AIMS Mathematics, 11(4): 8879–8902.
DOI: 10.3934/math.2026366
Received: 23 January 2026
Revised: 19 March 2026
Accepted: 24 March 2026
Published: 01 April 2026

https://www.aimspress.com/journal/Math

Research article

A key conditional quotient filter for nonlinear, non-Gaussian, and non-Markovian systems

Yue Zeng, Yuelin Zhao, Feng Wu* and Li Zhu

Dalian University of Technology, Dalian 116024, China

* **Correspondence:** E-mail: wufeng_chn@163.com.

Abstract: In this paper, we propose a novel efficient key conditional quotient filter (KCQF) for the estimation of state in the nonlinear system, which can be either Gaussian or non-Gaussian, and either Markovian or non-Markovian. The core idea of the proposed KCQF is that only the key measurement conditions, rather than all measurement conditions, should be used to estimate the state. Based on key measurement conditions, the quotient-form analytical integral expressions for the conditional probability density function, mean, and variance of state were derived using the principle of probability conservation, and were calculated using the Monte Carlo method, which thereby constructed the KCQF. Three numerical examples were given to demonstrate the superior estimation accuracy of KCQF, compared to ten filters. The experimental results demonstrated that the KCQF algorithm not only accurately addresses nonlinear problems with high precision but also directly handles navigation issues under time-varying noise conditions.

Keywords: estimation; filtering; nonlinear systems; stochastic systems; key conditional quotient
Mathematics Subject Classification: 93E10

Definition of principal symbols: x: state variable; y: measurement variable; w: process noise; v: measurement noise; $\varphi(\cdot)$: nonlinear state transition function of the system state vector; $\gamma(\cdot)$: measurement function; $p(\cdot)$: probability density function (PDF); $p(\cdot|\cdot)$: conditional probability density function; $\hat{\mathbf{x}}, \mathbf{P}$: mean and covariance; $I(\cdot)$: characteristic function; $\delta(\cdot)$: Dirac function; z: key measurement; β: key measurement error; $\mathbf{A}$: key measurement extraction operator; N_s: number of samples; $r(\cdot,\cdot)$: correlation coefficient; $\mathrm{cov}(\cdot,\cdot)$: covariance; E_{rms}: root-mean-square error; $\overline{E_{\mathrm{rms}}}$: time-averaged root-mean-square error; d: number of key conditions

1. Introduction

Estimating the true state of a system in the presence of random noise is a common problem, particularly prevalent in fields such as control [1], signal processing [2], position estimation [3–5], and mechanics [6–8]. Filtering algorithms are among the commonly employed methods to address this issue. Moreover, filtering problems can be categorized into linear filtering problems and nonlinear filtering problems depending on the models of the system state and the types of noise. A multitude of theories and algorithms for filtering have been developed to date.

The Kalman filter (KF), proposed by Kalman [9], was designed for linear systems driven by Gaussian noise, which can obtain a minimum mean-squared error estimate of the system state [10]. However, most models encountered in the real world are nonlinear. Therefore, to extend the KF to nonlinear systems, the extended Kalman filter (EKF) has gradually been proposed by researchers [11]. The EKF uses the Taylor series expansion to linearize nonlinear functions and then applies the KF for estimation; hence, it requires the system noises to follow Gaussian distributions. It performs well in estimating systems with insignificant nonlinear effects, but can encounter significant errors or divergence when dealing with systems where nonlinear effects are strong. Researchers have proposed various enhanced algorithms based on EKF, such as [12–14]. Nevertheless, the inherent shortcomings of EKF are difficult to overcome. To avoid the estimation error caused by linearization of a nonlinear system, Julier and Uhlman [15] proposed the unscented Kalman filter (UKF) algorithm. The UKF uses a series of discrete points to approximate the posterior probability density function (PDF), offering higher estimation accuracy. However, the UKF also requires that the noises follow Gaussian distributions. Since the proposal of the UKF, a variety of optimization methods for UKF have also emerged, such as [16–18]. The cubature Kalman filter, proposed by Arasaratnam et al. [19], is also a nonlinear filter. This filter uses a third-order spherical-radial rule to approximate the posterior mean and covariance matrix, making it a more suitable nonlinear filter for high-dimensional systems.

The particle filter (PF) [20] originated from the idea of sequential important sampling based on Bayesian sampling estimation. The PF is a Monte Carlo (MC)-based method that uses a set of weighted particles to approximate the posterior PDF of state. MC is a computational approach that solves mathematical problems by generating random samples and performing statistical simulations [21, 22]. The PF is a filter that is well-suited for the systems with strong nonlinearity and non-Gaussian noise. However, the PF suffers from the widely-known problem of particle depletion, wherein a significant fraction of particles loses their weights during update. Although there are some methods to alleviate the particle depletion, such as selecting appropriate importance probability density [23–25] and resampling [26–28], the problem of particle depletion cannot be completely solved.

Apart from particle filter, researchers have proposed a series of robust filtering algorithms to address the state estimation problem for nonlinear non-Gaussian systems with outlier-contaminated measurement noise. The variational Bayesian method (VB) achieves adaptive estimation of noise covariance by approximating the posterior distribution, which can effectively handle the problem of inaccurate measurement noise covariance but has limited performance in dealing with non-Gaussian noise [29–31]. The maximum correntropy Kalman filter adopts the robust maximum correntropy criterion as the optimality criterion instead of the mean square error, which significantly improves the robustness against non-Gaussian heavy-tailed noise and outliers [32, 33]. However, the maximum

correntropy criterion relies on the selection of the kernel bandwidth and is prone to numerical problems under large outliers [31]. In addition, Wei et al. [34] proposed an extended H∞ filtering method based on reproducing kernel Hilbert space (RKHS), which can guarantee the accuracy and robustness of estimation in non-Gaussian noise environments. Tseng et al. [35] presented a robust Huber-based cubature Kalman filter, which integrates the merits of Huber M-estimation and the cubature Kalman filter and provides robustness against deviations from Gaussian behavior. Tan et al. [36] proposed a PI-type exponential-based filter to improve the dynamic performance of the filtering error system.

Each filter has its own characteristics, which implies that different application scenarios might necessitate different filters to achieve optimal outcomes. These filters mostly assume that the state process is a Markovian process; therefore, when dealing with filtering problems where non-Markovian effects are significant, the accuracy will decrease. Constructing an appropriate filter for nonlinear, non-Gaussian, and non-Markovian systems remains an important topic worthy of research. In this paper, we are committed to developing a filter suitable for nonlinear systems by using the conditional probability density function (CPDF) of the system state. We first theoretically observe, without assumptions such as Gaussian distributions and Markov processes, that calculating the CPDF of the state when considering all measurement conditions may suffer from numerical computation instability. Numerical instability arises from the PDF of measurement noise, which is included in the numerator and denominator of the CPDF expression, and tends to approach zero as the iterations proceed. Therefore, when multiple iterative steps are involved, a situation may arise where a small numerator is divided by a small denominator, leading to significant rounding errors. This observation motivated us to come up with a new idea: The system state should be estimated using key measurement conditions instead of all measurement conditions. Centered around this idea, we use the principle of probability conservation to derive quotient-form analytical expressions for the CPDF, mean, and variance of the state based on key measurement conditions. These analytical expressions, based on key conditions, are in integral form and contain only the PDF of a finite number of key measurement noises in the numerator and denominator, which avoids a small numerator divided by a small denominator, thus making numerical computation more stable. We have also defined the reference value based on the correlation coefficient to help extract key measurement conditions, and employed the MC method to numerically solve the integral terms in the expressions.

