda_{3,n-1}\right)\right) \\ & \times \left(1-\left(1-\Lambda_{0,n-1}\right)\Lambda_{1,n}\left(1-\Lambda_{2,n}\right)\right) \\ & \times \left(1-\left(1-\Lambda_{0,n-3}\right)\Lambda_{1,n-2}\Lambda_{2,n-2}\Lambda_{3,n-2}\right)\bar{y}_n. \end{aligned} \tag{6}$$

We assume that three-step delay does not occur under the three-step random delay and packet loss system, i.e., $\Lambda_{3,n-1} = 0$ and $\Lambda_{3,n-2} = 0$, so (6) degenerates to (5). We further suppose that no two-step delay occurs under the two-step random delay and packet loss system, i.e., $\Lambda_{2,n} = 0$ and $\Lambda_{2,n-1} = 0$, and (5) degenerates to (4). Therefore (3) can be degenerated to (4).

$\Lambda_{0,n}$, $\Lambda_{1,n}$, ..., $\Lambda_{k-1,n}$, and $\Lambda_{k,n}$ are the binary Bernoulli distributed random scalar variables with given probabilities.

Table 1. Some meanings of the symbols related to data delay and packet loss.

Symbols	Meanings
$\overline{\Lambda}_{k,n}$	The probability of $\Lambda_{k,n} = 1$
$\Lambda_{0,n} = 1$	No delay or packet loss at t_n
$\Lambda_{0,n} = 0$	Delay or packet loss at t_n
$\Lambda_{1,n} = 1$	The delay greater than or equal to one-step at t_{n-1}
$\Lambda_{1,n} = 0$	The packet loss at t_{n-1}
$\Lambda_{r,n} = 1\ (r < k)$	The delay greater than or equal to r-step at t_{n-1}
$\Lambda_{r,n} = 0\ (r < k)$	$(r-1)$-step delay at t_{n-1}
$\Lambda_{k,n} = 1$	k-step delay at t_{n-1}
$\Lambda_{k,n} = 0$	$(k-1)$-step delay at t_{n-1}

Table 2. Data received at the receiving end when the system experiences a two-step random delay or packet loss.

n	1	2	3	4	5	6	7	8	9	10
$\Lambda_{0,n}$	1	0	0	0	1	0	0	0	0	0
$\Lambda_{1,n}$	–	–	1	0	1	–	1	1	0	1
$\Lambda_{2,n}$	–	–	1	–	0	–	1	0	–	0
Y_n	y_1	$\bar{y}_2$	$\bar{y}_3$	y_2	y_5	$\bar{y}_6$	$\bar{y}_7$	y_7	$\bar{y}_8$	y_9

$$\begin{aligned} & P\{\Lambda_{0,n}=1\} = \overline{\Lambda}_{0,n},\ P\{\Lambda_{0,n}=0\} = 1-\overline{\Lambda}_{0,n}, \\ & P\{\Lambda_{1,n}=1\} = \overline{\Lambda}_{1,n},\ P\{\Lambda_{1,n}=0\} = 1-\overline{\Lambda}_{1,n}, \\ & \qquad \vdots \\ & P\{\Lambda_{k,n}=1\} = \overline{\Lambda}_{k,n},\ P\{\Lambda_{k,n}=0\} = 1-\overline{\Lambda}_{k,n}. \end{aligned}$$

For convenience, Table 1 summarizes some meanings of the symbols related to data delays and packet loss. If $\Lambda_{0,n-1} = 0$, $\Lambda_{1,n} = 1$, $\Lambda_{2,n} = 0$, and $\Lambda_{0,n} = 0$, the moment t_n has the delay or packet loss and the moment t_{n-1} has a one-step delay. Therefore, the receiver gets y_{n-1} at the moment t_n, and its probability is $(1-\overline{\Lambda}_{0,n})(1-\overline{\Lambda}_{0,n-1})\overline{\Lambda}_{1,n}(1-\overline{\Lambda}_{2,n})$.

Table 2 indicates that y_1 and y_5 can arrive on time, y_7 and y_9 occur with the one-step delay, y_2 occurs with the two-step delay, and the rest of the data are lost, and the measurement prediction $\bar{y}_n$ at the current moment replaces the lost data.

3. REFACTORING THE SYSTEM MODEL

In order to convert the measurement model with delay as the measurement model without delay, the coefficients in (3) are described as

$$\begin{aligned} & A_{0,n} = \Lambda_{0,n}, \\ & A_{1,n} = \left(1-\Lambda_{0,n}\right)\left(1-\Lambda_{0,n-1}\right)\Lambda_{1,n}\left(1-\Lambda_{2,n}\right), \\ & \qquad \vdots \end{aligned}$$

$$A_{r,n} = (1-\Lambda_{0,n})(1-\Lambda_{0,n-r}) \times \prod_{i=1}^{r} \Lambda_{i,n-r+1}(1-\Lambda_{r+1,n-r+1}) \prod_{t=1}^{r-1} [1 - (1-\Lambda_{0,n-t}) \prod_{j=1}^{t} \Lambda_{j,n-t+1}(1-\Lambda_{t+1,n-t+1})],$$