In summary, here, we abandon the dual assumptions of Gaussian distribution and Markov property. Based on the principle of probability conservation, analytical expressions for the quotient-form conditional probability density function, mean value, and variance are derived, which are theoretically applicable to non-Markovian systems. Moreover, to address the problem of numerical instability when calculating the state conditional probability density function using all measurement conditions, a correlation coefficient is introduced to screen key measurement conditions. Finally, we employ the MC method to numerically solve the integral terms in the expressions, thus developing an effective key conditional quotient filter suitable for nonlinear systems, which can be Gaussian/non-Gaussian and Markovian/non-Markovian.

The remainder of this article is organized as follows: In Section 2, we elaborate on the mathematical model of nonlinear systems; In Section 3, we provide a detailed introduction to the KCQF proposed in this paper; In Section 4, we present three test cases, comparing the proposed KCQF with ten existing excellent filters to demonstrate its superiority in estimation accuracy; and in Section 5,we conclude

with a summary and outlook.

2. Mathematical models

Consider the problem of estimating the state of a nonlinear system taking the form

$$\begin{cases} \mathbf{x}_{k+1} = \varphi_k(\mathbf{x}_k,\ \mathbf{w}_k) \\ \mathbf{y}_{k+1} = \gamma_{k+1}(\mathbf{x}_{k+1}) + \mathbf{v}_{k+1} \end{cases}, \tag{2.1}$$

where $\mathbf{x}_k$ is the system state at time k, $\mathbf{y}_{k+1}$ is the measurement at time $k+1$, φ_k is the nonlinear state transition function, γ_{k+1} is the nonlinear measurement function, and $\mathbf{w}_k$ and $\mathbf{v}_{k+1}$ represent zero mean process noise and measurement noise. The joint PDFs of $\mathbf{w}_{0:k} = (\mathbf{w}_0,\ \mathbf{w}_1,\ \cdots,\ \mathbf{w}_k)$, and $\mathbf{v}_{1:k+1} = (\mathbf{v}_1,\ \mathbf{v}_2,\ \cdots,\ \mathbf{v}_{k+1})$ are denoted by $p_{w_{1:k}}(\mathbf{w}_{1:k})$ and $p_{v_{1:k+1}}(\mathbf{v}_{1:k+1})$, respectively. Here, noise is no longer assumed to be Gaussian white noise. At the initial moment, the initial state $\mathbf{x}_0$ is assumed to follow the distribution $p_{x_0}(\mathbf{x}_0)$. The current question is: How do we estimate $\mathbf{x}_{k+1}$ if a measurement matrix $\mathbf{y}_{1:k+1} = (\mathbf{y}_1,\ \mathbf{y}_2,\ \cdots,\ \mathbf{y}_{k+1})$ composed of measurement vectors at different k has been obtained? For the estimation of $\mathbf{x}_{k+1}$, the commonly used method is to solve the CPDF $p_{x_{k+1}|y_{1:k+1}}(\mathbf{x}_{k+1}|\mathbf{y}_{1:k+1})$. According to the Bayesian method [20], $p_{x_{k+1}|y_{1:k+1}}(\mathbf{x}_{k+1}|\mathbf{y}_{1:k+1})$ can be written as:

$$p_{x_{k+1}|y_{1:k+1}}(\mathbf{x}_{k+1}|\mathbf{y}_{1:k+1}) = \frac{p_{y_{k+1}|x_{k+1}}(\mathbf{y}_{k+1}|\mathbf{x}_{k+1})\, p_{x_{k+1}|y_{1:k}}(\mathbf{x}_{k+1}|\mathbf{y}_{1:k})}{p_{y_{k+1}|y_{1:k}}(\mathbf{y}_{k+1}|\mathbf{y}_{1:k})}. \tag{2.2}$$

If $p_{x_{k+1}|y_{1:k+1}}(\mathbf{x}_{k+1}|\mathbf{y}_{1:k+1})$ is known, then the mean and variance of $\mathbf{x}_{k+1}$ can be expressed as:

$$\hat{\mathbf{x}}_{k+1} = E(\mathbf{x}_{k+1}|\mathbf{y}_{1:k+1}) = \int_{-\infty}^{+\infty} p_{x_{k+1}|y_{1:k+1}}(\mathbf{x}_{k+1}|\mathbf{y}_{1:k+1})\, \mathbf{x}_{k+1} \mathrm{d}\mathbf{x}_{k+1}, \tag{2.3}$$

and

$$\mathbf{P}_{k+1} = \int_{-\infty}^{+\infty} \left[\begin{array}{l} p_{x_{k+1}|y_{1:k+1}}(\mathbf{x}_{k+1}|\mathbf{y}_{1:k+1}) \\ \times (\mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1})(\mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1})^{\mathrm{T}} \end{array} \right] \mathrm{d}\mathbf{x}_{k+1}. \tag{2.4}$$

The $\hat{\mathbf{x}}_{k+1}$ can be used as an estimate of $\mathbf{x}_{k+1}$ when $\mathbf{y}_{1:k+1}$ is known. Calculating (2.3) and (2.4) is challenging because it is often difficult to explicitly express the $p_{x_{k+1}|y_{1:k+1}}(\mathbf{x}_{k+1}|\mathbf{y}_{1:k+1})$. In existing research, KF, EKF, UKF, CKF, and PF can be employed for estimation. However, these methods assume that the state $\mathbf{x}_{k+1}$ is Markovian, and the computational accuracy will decrease when dealing with non-Markovian situations.

In the following section, we have precisely derived expressions for the CPDF $p_{x_{k+1}|y_{1:k+1}}(\mathbf{x}_{k+1}|\mathbf{y}_{1:k+1})$, $\hat{\mathbf{x}}_{k+1}$, and $\mathbf{P}_{k+1}$ without assumptions such as Gaussian distributions and Markov processes, and analyzed that calculating these expressions will lead to numerical instability. To overcome the instability, new quotient-form expressions for the estimation of state based on key measurement conditions have derived, proposing the key conditional quotient filter.