$$\vdots$$

$$A_{k,n} = (1-\Lambda_{0,n})(1-\Lambda_{0,n-k}) \prod_{i=1}^{k} \Lambda_{i,n-k+1} \times \prod_{t=1}^{k-1} [1-(1-\Lambda_{0,n-t}) \times \prod_{j=1}^{t} \Lambda_{j,n-t+1}(1-\Lambda_{t+1,n-t+1})],$$

$$A_{k+1,n} = (1-\Lambda_{0,n}) \left[1-(1-\Lambda_{0,n-k}) \prod_{i=1}^{k} \Lambda_{i,n-k+1}\right] \times \prod_{t=1}^{k-1} [1-(1-\Lambda_{0,n-t}) \times \prod_{j=1}^{t} \Lambda_{j,n-t+1}(1-\Lambda_{t+1,n-t+1})].$$

All coefficients satisfy $\sum_{i=0}^{k+1} A_{i,n} = 1$, where all $A_{i,n}$ has only one 1, and the others are 0. Their expectation and variance can be expressed as

$$\bar{A}_{0,n} = E\{A_{0,n}\} = E\{A_{0,n}^2\},$$
$$\bar{A}_{1,n} = E\{A_{1,n}\} = E\{A_{1,n}^2\},$$
$$\bar{A}_{k,n} = E\{A_{k,n}\} = E\{A_{k,n}^2\},$$
$$\bar{A}_{k+1,n} = E\{A_{k+1,n}\} = E\{A_{k+1,n}^2\}.$$

One can see that $E\{A_{i,n}, A_{j,n}\} = 0$ when $i \neq j$. Based on the above-defined coefficients, the measurement equation (3) can be rewritten as

$$Y_n = A_{0,n} y_n + A_{1,n} y_{n-1} + A_{2,n} y_{n-2} + \cdots + A_{k,n} y_{n-k} + A_{k+1,n} \bar{y}_n. \tag{7}$$

In order to employ x_n to represent y_{n-k}, (1) is recognized to attain the backward time solution [10]

$$x_{n-k} = B^{-k} \left(x_n - \sum_{i=1}^{k-1} B^i \omega_{n-i} \right). \tag{8}$$

A novel delay-free measurement matrix can be obtained by substituting the measurement y_n and the measurement with delays $y_{n-1}, y_{n-2}, ..., y_{n-k}$ into model (8).

$$\begin{aligned} Y_n &= A_{0,n}(Cx_n + \nu_n) + A_{1,n}(Cx_{n-1} + \nu_{n-1}) \\ &\quad + \ldots + A_{k,n}(Cx_{n-k} + \nu_{n-k}) + A_{k+1,n}(CBx_{n-1}) \\ &= A_{0,n}(Cx_n + \nu_n) + A_{1,n}\left(C\left(B^{-1}(x_n - \omega_n)\right) + \nu_{n-1}\right) \\ &\quad + \ldots + A_{k,n}\left(C\left(B^{-k}\left(x_n - \sum_{i=0}^{k-1} B^i \omega_{n-i}\right)\right) + \nu_{n-k}\right) \\ &\quad + A_{k+1,n}\left(CB\left(B^{-1}(x_n - \omega_n)\right)\right) \\ &= \left(A_{k+1,n}C + \sum_{i=0}^{k} A_{i,n} CB^{-i}\right) x_n + \sum_{t=0}^{k} (A_{t,n}\, \nu_{n-t}) \\ &\quad - \sum_{r=1}^{k} \sum_{j=r}^{k} (A_{j,n}\, CB^{-j+r-1}\, \omega_{n-r+1}) - A_{k+1,n} C \omega_n \\ &= \bar{C}_n x_n + \bar{\nu}_n, \end{aligned} \tag{9}$$

where the modified measurement matrix $\bar{C}_n$ and measurement noise $\bar{\nu}_n$ are given as

$$\bar{C}_n = A_{k+1,n} C + \sum_{i=0}^{k} A_{i,n} CB^{-i}, \tag{10}$$

$$\bar{\nu}_n = \sum_{t=0}^{k} (A_{t,n} \nu_{n-t}) - \sum_{r=1}^{k} \sum_{j=r}^{k} \left(A_{j,n} CB^{-j+r-1} \omega_{n-r+1}\right) - A_{k+1,n} C \omega_n. \tag{11}$$

The following simple transformation obtains the measurement noise variance expression.

$$\begin{aligned} \bar{R} &= E\left\{\bar{\nu}_n \bar{\nu}_n^T\right\} \\ &= \sum_{t=0}^{k} \left(\bar{A}_{t,n} R_{n-t}\right) + A_{k+1,n} C Q_n C^T \\ &\quad + \sum_{r=1}^{k} \sum_{j=r}^{k} \left(\bar{A}_{j,n} CB^{-j+r-1}\, Q_{n-r+1} \left(CB^{-j+r-1}\right)^T\right). \end{aligned} \tag{12}$$

Since the modified measurement noise $\bar{\nu}_n$ contains the state noise ω_n, $\bar{\nu}_n$ and ω_n are correlated, and their cross-covariance can be expressed as

$$\begin{aligned} O_n &= E\left\{\bar{\nu}_n \omega_n^T\right\} \\ &= -\bar{A}_{k+1,n} C Q_n - \sum_{i=1}^{k} \bar{A}_{i,n} CB^{-i} Q_n. \end{aligned} \tag{13}$$

Ignoring the correlation between measurement and state noises to modify the algorithm directly complicates the problem when performing the matrix inversion operation. The de-correlation operation is applied between the measurement noise $\bar{\nu}_n$ and the state noise ω_n to avoid errors in modifying the algorithm.

3.1. De-correlation of $\bar{\nu}_n$ and ω_n

Both KF and MCKF are derived under the assumption that the process noise and observation noise are independent of each other. Therefore, if the standard KF and MCKF are used to solve the problem of correlation between process and observation noise, the estimation performance of the filters will be degraded due to the unused information in the noise cross-correlation [29]. To solve the above problem, this paper takes the approach of reconstructing the process equations and the corresponding process noise so that the process and observation noise are no longer correlated.

The Lagrange multiplier method proposed in [30,31] is utilized to rewrite (1) as

$$\begin{aligned} x_n &= Bx_{n-1} + \omega_n + \lambda_n \left(Y_n - \bar{C}_n x_n - \bar{v}_n\right) \\ &= D_n x_{n-1} + U_n + \zeta_n, \end{aligned} \tag{14}$$

where $D_n = B - \lambda_n \bar{C}_n B$, $U_n = \lambda_n Y_n$, and $\zeta_n = \left(I - \lambda_n \bar{C}_n\right)\omega_n - \lambda_n \bar{v}_n$. The new state covariance matrix Q_ζ can be expressed as

$$\begin{aligned} Q_\zeta &= E\left\{\zeta_n \zeta_n^T\right\} \\ &= E\left\{\left[\left(I - \lambda_n \bar{C}_n\right)\omega_n - \lambda_n \bar{v}_n\right][\cdots]^T\right\} \\ &= \left(I - \lambda_n \bar{C}_n\right) Q_n \left(I - \lambda_n \bar{C}_n\right)^T + \lambda_n \bar{R}_n \lambda_n^T \\ &\quad - \left(I - \lambda_n \bar{C}_n\right) O^T \lambda_n^T - \lambda_n O_n \left(I - \lambda_n \bar{C}_n\right)^T. \end{aligned} \tag{15}$$

Lagrange multipliers can be solved by utilizing the uncorrelation between noises $E\left\{\zeta_n \bar{v}_n^T\right\} = 0$

$$\begin{aligned} 0 &= E\left\{\left[\left(I - \lambda_n \bar{C}_n\right)\omega_n - \lambda_n \bar{v}_n\right]\bar{v}_n^T\right\} \\ &= -\left(I - \bar{A}_{0,n}\lambda_n C\right) Q_n \sum_{i=1}^{k}\left(\bar{A}_{i,n}\left(CB^{-i}\right)^T\right) \\ &\quad - Q_n C^T \bar{A}_{k+1,n} - \lambda_n \sum_{t=0}^{k}\left(\bar{A}_{t,n} R_{n-t}\right) \\ &\quad - \lambda_n \sum_{r=2}^{k}\sum_{j=r}^{k}\left(\bar{A}_{j,n}\right. \\ &\quad \left.\times CB^{-j+r-1} Q_{n-r+1}\left(CB^{-j+r-1}\right)^T\right), \end{aligned} \tag{16}$$

$$\begin{aligned} \lambda_n &= -Q_n\left(\sum_{i=1}^{k}\left(\bar{A}_{i,n}\left(CB^{-i}\right)^T\right) + \bar{A}_{k+1,n} C^T\right) \\ &\quad \times \left(\sum_{t=0}^{k}\left(\bar{A}_{t,n} R_{n-t}\right) \sum_{r=2}^{k}\sum_{j=r}^{k}\left(\bar{A}_{j,n} CB^{-j+r-1}\right.\right. \\ &\quad \left.\left.\times Q_{n-r+1}\left(CB^{-j+r-1}\right)^T\right)\right)^{-1}. \end{aligned} \tag{17}$$

Subsequently, the new state space model is obtained by substituting the above Lagrange multiplier into (14), and then the filter algorithm is modified.

4. MODIFICATION OF KF AND MCKF ALGORITHMS

The KF algorithm with k-step delay is designed using (9) and (14).