3. Key conditional quotient filter

3.1. A quotient-form expression for conditional probability density function

In this section, we propose a new filter named the key conditional quotient filter (KCQF). To derive this filter, we first present another equivalent form of $p_{x_{k+1}|y_{1:k+1}}(\mathbf{x}_{k+1}|\mathbf{y}_{1:k+1})$ (equivalent to (2.2),

$$p_{x_{k+1}|y_{1:k+1}}(\mathbf{x}_{k+1}|\mathbf{y}_{1:k+1}) = \frac{p_{x_{k+1},y_{1:k+1}}(\mathbf{x}_{k+1},\ \mathbf{y}_{1:k+1})}{p_{y_{1:k+1}}(\mathbf{y}_{1:k+1})}. \tag{3.1}$$

Equation (3.1) can be directly given using the CPDF formula. Using (3.1), a new quotient-form expression for the estimation of $\mathbf{x}_{k+1}$ can be provided. To derive this expression, we first introduce the principle of probability conservation. The principle of probability conservation was initially proposed for the study of random dynamical systems [37]. Taking a random dynamical system $u = g(\varepsilon, t)$ as an example, a random interval $D(t) := \left[\underline{u},\ \bar{u}\right]$ at time t, within a short enough time interval $[t,\ t+\Delta t]$, changes to $D(t+\Delta t) := \left[\underline{u}(t+\Delta t),\ \bar{u}(t+\Delta t)\right]$ as the dynamical system $u(t+\Delta t) = g(\varepsilon, t+\Delta t)$ evolves. If, during $[t,\ t+\Delta t]$, no new random information is added to this system, and no random information is lost, then probability is conserved, which can be expressed as:

$$P(\varphi(\varepsilon, t) \in D(t)) = P(\varphi(\varepsilon, t+\Delta t) \in D(t+\Delta t)). \tag{3.2}$$

The aforementioned principle of probability conservation essentially states that: A nonlinear mapping $g(\varepsilon, t) \to g(\varepsilon, t+\Delta t)$ that is time-dependent and does not contain additional random information is probability-preserving. From this perspective, any mapping (not necessarily time-dependent), as long as it does not obtain or loss new random information, is also probability-preserving. Based on this idea, we can derive the following more general theorem of principle of probability conservation.

Lemma 1 (Principle of probability conservation). There exist two random real vectors $\mathbf{z}$ and θ, where the PDF of θ is $p_\theta(\theta)$, and $\mathbf{z}$ can be expressed in terms of θ as follows:

$$\mathbf{z} = (\mathbf{z}_1,\ \mathbf{z}_2,\ \cdots,\ \mathbf{z}_n)^{\mathrm{T}} = (h_1(\theta),\ h_2(\theta),\ \cdots,\ h_n(\theta))^{\mathrm{T}} = \mathbf{h}(\theta), \tag{3.3}$$

where $\mathbf{h}(\theta)$ is a function that depends only on θ and does not include any other random parameters. Then the joint PDF of $\mathbf{z}$ and θ is:

$$p_{z,\theta}(\mathbf{z}, \theta) = p_\theta(\theta)\,\delta(\mathbf{z} - \mathbf{h}(\theta)), \tag{3.4}$$

where $\delta(\mathbf{z} - \mathbf{h}(\theta)) = \prod_{i=1}^{n} \delta(z_i - h_i(\theta))$, and $\delta(\mathbf{.})$ is Dirac function.

Proof. Take any two random vectors $\mathbf{a}$ and $\mathbf{b}$, whose vector lengths are the same as those of $\mathbf{z}$ and θ, respectively, and we analyze the probability of $\mathbf{z} \le \mathbf{a} \bigcap \theta \le \mathbf{b}$. On the one hand, the probability can be represented by $p_{z,\theta}(\mathbf{z}, \theta)$ as

$$P\left(\mathbf{z} \le \mathbf{a} \bigcap \theta \le \mathbf{b}\right) = \int_{-\infty}^{\mathbf{a}} \int_{-\infty}^{\mathbf{b}} p_{z,\theta}(\mathbf{z}, \theta) \mathrm{d}\theta \mathrm{d}\mathbf{z} = \int_{-\infty}^{\mathbf{b}} \left[\int_{-\infty}^{\mathbf{a}} p_{z,\theta}(\mathbf{z}, \theta) \mathrm{d}\mathbf{z}\right] \mathrm{d}\theta. \tag{3.5}$$

On the other hand, consider that $\mathbf{z} = \mathbf{h}(\theta)$ is a function that maps θ to $\mathbf{z}$. In this mapping, no other random factors are introduced. Therefore, the probability of $\mathbf{z} \le \mathbf{a} \bigcap \theta \le \mathbf{b}$ should be the same as the probability of $\mathbf{h}(\theta) \le \mathbf{a} \bigcap \theta \le \mathbf{b}$, which means probability conservation, so there is:

$$P\left(\mathbf{z} \le \mathbf{a} \bigcap \theta \le \mathbf{b}\right) = \int_{-\infty}^{+\infty} I(\mathbf{h}(\theta) \le \mathbf{a}) I(\theta \le \mathbf{b}) p_\theta(\theta) \mathrm{d}\theta, \tag{3.6}$$

where, $I(\mathbf{.})$ is the characteristic function. By combining (3.5) and (3.6), we have

$$P\left(\mathbf{z} \le \mathbf{a} \bigcap \theta \le \mathbf{b}\right) = \int_{-\infty}^{\mathbf{b}} \left[\int_{-\infty}^{\mathbf{a}} p_{z,\theta}(\mathbf{z}, \theta) \mathrm{d}\mathbf{z}\right] \mathrm{d}\theta = \int_{-\infty}^{+\infty} I(\mathbf{h}(\theta) \le \mathbf{a}) I(\theta \le \mathbf{b}) p_\theta(\theta) \mathrm{d}\theta. \tag{3.7}$$

Taking the derivative of $\mathbf{a}$ and $\mathbf{b}$ on both sides of (3.7), there is

$$p_{z,\theta}(\mathbf{a}, \mathbf{b}) = \delta(\mathbf{a} - \mathbf{h}(\mathbf{b})) p_\theta(\mathbf{b}). \tag{3.8}$$

At this point, $\mathbf{a}$ and $\mathbf{b}$ can be replaced by any vector, so letting $\mathbf{a} = \mathbf{z}$ and $\mathbf{b} = \theta$ yields (3.4). □

Using Lemma 1, the following theorem can be derived:

Theorem 2. For (2.1), when $\mathbf{y}_{1:k+1}$ is known, $p_{x_{k+1}|y_{1:k+1}}(\mathbf{x}_{k+1}|\mathbf{y}_{1:k+1})$ can be expressed as:

$$p_{x_{k+1}|y_{1:k+1}}(\mathbf{x}_{k+1}|\mathbf{y}_{1:k+1}) = \frac{\int_{-\infty}^{+\infty} \begin{bmatrix} p_{x_0}(\mathbf{x}_0) p_{w_{0:k}}(\mathbf{w}_{0:k}) \\ \times p_{v_{1:k+1}}(\mathbf{y}_{1:k+1} - \gamma_{1:k+1}(\mathbf{x}_{1:k+1})) \\ \times \delta(\mathbf{x}_{k+1} - \varphi_k(\mathbf{x}_k, \mathbf{w}_k)) \end{bmatrix} \mathrm{d}\mathbf{x}_0 \mathrm{d}\mathbf{w}_{0:k}}{\int_{-\infty}^{+\infty} \begin{bmatrix} p_{x_0}(\mathbf{x}_0) p_{w_{0:k}}(\mathbf{w}_{0:k}) \\ \times p_{v_{1:k+1}}(\mathbf{y}_{1:k+1} - \gamma_{1:k+1}(\mathbf{x}_{1:k+1})) \end{bmatrix} \mathrm{d}\mathbf{x}_0 \mathrm{d}\mathbf{w}_{0:k}}, \tag{3.9}$$

where $p_{w_{0:k}}(\mathbf{w}_{0:k}) = \prod_{i=0}^{k} p_{w_i}(\mathbf{w}_i)$, $p_{v_{1:k+1}}(\mathbf{y}_{1:k+1} - \gamma_{1:k+1}(\mathbf{x}_{1:k+1})) = \prod_{i=1}^{k+1} p_{v_i}(\mathbf{y}_i - \gamma_i(\mathbf{x}_i))$, $\gamma_{1:k+1}(\mathbf{x}_{1:k+1}) = (\gamma_1(\mathbf{x}_1), \gamma_2(\mathbf{x}_2), \cdots, \gamma_{k+1}(\mathbf{x}_{k+1}))$.

Proof. According to (2.1):

$$\begin{aligned} \mathbf{x}_{k+1} &= \varphi_k(\mathbf{x}_k, \mathbf{w}_k) = \varphi_k(\varphi_{k-1}(\mathbf{x}_{k-1}, \mathbf{w}_{k-1}), \mathbf{w}_k) \\ &= \cdots = \varphi_k(\varphi_{k-1}(\cdots \varphi_0(\mathbf{x}_0, \mathbf{w}_0)), \mathbf{w}_k) \\ &= \tilde{\varphi}_k(\mathbf{x}_0, \mathbf{w}_{0:k}) \\ \mathbf{y}_{k+1} &= \gamma_{k+1}(\mathbf{x}_{k+1}) + \mathbf{v}_{k+1} \end{aligned}. \tag{3.10}$$

According to Lemma 1, there is a joint PDF as follows:

$$p(\mathbf{x}_{k+1}, \mathbf{y}_{1:k+1}, \mathbf{x}_0, \mathbf{v}_{1:k+1}, \mathbf{w}_{0:k}) = \begin{bmatrix} p_{x_0}(\mathbf{x}_0) p_{w_{0:k}}(\mathbf{w}_{0:k}) p_{v_{1:k+1}}(\mathbf{v}_{1:k+1}) \\ \times \delta(\mathbf{x}_{k+1} - \tilde{\varphi}_k(\mathbf{x}_0, \mathbf{w}_{0:k})) \prod_{i=1}^{k+1} \delta(\mathbf{y}_i - \gamma_i(\mathbf{x}_i) - \mathbf{v}_i) \end{bmatrix}. \tag{3.11}$$

Based on (3.11), the marginal PDF can be calculated as:

$$\begin{aligned} p_{x_{k+1},y_{1:k+1}}(\mathbf{x}_{k+1}, \mathbf{y}_{1:k+1}) &= \int_{-\infty}^{+\infty} p(\mathbf{x}_{k+1}, \mathbf{y}_{1:k+1}, \mathbf{x}_0, \mathbf{v}_{1:k+1}, \mathbf{w}_{0:k}) \mathrm{d}\mathbf{x}_0 \mathrm{d}\mathbf{w}_{0:k} \mathrm{d}\mathbf{v}_{1:k+1} \\ &= \int_{-\infty}^{+\infty} \begin{bmatrix} p_{x_0}(\mathbf{x}_0) p_{w_{0:k}}(\mathbf{w}_{0:k}) p_{v_{1:k+1}}(\mathbf{y}_{1:k+1} - \gamma_{1:k+1}) \\ \times \delta(\mathbf{x}_{k+1} - \tilde{\varphi}_k(\mathbf{x}_0, \mathbf{w}_{0:k})) \end{bmatrix} \mathrm{d}\mathbf{x}_0 \mathrm{d}\mathbf{w}_{0:k} \end{aligned}, \tag{3.12}$$

and

$$p_{y_{1:k+1}}(\mathbf{y}_{1:k+1}) = \int_{-\infty}^{+\infty} p_{x_{k+1},y_{1:k+1}}(\mathbf{x}_{k+1}, \mathbf{y}_{1:k+1}) \mathrm{d}\mathbf{x}_0 \mathrm{d}\mathbf{w}_{0:k} \mathrm{d}\mathbf{x}_{k+1} = \int_{-\infty}^{+\infty} \left[p_{x_0}(\mathbf{x}_0)\, p_{w_{0:k}}(\mathbf{w}_{0:k})\, p_{v_{1:k+1}}(\mathbf{y}_{1:k+1} - \gamma_{1:k+1})\right] \mathrm{d}\mathbf{x}_0 \mathrm{d}\mathbf{w}_{0:k} \tag{3.13}$$

Substituting (3.12) and (3.13) into (3.1) yields (3.9). □

Using the conditional probability density function obtained from Theorem 2, the mean and variance of the state can be calculated, leading to the following corollary:

Corollary 3. For (2.1), when $\mathbf{y}_{1:k+1}$ is known, the mean and variance of $\mathbf{x}_{k+1}$ are

$$\hat{\mathbf{x}}_{k+1} = \frac{\int_{-\infty}^{+\infty} \left[p_{x_0}(\mathbf{x}_0)\, p_{w_{0:k}}(\mathbf{w}_{0:k})\, \varphi_k(\mathbf{x}_k,\ \mathbf{w}_k)\, p_{v_{1:k+1}}(\mathbf{y}_{1:k+1} - \gamma_{1:k+1})\right] \mathrm{d}\mathbf{x}_0 \mathrm{d}\mathbf{w}_{0:k}}{\int_{-\infty}^{+\infty} \left[p_{x_0}(\mathbf{x}_0)\, p_{w_{0:k}}(\mathbf{w}_{0:k})\, p_{v_{1:k+1}}(\mathbf{y}_{1:k+1} - \gamma_{1:k+1})\right] \mathrm{d}\mathbf{x}_0 \mathrm{d}\mathbf{w}_{0:k}}, \tag{3.14}$$

and

$$\mathbf{P}_{k+1} = \frac{\int_{-\infty}^{+\infty} \left[\begin{array}{l} p_{x_0}(\mathbf{x}_0)\, p_{w_{0:k}}(\mathbf{w}_{0:k})\, p_{v_{1:k+1}}(\mathbf{y}_{1:k+1} - \gamma_{1:k+1}) \\ \times (\varphi_k(\mathbf{x}_k,\ \mathbf{w}_k) - \hat{\mathbf{x}}_{k+1}) (\varphi_k(\mathbf{x}_k,\ \mathbf{w}_k) - \hat{\mathbf{x}}_{k+1})^{\mathrm{T}} \end{array}\right] \mathrm{d}\mathbf{x}_0 \mathrm{d}\mathbf{w}_{0:k}}{\int_{-\infty}^{+\infty} \left[p_{x_0}(\mathbf{x}_0)\, p_{w_{0:k}}(\mathbf{w}_{0:k}) \times p_{v_{1:k+1}}(\mathbf{y}_{1:k+1} - \gamma_{1:k+1})\right] \mathrm{d}\mathbf{x}_0 \mathrm{d}\mathbf{w}_{0:k}}. \tag{3.15}$$