$$\hat{x}_n^- = D_n \hat{x}_{n-1} + U_n, \tag{18}$$

$$\begin{aligned} P_n^- &= E\left\{\left(x_n - \hat{x}_n^-\right)\left(x_n - x_n^-\right)^T\right\} \\ &= E\left\{\left(D_n x_{n-1} + \zeta_n - D_n \hat{x}_{n-1}\right)(\cdots)^T\right\} \\ &= D P_n D_n^T + Q_\zeta, \end{aligned} \tag{19}$$

$$\hat{x}_n = \hat{x}_n^- + K_n\left(Y_n - \bar{C}_n \hat{x}_n^-\right), \tag{20}$$

$$\begin{aligned} P_n &= E\left\{\left(x_n - \hat{x}_n\right)\left(x_n - \hat{x}_n\right)^T\right\} \\ &= E\left\{\left[\left(x_n - \hat{x}_n^-\right) - K_n\left(\bar{C}_n x_n - \bar{C}_n \hat{x}_n^-\right) - K_n \bar{v}_n\right]\right. \\ &\quad \left.\times \left[\left(x_n - \hat{x}_n^-\right) - K_n\left(\bar{C}_n x_n - \bar{C}_n \hat{x}_n^-\right) - K_n \bar{v}_n\right]^T\right\} \\ &= \left(I - K_n \bar{C}_n\right) P_n^- \left(I - K_n \bar{C}_n\right)^T + K_n \bar{R}_n K_n^T, \end{aligned} \tag{21}$$

$$K_n = P_n^- \bar{C}_n\left(\bar{C}_n P_n^- \bar{C}_n^T + \bar{R}_n\right)^{-1}, \tag{22}$$

$$\bar{y}_n = CB\hat{x}_{n-1}, \tag{23}$$

where (23) denotes the predicted measurement value at the current moment.

Meanwhile, relations (9) and (14) are employed to design the MCKF algorithm with k-step delay, in which its one-step prediction estimation and one-step prediction covariance are equivalent to (19) and (20), respectively. According to (9), we can obtain

$$\begin{bmatrix} \hat{x}_n^- \\ Y_n \end{bmatrix} = \begin{bmatrix} I_n \\ \bar{C}_n \end{bmatrix} x_n + \sigma_n, \tag{24}$$

where $\sigma_n = \begin{bmatrix} \hat{x}_n^- - x_n \\ \bar{v}_n \end{bmatrix}$ with

$$\begin{aligned} E\left\{\sigma_n \sigma_n^T\right\} &= \begin{bmatrix} P_n^- & 0 \\ 0 & \bar{R}_n \end{bmatrix} \\ &= \begin{bmatrix} L_{pn} L_{pn}^T & 0 \\ 0 & L_{rn} L_{rn}^T \end{bmatrix} \\ &= L_n L_n^T. \end{aligned}$$

Therefore, $L_n = \begin{bmatrix} L_{pn} & 0 \\ 0 & L_{rn} \end{bmatrix}$. Left multiplying both sides of (24) by L_n^{-1} gives

$$\alpha^n = \beta^n x_n + e^n, \tag{25}$$

where $\alpha^n = L_n^{-1}\begin{bmatrix} \hat{x}_n^- \\ Y_n \end{bmatrix}$, $\beta^n = L_n^{-1}\begin{bmatrix} I_n \\ \bar{C}_n \end{bmatrix}$, and $e^n = L_n^{-1}\sigma_n$. e_i^n is defined as the i-th element of e^n. Therefore, $\alpha_i^n = \beta_i^n x_n + e_i^n$.

The estimation of $\tilde{x}_n$, gain $\tilde{K}_n$, and covariance matrix $\tilde{P}_n$ of MCKF can be updated as

$$\tilde{x}_n = \hat{x}_n^- + \tilde{K}_n\left(Y_n - \bar{C}_n \hat{x}_n^-\right), \tag{26}$$

$$\tilde{K}_n = \tilde{P}_n^- \bar{C}_n^T\left(\bar{C}_n \tilde{P}_n^- \bar{C}_n^T + \tilde{R}_n\right)^{-1}, \tag{27}$$

$$\tilde{P}_n^- = L_{pn} T_x^{-1} L_{pn}^T, \tag{28}$$

$$\tilde{R}_n = L_{rn} T_y^{-1} L_{rn}^T, \tag{29}$$

$$\tilde{P}_n = \left(I - \tilde{K}_n \bar{C}_n\right) P_n^- \left(I - \tilde{K}_n \bar{C}_n\right)^T + \tilde{K}_n \bar{R}_n \tilde{K}_n^T, \tag{30}$$

where $T_x = \mathrm{diag}(G_\eta(e_1^n), \ldots, G_\eta(e_l^n))$, $T_y = \mathrm{diag}(G_\eta(e_{l+1}^n), \ldots, G_\eta(e_{l+M}^n))$, and $G_\eta(e) = \exp(\frac{-e^2}{2\eta^2})$.

5. EXPERIMENTAL TESTING

In this section a simulation of a vehicle tracking system [16] is provided and two-step and three-step of random delay and packet loss are used to represent k steps of random delay and packet loss. The KF and MCKF algorithms are compared under Gaussian and non-Gaussian noise systems, respectively. Consider the following linear system

$$\begin{bmatrix} x_1(n) \\ x_2(n) \\ x_3(n) \\ x_4(n) \end{bmatrix} = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(n-1) \\ x_2(n-1) \\ x_3(n-1) \\ x_4(n-1) \end{bmatrix} + \begin{bmatrix} \omega_1(n) \\ \omega_2(n) \\ \omega_3(n) \\ \omega_4(n) \end{bmatrix},$$

$$\begin{bmatrix} y_1(n) \\ y_2(n) \end{bmatrix} = \begin{bmatrix} -1 & 0 & -1 & 0 \\ 0 & -1 & 0 & -1 \end{bmatrix} \begin{bmatrix} x_1(n) \\ x_2(n) \\ x_3(n) \\ x_4(n) \end{bmatrix} + \begin{bmatrix} v_1(n) \\ v_2(n) \end{bmatrix},$$

where $x_1(n)$, $x_2(n)$, $x_3(n)$, and $x_4(n)$ indicate the horizontal position, the vertical position, and the horizontal velocity and vertical velocity respectively, and the sampling time $T = 0.3$ s. First, Gaussian noise is considered as the system noise.