The quotient-form mean and variance of $\mathbf{x}_{k+1}$ can be calculated using (3.14) and (3.15). However, it must be pointed out that calculating the mean and variance of $\mathbf{x}_{k+1}$ based on (3.14) and (3.15) may result in a very small denominator and numerator, leading to numerical calculation failure. To explain this point, we use the example where the measurement noise is white noise. At this point, $p_{v_{1:k+1}}(\mathbf{y}_{1:k+1} - \gamma_{1:k+1}) = \prod_{i=1}^{k+1} p_{v_i}\left(\mathbf{y}_i - \gamma_i\left(\mathbf{x}_i\right)\right)$, where the PDF $p_{v_i}\left(\mathbf{y}_i - \gamma_i\left(\mathbf{x}_i^{(j)}\right)\right)$ represents the likelihood of $\mathbf{y}_i - \gamma_i\left(\mathbf{x}_i\right)$ occurring. Since $\mathbf{v}_i$ represents zero mean noise, $p_{v_i}\left(\mathbf{y}_i - \gamma_i\left(\mathbf{x}_i^{(j)}\right)\right)$ essentially represents the likelihood of $\gamma_i\left(\mathbf{x}_i\right)$ being close to $\mathbf{y}_i$, and the closer $\gamma_i\left(\mathbf{x}_i\right)$ and $\mathbf{y}_i$ are, the greater the likelihood is. $\prod_{i=1}^{k+1} p_{v_i}\left(\mathbf{y}_i - \gamma_i\left(\mathbf{x}_i\right)\right)$ represents the likelihood that the measurement $\mathbf{y}_i$ is also very close to $\gamma_i\left(\mathbf{x}_i\right)$ at all moments. Thus, as k increases, this likelihood will gradually decrease; i.e., $\lim_{k\to\infty} \prod_{i=1}^{k+1} p_{v_i}\left(\mathbf{y}_i - \gamma_i\left(\mathbf{x}_i\right)\right) = 0$. Even if the measurement noise is not white noise, $p_{v_{1:k+1}}(\mathbf{y}_{1:k+1} - \gamma_{1:k+1})$ still represents the likelihood that the measurement $\mathbf{y}_i$ is very close to $\gamma_i\left(\mathbf{x}_i\right)$ at all times. Therefore, as k increases, this likelihood will gradually decrease, indicating that $p_{v_{1:k+1}}(\mathbf{y}_{1:k+1} - \gamma_{1:k+1})$ approaches 0, which will result in serious rounding errors and numerical instability.

To address the numerical instability issue analyzed above, in the following subsection, we propose a new key conditional quotient filter.

3.2. Key conditional quotient filter

The reason for the instability of the model calculation in the previous subsection is essentially due to too many measurement conditions. Note that (3.14) and (3.15) are quotient-form analytical expressions for the estimation of $\mathbf{x}_{k+1}$ when $\mathbf{y}_{1:k+1} = (\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_{k+1})$ is known. As k increases, $p_{v_{i:k+1}}(\mathbf{y}_{i:k+1} - \gamma_{i:k+1})$ will decrease, resulting in smaller numerators and denominators in (3.14) and (3.15), leading to significant errors in numerical calculations. In fact, it seems that not all

measurement conditions are necessary to estimate $\mathbf{x}_{k+1}$. For instance, when estimating $\mathbf{x}_{k+1}$, although measurements $\mathbf{y}_{1:k+1} = (\mathbf{y}_1,\ \mathbf{y}_2,\ \cdots,\ \mathbf{y}_{k+1})$ have been taken, it is not always necessary to utilize all the measurements. Take the estimation problem of a satellite's trajectory as an example. Take the state equation of the satellite around the Earth, with the state includes four variables: the radius of the satellite trajectory, the rate of change of the radius, the angle of the satellite trajectory, and the rate of change of the angle. Measurements of the satellite's radius and angle of motion are taken daily to obtain $(\mathbf{y}_1,\ \mathbf{y}_2,\ \cdots,\ \mathbf{y}_{30})$. Now, it is necessary to estimate the state $\mathbf{x}_{30}$ at the end of the month. The state $\mathbf{x}_{30}$ at the end of the month is different from $\mathbf{x}_1$ at the beginning of the month. Moreover, the impact of the initial measurement $\mathbf{y}_1$ on $\mathbf{x}_{30}$ is not as critical as the impact of the measurement $\mathbf{y}_{30}$ on $\mathbf{x}_{30}$. Alternatively, when estimating the radius dynamics, the measurement radius is a more important measurement condition compared to the measurement angle. Thus, the fundamental premise of the KCQF is that, despite having obtained a great number of measurements $\mathbf{y}_{1:k+1} = (\mathbf{y}_1,\ \mathbf{y}_2,\ \cdots,\ \mathbf{y}_{k+1})$, to accurately estimate $\mathbf{x}_{k+1}$, one should focus on selecting the measurement conditions that are key for $\mathbf{x}_{k+1}$ and disregard measurements with weaker correlations.

Assuming that within the obtained measurements, only a portion of the measurements is key for estimating $\mathbf{x}_{k+1}$, it follows that an estimation of the state should be based solely on the key measurement conditions. To concretely illustrate this concept, let us denote the key measurements as $\mathbf{z}_{k+1} := \mathbf{y}_{1:k+1}\mathbf{A}_{k+1}$. Here, $\mathbf{A}_{k+1}$ represents the operational operator for extracting the measurement conditions that are key for the estimation of $\mathbf{x}_{k+1}$ from the measurement matrix $\mathbf{y}_{1:k+1}$. For instance, if $(\mathbf{y}_k,\ \mathbf{y}_{k+1})$ is very important for the estimation of $\mathbf{x}_{k+1}$, we can set $\mathbf{A}_{k+1} = \begin{bmatrix} 0 & \cdots & 1 & 0 \\ 0 & \cdots & 0 & 1 \end{bmatrix}^{\mathrm{T}}$, thus $\mathbf{z}_{k+1} = \mathbf{y}_{1:k+1}\mathbf{A}_{k+1} = (\mathbf{y}_k,\ \mathbf{y}_{k+1})$. The measurement error of $\mathbf{z}_{k+1}$ is denoted as $\beta_{k+1} := \mathbf{v}_{1:k+1}\mathbf{A}_{k+1}$.

To estimate the $\mathbf{x}_{k+1}$ based on key measurement conditions, it is necessary to know the CPDF $p_{x_{k+1}|z_{k+1}}(\mathbf{x}_{k+1}|\mathbf{z}_{k+1})$.