$$\omega(n) \sim N(0, 0.001),$$
$$v(n) \sim N(0, 10).$$

Under the Gaussian noise system described above, we conducted experiments with two-step and three-step random data delays and packet loss, respectively. In the two-step random delay and packet loss experiments, we assumed that the probability of delay or packet loss is 0.3, the delay probability is 0.3×0.8, and the two-step delay probability is $0.3 \times 0.8 \times 0.5$. In the three-step random delay experiments, we assumed that the probability of delay or packet loss is 0.3, the delay probability is 0.3×0.8, the one-step delay probability is $0.3 \times 0.8 \times 0.5$, and the three-step delay probability is $0.3 \times 0.8 \times 0.5 \times 0.5$, with the kernel bandwidth of MCKF set to $\eta = 4$. Based on the above parameters, 100 experiments are performed to obtain the following simulation results, summarized in Figs. 2-3 and Tables 3-4. Tables 3-4 indicate that the KF algorithm's estimation performance is generally superior to that of MCKF when the Gaussian system occurs with two-step and three-step random delay and packet loss.

One hundred experiments with different probabilities ($\Lambda = 0.1, 0.2, ..., 0.9, 1$) are conducted for the Gaussian system to investigate the effect of data delays or loss on the estimation performance of KF and MCKF. The results are illustrated in Fig. 4. As presented in Fig. 4, the estimation efficiency of KF and MCKF increases significantly with increasing probabilities, and the estimation performance of KF is consistently superior to that of MCKF.

Second, the measurement noise is modified to a heavy-tailed non-Gaussian one with a mixed Gaussian distribution. The parameter settings are the same as in the Gaussian condition, and the experimental results are presented in Figs. 5-6 and Tables 5-6. One can see that the estimation performance of MCKF outperforms that of KF when

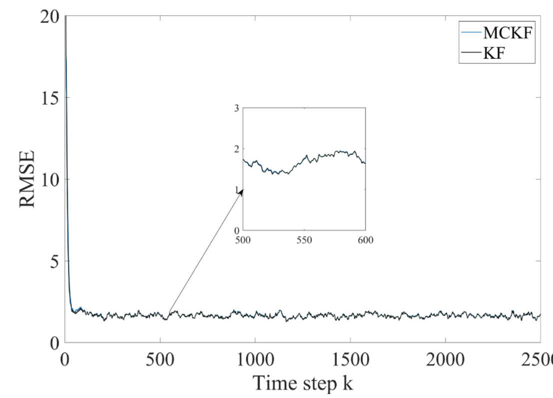


Fig. 2. Average mean squared error of KF and MCKF for two-step random delay and packet loss occurring under Gaussian noise system.

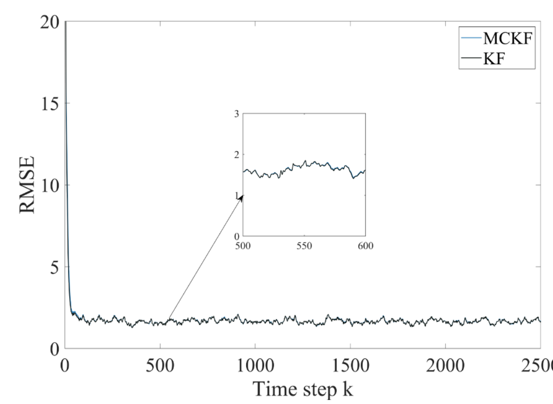


Fig. 3. Average mean squared error of KF and MCKF for three-step random delay and packet loss occurring under Gaussian noise system.

Table 3. Mean square error of KF and MCKF for two-step random delay and packet loss occurring under Gaussian noise system.

Filter	s1	s2	s3	s4
KF	0.7173	0.7116	0.1702	0.1527
MCKF	0.7274	0.7172	0.1731	0.1533

Table 4. Mean square error of KF and MCKF for three-step random delay and packet loss occurring under Gaussian noise system.

Filter	s1	s2	s3	s4
KF	0.7383	0.7179	0.1730	0.1528
MCKF	0.7494	0.7227	0.1758	0.1532

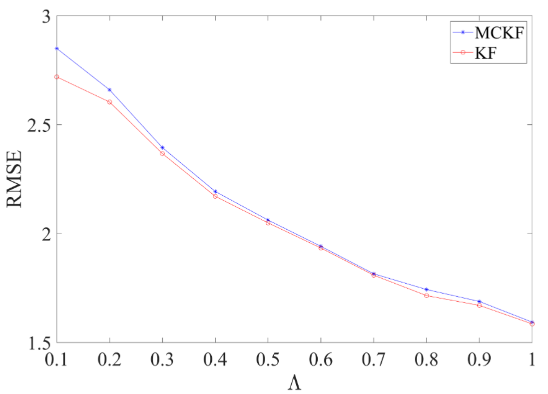


Fig. 4. Impact of the probability of data two-step delay or packet loss on the performance of KF and MCKF estimation under Gaussian noise system.