Using Lemma 1 and introducing the idea of key conditions on the basis of Theorem 2, the following theorem can be established:

Theorem 4. For (2.1), when $\mathbf{y}_{1:k+1}$, $\mathbf{A}_{k+1}$ and the PDF $p_{\beta_{k+1}}(\beta_{k+1})$ of β_{k+1} is known, and let $\gamma_{1:k+1}(\mathbf{x}_{1:k+1}) = (\gamma_1(\mathbf{x}_1),\ \gamma_2(\mathbf{x}_2),\ \cdots,\ \gamma_{k+1}(\mathbf{x}_{k+1}))$, and $\mathbf{z}_{k+1} = \mathbf{y}_{1:k+1}\mathbf{A}_{k+1}$, then the CPDF $p_{x_{k+1}|z_{k+1}}(\mathbf{x}_{k+1}|\mathbf{z}_{k+1})$ can be expressed as:

$$p_{x_{k+1}|z_{k+1}}(\mathbf{x}_{k+1}|\mathbf{z}_{k+1}) = \frac{\int_{-\infty}^{+\infty}\begin{bmatrix} p_{x_0}(\mathbf{x}_0)\, p_{w_{0:k}}(\mathbf{w}_{0:k}) \\ \times\, p_{\beta_{k+1}}(\mathbf{z}_{k+1} - \gamma_{1:k+1}(\mathbf{x}_{1:k+1})\,\mathbf{A}_{k+1}) \\ \times\, \delta(\mathbf{x}_{k+1} - \varphi_k(\mathbf{x}_k,\ \mathbf{w}_k)) \end{bmatrix}\mathrm{d}\mathbf{x}_0\mathrm{d}\mathbf{w}_{0:k}}{\int_{-\infty}^{+\infty}\begin{bmatrix} p_{x_0}(\mathbf{x}_0)\, p_{w_{0:k}}(\mathbf{w}_{0:k}) \\ \times\, p_{\beta_{k+1}}(\mathbf{z}_{k+1} - \gamma_{1:k+1}(\mathbf{x}_{1:k+1})\,\mathbf{A}_{k+1}) \end{bmatrix}\mathrm{d}\mathbf{x}_0\mathrm{d}\mathbf{w}_{0:k}}. \tag{3.16}$$

Proof. According to (2.1)

$$\begin{aligned} \mathbf{x}_{k+1} &= \boldsymbol{\varphi}_k(\mathbf{x}_k, \mathbf{w}_k) = \tilde{\boldsymbol{\varphi}}_k(\mathbf{x}_0, \mathbf{w}_{0:k}) \\ \mathbf{z}_{k+1} &= \boldsymbol{\gamma}_{1:k+1}(\mathbf{x}_{1:k+1})\mathbf{A}_{k+1} + \boldsymbol{\beta}_{k+1}. \end{aligned} \tag{3.17}$$

According to Lemma 1, there is a joint PDF as follows:

$$\begin{aligned} p(\mathbf{x}_{k+1}, \mathbf{z}_{k+1}, \mathbf{x}_0, \beta_{k+1}, \mathbf{w}_{0:k}) = {} & p_{x_0}(\mathbf{x}_0)\, p_{w_{0:k}}(\mathbf{w}_{0:k})\, p_{\beta_{k+1}}(\beta_{k+1}) \\ & \times \delta(\mathbf{x}_{k+1} - \tilde{\varphi}_k(\mathbf{x}_0, \mathbf{w}_{0:k}))\, \delta(\mathbf{z}_{k+1} - \gamma_{1:k+1}(\mathbf{x}_{1:k+1})\,\mathbf{A}_{k+1} - \beta_{k+1}). \end{aligned} \tag{3.18}$$

According to (3.18), and analogous to the derivations of (3.12) and (3.13), the marginal PDF can be calculated as:

$$p_{x_{k+1},z_{k+1}}\left(\mathbf{x}_{k+1},\mathbf{z}_{k+1}\right)=\int_{-\infty}^{+\infty}\begin{bmatrix}p_{x_0}\left(\mathbf{x}_0\right)p_{w_{0:k}}\left(\mathbf{w}_{0:k}\right)\\ \times p_{\beta_{k+1}}\left(\mathbf{z}_{k+1}-\gamma_{1:k+1}\left(\mathbf{x}_{1:k+1}\right)\mathbf{A}_{k+1}\right)\\ \times\delta\left(\mathbf{x}_{k+1}-\tilde{\varphi}_k\left(\mathbf{x}_0,\mathbf{w}_{0:k}\right)\right)\end{bmatrix}\mathrm{d}\mathbf{x}_0\mathrm{d}\mathbf{w}_{0:k},\tag{3.19}$$

and

$$p_{z_{k+1}}\left(\mathbf{z}_{k+1}\right)=\int_{-\infty}^{+\infty}\left[p_{x_0}\left(\mathbf{x}_0\right)p_{w_{0:k}}\left(\mathbf{w}_{0:k}\right)p_{\beta_{k+1}}\left(\mathbf{z}_{k+1}-\gamma_{1:k+1}\left(\mathbf{x}_{1:k+1}\right)\mathbf{A}_{k+1}\right)\right]\mathrm{d}\mathbf{x}_0\mathrm{d}\mathbf{w}_{0:k}.\tag{3.20}$$

Substituting (3.19) and (3.20) into $p_{x_{k+1}|z_{k+1}}\left(\mathbf{x}_{k+1}|\mathbf{z}_{k+1}\right)=\frac{p_{\mathbf{x}_{k+1},\mathbf{z}_{k+1}}\left(\mathbf{x}_{k+1},\,\mathbf{z}_{k+1}\right)}{p_{\mathbf{z}_{k+1}}\left(\mathbf{z}_{k+1}\right)}$ yields (3.16). □

Using the conditional probability density function with key measurement conditions obtained from Theorem 4, the mean and variance of the state can be calculated, yielding the following corollary:

Corollary 5. For (2.1), when $\mathbf{z}_{k+1}=\mathbf{y}_{1:k+1}\mathbf{A}_{k+1}$, $\beta_{k+1}=\mathbf{v}_{1:k+1}\mathbf{A}_{k+1}$ and $p_{\beta_{k+1}}\left(\beta_{k+1}\right)$ is known, the mean and variance of $\mathbf{x}_{k+1}$ can be expressed as:

$$\hat{\mathbf{x}}_{k+1|\mathbf{z}_{k+1}}=\frac{\int_{-\infty}^{+\infty}\begin{bmatrix}p_{x_0}\left(\mathbf{x}_0\right)p_{w_{0:k}}\left(\mathbf{w}_{0:k}\right)\varphi_k\left(\mathbf{x}_k,\ \mathbf{w}_k\right)\\ \times p_{\beta_{k+1}}\left(\mathbf{z}_{k+1}-\gamma_{1:k+1}\left(\mathbf{x}_{1:k+1}\right)\mathbf{A}_{k+1}\right)\end{bmatrix}\mathrm{d}\mathbf{x}_0\mathrm{d}\mathbf{w}_{0:k}}{\int_{-\infty}^{+\infty}\left[p_{x_0}\left(\mathbf{x}_0\right)p_{w_{0:k}}\left(\mathbf{w}_{0:k}\right)p_{\beta_{k+1}}\left(\mathbf{z}_{k+1}-\gamma_{1:k+1}\left(\mathbf{x}_{1:k+1}\right)\mathbf{A}_{k+1}\right)\right]\mathrm{d}\mathbf{x}_0\mathrm{d}\mathbf{w}_{0:k}},\tag{3.21}$$

and

$$\mathbf{P}_{k+1|\mathbf{z}_{k+1}}=\frac{\int_{-\infty}^{+\infty}\begin{bmatrix}p_{x_0}\left(\mathbf{x}_0\right)p_{w_{0:k}}\left(\mathbf{w}_{0:k}\right)p_{\beta_{k+1}}\left(\mathbf{z}_{k+1}-\gamma_{1:k+1}\left(\mathbf{x}_{1:k+1}\right)\mathbf{A}_{k+1}\right)\\ \times\left(\varphi_k\left(\mathbf{x}_k,\ \mathbf{w}_k\right)-\hat{\mathbf{x}}_{k+1|\mathbf{z}_{1+k}}\right)\left(\varphi_k\left(\mathbf{x}_k,\ \mathbf{w}_k\right)-\hat{\mathbf{x}}_{k+1|\mathbf{z}_{1+k}}\right)^{\mathrm{T}}\end{bmatrix}\mathrm{d}\mathbf{x}_0\mathrm{d}\mathbf{w}_{0:k}}{\int_{-\infty}^{+\infty}\left[p_{x_0}\left(\mathbf{x}_0\right)p_{w_{0:k}}\left(\mathbf{w}_{0:k}\right)p_{\beta_{k+1}}\left(\mathbf{z}_{k+1}-\gamma_{1:k+1}\left(\mathbf{x}_{1:k+1}\right)\mathbf{A}_{k+1}\right)\right]\mathrm{d}\mathbf{x}_0\mathrm{d}\mathbf{w}_{0:k}}.\tag{3.22}$$

Equations (3.21) and (3.22) can be used to calculate the mean and variance of $\mathbf{x}_{k+1}$ based on the key conditions. The MC method is utilized to solve the high-dimensional integrals present in (3.21) and (3.22). Assuming sampling based on distributions $p_{x_0}\left(\mathbf{x}_0\right)$ and $p_{w_{0:k}}\left(\mathbf{w}_{0:k}\right)$, N_{s} samples $\mathbf{x}_0^{(j)}$ and $\mathbf{w}_{0:k}^{(j)}$ can be obtained, where $j=1,\ 2,\ \cdots,\ N_{\mathrm{s}}$. By iteratively calculating in terms of (2.1), N_{s} samples of state $x_{k+1}^{(j)}=\varphi_k\left(\mathbf{x}_k^{(j)},\ \mathbf{w}_k^{(j)}\right)$ can be obtained. According to the MC method,

$$\begin{aligned}&\int_{-\infty}^{+\infty}\left[p_{x_0}\left(\mathbf{x}_0\right)p_{w_{0:k}}\left(\mathbf{w}_{0:k}\right)p_{\beta_{k+1}}\left(\mathbf{z}_{k+1}-\gamma_{1:k+1}\left(\mathbf{x}_{1:k+1}\right)\mathbf{A}_{k+1}\right)\right]\mathrm{d}\mathbf{x}_0\mathrm{d}\mathbf{w}_{0:k}\\ &\approx\frac{1}{N_{\mathrm{s}}}\sum_{j=1}^{N_{\mathrm{s}}}p_{\beta_{k+1}}\left(\mathbf{z}_{k+1}-\gamma_{1:k+1}\left(\mathbf{x}_{1:k+1}^{(j)}\right)\mathbf{A}_{k+1}\right)\end{aligned},\tag{3.23}$$

$$\begin{aligned}&\int_{-\infty}^{+\infty}\begin{bmatrix}p_{x_0}\left(\mathbf{x}_0\right)p_{w_{0:k}}\left(\mathbf{w}_{0:k}\right)\varphi_k\left(\mathbf{x}_k,\ \mathbf{w}_k\right)\\ \times p_{\beta_{k+1}}\left(\mathbf{z}_{k+1}-\gamma_{1:k+1}\left(\mathbf{x}_{1:k+1}\right)\mathbf{A}_{k+1}\right)\end{bmatrix}\mathrm{d}\mathbf{x}_0\mathrm{d}\mathbf{w}_{0:k}\\ &\approx\frac{1}{N_{\mathrm{s}}}\sum_{j=1}^{N_{\mathrm{s}}}\begin{bmatrix}\varphi_k\left(\mathbf{x}_k^{(j)},\ \mathbf{w}_k^{(j)}\right)\\ \times p_{\beta_{k+1}}\left(\mathbf{z}_{k+1}-\gamma_{1:k+1}\left(\mathbf{x}_{1:k+1}^{(j)}\right)\mathbf{A}_{k+1}\right)\end{bmatrix}\end{aligned},\tag{3.24}$$

and

$$\begin{aligned}&\int_{-\infty}^{+\infty}\left[\begin{array}{l}p_{x_0}\left(\mathbf{x}_0\right)p_{w_{0:k}}\left(\mathbf{w}_{0:k}\right)p_{\beta_{k+1}}\left(\mathbf{z}_{k+1}-\gamma_{1:k+1}\left(\mathbf{x}_{1:k+1}\right)\mathbf{A}_{k+1}\right)\\ \times\left(\varphi_k\left(\mathbf{x}_k,\ \mathbf{w}_k\right)-\hat{\mathbf{x}}_{k+1|\mathbf{z}_{1+k}}\right)\left(\varphi_k\left(\mathbf{x}_k,\ \mathbf{w}_k\right)-\hat{\mathbf{x}}_{k+1|\mathbf{z}_{1+k}}\right)^{\mathrm{T}}\end{array}\right]\mathrm{d}\mathbf{x}_0\mathrm{d}\mathbf{w}_{0:k}\\ &\approx\frac{1}{N_{\mathrm{s}}}\sum_{j=1}^{N_{\mathrm{s}}}\left[\begin{array}{l}\left(\varphi_k\left(\mathbf{x}_k^{(j)},\ \mathbf{w}_k^{(j)}\right)-\hat{\mathbf{x}}_{k+1|\mathbf{z}_{1+k}}\right)\left(\varphi_k\left(\mathbf{x}_k^{(j)},\ \mathbf{w}_k^{(j)}\right)-\hat{\mathbf{x}}_{k+1|\mathbf{z}_{1+k}}\right)^{\mathrm{T}}\\ \times p_{\beta_{k+1}}\left(\mathbf{z}_{k+1}-\gamma_{1:k+1}\left(\mathbf{x}_{1:k+1}^{(j)}\right)\mathbf{A}_{k+1}\right)\end{array}\right]\end{aligned}. \tag{3.25}$$

We substitute (3.23)–(3.25) into (3.21) and (3.22) to calculate $\hat{\mathbf{x}}_{k+1|\mathbf{z}_{k+1}}$ and $\mathbf{P}_{k+1|\mathbf{z}_{k+1}}$. Due to the law of large numbers [38], the error order of (3.23)–(3.25) is $O\left(N_{\mathrm{s}}^{-0.5}\right)$. Therefore, as the sample size N_s increases, the results of (3.23)–(3.25) will gradually converge to the exact integral.