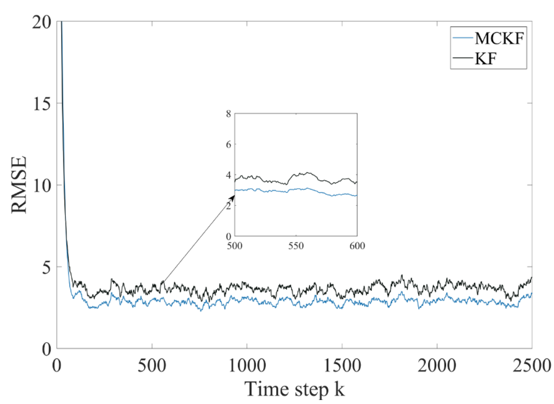


Fig. 5. Average mean squared error of KF and MCKF for two-step random delay and packet loss occurring under non-Gaussian noise system.

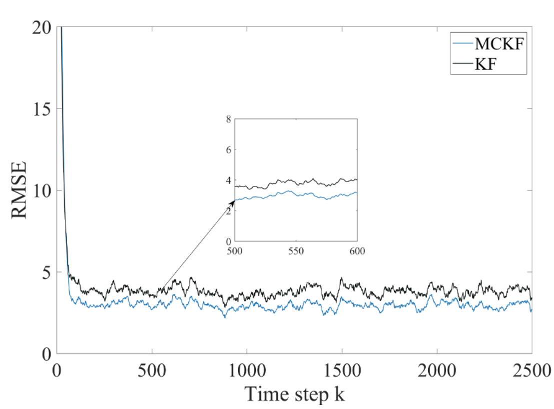


Fig. 6. Average mean squared error of KF and MCKF for three-step random delay and packet loss occurring under non-Gaussian noise system.

two-step and three-step random delay and packet loss occur.

Table 5. Mean square error of KF and MCKF for two-step random delay and packet loss occurring under non-Gaussian noise system.

Filter	s1	s2	s3	s4
KF	1.8345	1.7339	0.2618	0.2190
MCKF	1.4514	1.3500	0.2498	0.2053

Table 6. Mean square error of KF and MCKF for three-step random delay and packet loss occurring under non-Gaussian noise system.

Filter	s1	s2	s3	s4
KF	1.8574	1.7988	0.2681	0.2230
MCKF	1.4915	1.3858	0.2564	0.2087

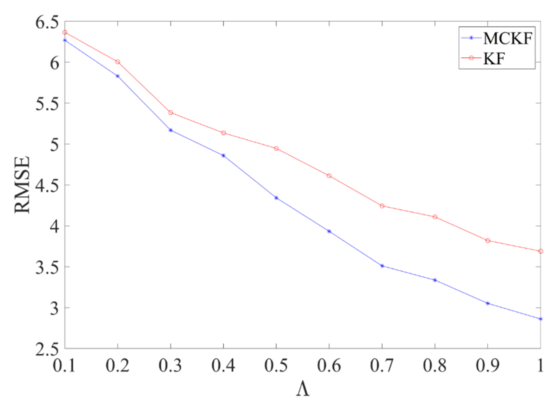


Fig. 7. Impact of the probability of data two-step delay or packet loss on the performance of KF and MCKF estimation under non-Gaussian noise system.

$$\omega(n) \sim N(0, 0.001),$$
$$v(n) \sim 0.9N(0, 0.001) + 0.1N(0, 100).$$

Finally, 100 experiments are performed with different probabilities ($\Lambda = 0.1, 0.2, ..., 0.9, 1$) for the non-Gaussian system, and the results are depicted in Fig. 7. Fig. 7 indicates that the estimation efficiency of KF and MCKF improves significantly with increasing probability under the non-Gaussian noise system, and the estimation performance of MCKF is consistently superior to that of KF. Then, the presented one-step random delay and packet loss MCKF is compared with the KF algorithm proposed by Uribe-Murcia *et al.* [16] under a non-Gaussian noise system, and the results are depicted in Fig. 8. Therefore, as presented in Fig. 8, the modified MCKF algorithm in this paper is considerably superior to the KF algorithm under the non-Gaussian system.

6. CONCLUSION

This paper extends the one-step delay or packet loss to k-step delay or packet loss to render the state estimation

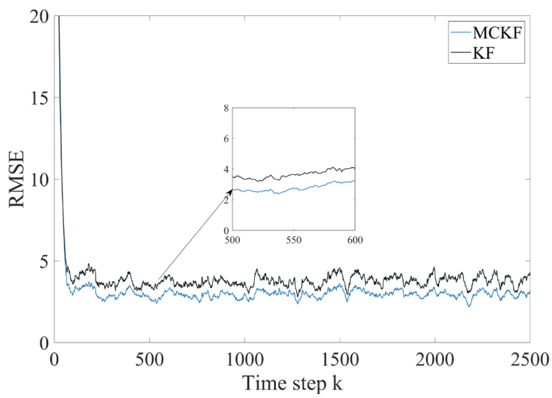


Fig. 8. Mean square error of KF and MCKF for one-step random delay and packet loss occurring under non-Gaussian noise system.

problem more general. Subsequently, KF and MCKF algorithms are developed for systems with k-step random delay or packet loss to achieve better estimation performance and stronger robustness. The state space model is converted to a delay-free form to simplify the complex problem, and the lost data is replaced with the measurement prediction at the current moment. Finally, the estimation performances of the modified KF and MCKF algorithms are compared under different noise systems. The results demonstrate the superiority of the KF algorithm to MCKF in terms of estimation performance in a Gaussian system with delay or packet loss. In contrast, the estimation error of MCKF is smaller than that of KF under the non-Gaussian noise.

CONFLICT OF INTEREST

The authors declare that there is no competing financial interest or personal relationship that could have appeared to influence the work reported in this paper.

REFERENCES

[1] C. Abbondanza, T. M. Chin, R. S. Gross, M. B. Heflin, J. Parker, B. Soja, T. M. Dam, and X. P. Wu, "JTRF2014, the JPL Kalman filter and smoother realization of the international terrestrial reference system," *Journal of Geophysical Research: Solid Earth*, vol. 122, pp. 8474 - 8510, 2017.

[2] H. B. Khamseh, S. Ghorbani, and F. Janabi-Sharifi, "Unscented Kalman filter state estimation for manipulating unmanned aerial vehicles," *Aerospace Science and Technology*, vol. 92, pp. 446-463, 2019.

[3] D. H. Yan, W. G. Zhang, H. Chen, and J. P. Shi, "Robust control strategy for multi-UAVs system using MPC combined with Kalman-consensus filter and disturbance observer," *ISA Transactions*, vol. 135, pp. 35-51, 2023.

[4] V. Pesce, S. Silvestrini, and M. Lavagna, "Radial basis function neural network aided adaptive extended Kalman filter for spacecraft relative navigation," *Aerospace Science and Technology*, vol. 96, 105527, 2020.

[5] Z. Y. Liu, Y. Z. Li, Y. Q. Wu, and S. H. He, "Formation control of nonholonomic unmanned ground vehicles via unscented Kalman filter-based sensor fusion approach," *ISA Transactions*, vol. 125, pp. 60-71, 2022.

[6] Q. Y. Qi, H. S. Zhang, and Z. J. Ji, "Further results on stabilization for NCSs with packet losses and transmission delay: UDP case," *Journal of the Franklin Institute*, vol. 356, no. 8, pp. 4601-4621, 2019.

[7] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. I. Jordan, and S. S. Sastry, "Kalman filtering with intermittent observations," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1453-1464, 2004.

[8] J. J. Bai, R. Q. Lu, H. Y. Su, and A. K. Xue, "Modeling and H_∞ control of wireless networked control system with both delay and packet loss," *Journal of the Franklin Institute*, vol. 352, no. 10, pp. 3915-3928, 2015.

[9] F. L. Qu, Z. H. Guan, D. X. He, and M. Chi, "Event-triggered control for networked control systems with quantization and packet losses," *Journal of the Franklin Institute*, vol. 352, no. 3, pp. 974-986, 2015.

[10] D. R. Ding, Z. D. Wang, B. Shen, and H. L. Dong, "Event-triggered distributed H_∞ state estimation with packet dropouts through sensor networks," *IET Control Theory and Applications*, vol. 9, pp. 1948-1955, August 2015.

[11] A. Mohammadzadeh, B. Tavassoli, and B. Moaveni, "Simultaneous estimation of state and packet-loss occurrences in networked control systems," *ISA Transactions*, vol. 107, pp. 307-315, 2020.

[12] S. Sun, "Optimal linear filters for discrete-time systems with randomly delayed and lost measurements with/without time stamps," *IEEE Transactions on Automatic Control*, vol. 58, no. 6, pp. 1551-1556, 2012.

[13] D. Liu, Z. Wang, Y. Liu, and F. E. Alsaadi, "Extended Kalman filtering subject to random transmission delays: Dealing with packet disorders," *Information Fusion*, vol. 60, pp. 80-86, 2020.

[14] H. Rezaei, R. M. Esfanjani, and M. H. Sedaaghi, "Improved robust finite-horizon Kalman filtering for uncertain networked time-varying systems," *Information Sciences*, vol. 293, pp. 263-274, 2015.

[15] G. Gu, Y. Tang, and F. Qian, "Kalman filtering over the random delay and packet drop channel," *SIAM Journal on Control and Optimization*, vol. 59, no. 4, pp. 2454-2476, 2021.

[16] K. Uribe-Murcia, Y. S. Shmaliy, and J. Andrade-Lucio, "Unbiased FIR, Kalman, and game theory H_∞ filtering under bernoulli distributed random delays and packet dropouts," *Neurocomputing*, vol. 442, pp. 89-97, 2021.

[17] A. Hermoso-Carazo and J. Linares-Pérez, "Unscented filtering algorithm using two-step randomly delayed observations in nonlinear systems," *Applied Mathematical Modelling*, vol. 33, no. 9, pp. 3705-3717, 2009.

[18] X. X. Wang, Y. Liang, Q. Pan, and C. H. Zhao, "Gaussian filter for nonlinear systems with one-step randomly delayed measurements," *Automatica*, vol. 49, no. 4, pp. 976-986, 2013.

[19] X. Fan, G. Wang, J. Han, and Y. Wang, "A background-impulse Kalman filter with non-Gaussian measurement noises," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 53, no. 4, pp. 2434-2443, 2022.

[20] G. Wang, Y. Zhang, and X. Wang, "Iterated maximum correntropy unscented Kalman filters for non-Gaussian systems," *Signal Processing*, vol. 163, pp. 87-94, 2019.

[21] J. He, G. Wang, H. Yu, J. M. Liu, and B. Peng, "Generalized minimum error entropy kalman filter for non-gaussian noise," *ISA Transactions*, vol. 136, pp. 663-675, 2023.

[22] Y. G. Zhang, Y. L. Huang, N. Li, and L. Zhao, "Particle filter with one-step randomly delayed measurements and unknown latency probability," *International Journal of Systems Science*, vol. 47, no. 1, pp. 209-221, 2016.

[23] Y. L. Huang, Y. G. Zhang, N. Li, and L. Zhao, "Particle filter for nonlinear systems with multiple step randomly delayed measurements," *Electronics Letters*, vol. 51, no. 23, pp. 1859-1861, 2015.

[24] B. Xu, X. Wang, J. Zhang, and A. A. Razzaqi, "Maximum correntropy delay Kalman filter for SINS/USBL integrated navigation," *ISA Transactions*, vol. 117, pp. 274-287, 2021.

[25] B. Xu, X. Wang, Y. Guo, J. Zhang, and A. A. Razzaqi, "A novel adaptive filter for cooperative localization under time-varying delay and non-Gaussian noise," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1-15, 2021.

[26] S. K. Nanda, G. Kumar, V. Bhatia, and A. K. Singh, "Kalman filtering with delayed measurements in non-Gaussian environments," *IEEE Access*, vol. 9, pp. 123231-123244, 2021.

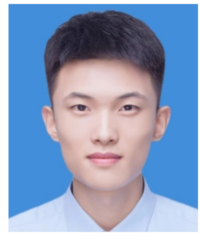
[27] R. K. Tiwari, S. Bhaumik, P. Date, and T. Kirubarajan, "Particle filter for randomly delayed measurements with unknown latency probability," *Sensors (Basel, Switzerland)*, vol. 20, no. 19, e5689, 2020.

[28] B. D. Chen, X. Liu, H. Q. Zhao, and J. C. Principe, "Maximum correntropy Kalman filter," *Automatica*, vol. 76, pp. 70-77, 2017.

[29] G. Chang, "Marginal unscented Kalman filter for cross-correlated process and observation noise at the same epoch," *IET Radar, Sonar & Navigation*, vol. 8, no. 1, pp. 54-64, 2014.

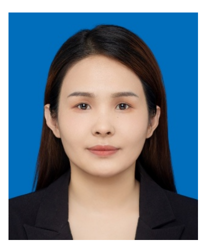
[30] Y. Shmaliy, S. Y. Zhao, and C. K. Ahn, "Kalman and UFIR state estimation with colored measurement noise using backward euler method," *IET Signal Processing*, vol. 14, pp. 64-71, 2020.

[31] T. Kailath, A. H. Sayed, and B. Hassibi, "Linear estimation," *Prentice*, 2000.



Zheng Liu received his B.S. degree in computer science and technology from Shandong University of Science and Technology in 2021. He is currently pursuing an M.S. degree with the School of Information Science and Engineering, Shandong Normal University, Jinan, China. His research interests include delay and packet loss signal estimation, linear quadratic non-Gaussian control, and fusion filtering.

Xinmin Song received his B.S. degree from the Department of Mathematics and an M.S. degree from the Institute of Automation of Qufu Normal University, in 2003 and 2006, respectively, and a Ph.D. degree from the School of Control Science and Engineering of Shandong University in 2010. Since 2010, he has been with the School of Information Science and Engineering, Shandong Normal University, China, where he is currently a professor. From 2014 to 2015, he was a research fellow at Hong Kong University of Science and Technology. From 2015 to 2016, he was a postdoctoral fellow at Yeungnam University. From 2016 to 2017, he was a research fellow at Western Sydney University. His current research interests mainly include time-delay systems, multiplicative noise systems, signal estimation, and information theory learning. In these areas, he has published over 60 journal articles.



Min Zhang received her B.S. degree in computer science and technology from Jining University in 2018, and an M.S. degree in computer software and theory from Shandong Normal University in 2021. She is currently pursuing a Ph.D. degree with the School of Information Science and Engineering, Shandong Normal University, Jinan, China. Her current research interests mainly include emotion recognition, feature extraction, signal estimation, and information theory learning.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.