Corollary 5 provides the quotient-form estimation expressions for the state based on key conditions. What distinguishes these expressions from (3.14) and (3.15), which are based on all measurement conditions, is that in the estimation expressions for the state based on key conditions, only the PDFs of some key measurements are used. When calculating using (3.14) and (3.15), the high-dimensional joint PDF $p_{v_{1:k+1}}\left(\mathbf{y}_{1:k+1}-\gamma_{1:k+1}\right)$ that appears in the denominator will gradually approach 0 as k increases, leading to calculation failure. When calculating using (3.23)-(3.25), $p_{\beta_{k+1}}\left(\mathbf{z}_{k+1}-\gamma_{1:k+1}\left(\mathbf{x}_{1:k+1}\right)\mathbf{A}_{k+1}\right)$ is merely the joint PDF of the errors in the key measurements. If the number of key conditions does not increase with the increase in k, then $p_{\beta_{k+1}}\left(\mathbf{z}_{k+1}-\gamma_{1:k+1}\left(\mathbf{x}_{1:k+1}\right)\mathbf{A}_{k+1}\right)$ will not tend toward 0. Therefore, estimating the state based on key conditions can prevent the issue of dividing a small number by another small number, which effectively ensures computational stability.

Moreover, the issue of how to extract key measurements requires further discussion. In this paper, we use correlation coefficients to assess the reference value of different measurements for the estimation of state $\mathbf{x}_{k+1}$. For instance, when considering whether to utilize the i-th measurement data in $\mathbf{y}_j$ (denoted as $y_{j,i}$) to estimate the m-th state in $\mathbf{x}_{k+1}$ (denoted as $x_{k+1,m}$), the reference value of $y_{j,i}$ can be defined as

$$r\left(y_{i,j},x_{k+1,m}\right)=\left|\frac{\operatorname{cov}\left(y_{i,j},x_{k+1,m}\right)}{\sqrt{\operatorname{cov}\left(y_{i,j},y_{i,j}\right)}\sqrt{\operatorname{cov}\left(x_{k+1,m},x_{k+1,m}\right)}}\right|, \tag{3.26}$$

where $\operatorname{cov}(\mathbf{.}\ ,\ \mathbf{.})$ represents the covariance of two random variables, which can also be conveniently calculated through MC integration. When evaluating $\mathbf{x}_{k+1}$, we extrapolate k' steps forward from the current $k+1$ steps and assume that measurements prior to k' have little reference value for evaluating $\mathbf{x}_{k+1}$. Then, using (3.26), we calculate the reference value of the measurement data within the time steps k' to $k+1$, and select the d measurements with the greatest reference value as the key measurements $\mathbf{z}_{k+1}$ for evaluating $\mathbf{x}_{k+1}$.

After the key measurement conditions $\mathbf{z}_{k+1}$ are determined, its error is $\beta_{k+1}:=\mathbf{v}_{1:k+1}\mathbf{A}_{k+1}=\left(\beta_{k_1},\ \cdots,\ \beta_{k_d}\right)$, where $\beta_{k_1},\ \beta_{k_2},\ \cdots,\ \beta_{k_d}$ represents the measurement noise of the key measurement conditions. In fact, $\beta_{k_1},\ \beta_{k_2},\ \cdots,\ \beta_{k_d}$ are composed of d random variables extracted from the measurement noise $\mathbf{v}_{1:k+1}$. Therefore, $p_{\beta_{k+1}}\left(\beta_{k+1}\right)$ is the marginal PDF of $p_{v_{1:k+1}}\left(\mathbf{v}_{1:k+1}\right)$ and can be expressed as:

$$p_{\beta_{k+1}}\left(\beta_{k+1}\right)=\int_{-\infty}^{+\infty}p_{v_{1:k+1}}\left(\mathbf{v}_{1:k+1}\right)\mathrm{d}v_{i_1}\mathrm{d}v_{i_2}\cdots\mathrm{d}v_{i_{k+1-d}}, \tag{3.27}$$

where $v_{i_1},\ v_{i_2},\ \cdots,\ v_{i_{k+1-d}}$ represents the measurement noise of the data that have not been selected as key measurement conditions. When the measurement noise $\mathbf{v}_{1:k+1}$ is a Gaussian process or white noise,

$p_{\beta_{k+1}}(\beta_{k+1})$ can be obtained using (3.27). When the situation of $p_{v_{1:k+1}}(\mathbf{v}_{1:k+1})$ is more complex, MC can also be used to approximate the integration of (3.27).

This new filter proposed in Section 3.2 is named the key conditional quotient filter (KCQF). Considering the estimation of the mean and variance of x_{k+1} when key measurements are known, the pseudocode of the KCQF is shown in Algorithm 1, in which K is the number of time steps.

In the next section, we will further demonstrate the advantages of KCQF through three numerical examples.

Algorithm 1. Key Conditional Quotient Filter.

Input: $p_{x_0}(\mathbf{x}_0)$, $p_{w_{0:k}}(\mathbf{w}_{0:k})$, $p_{v_{1:k+1}}(\mathbf{v}_{1:k+1})$, $\mathbf{y}_{0:K}$, φ_k, γ_{k+1}, d, K

Output $\hat{\mathbf{x}}_{k+1|\mathbf{z}_{k+1}}$, $\mathbf{P}_{k+1|\mathbf{z}_{k+1}}$

Sample according to $p_{x_0}(\mathbf{x}_0)$ and $p_{w_{0:k}}(\mathbf{w}_{0:k})$ to N_{s} samples $\mathbf{x}_0^{(j)}$ and $\mathbf{w}_{0:k}^{(j)}$, $j = 1, 2, \cdots, N_{\mathrm{s}}$;

for $k = 1: K$

for $j = 1: N_s$

Calculate $\mathbf{x}_{k+1}^{(j)} = \varphi_k\left(\mathbf{x}_k^{(j)}, \mathbf{w}_k^{(j)}\right)$ using (2.1) ;

end

end

for $k = 1: K$

Using (3.26) to select d key measurements $\mathbf{z}_{k+1}$, and obtain $p_{\beta_{k+1}}(\beta_{k+1})$ based on (3.27);

Calculate three high-dimensional integrals using (3.23)-(3.25);

Calculate $\hat{\mathbf{x}}_{k+1|\mathbf{z}_{k+1}}$ and $\mathbf{P}_{k+1|\mathbf{z}_{k+1}}$ using (3.21) and (3.22) ;

end

The main computational overhead of the proposed KCQF algorithm comes from Monte Carlo sampling and high-dimensional integral solving under key measurements. The theoretical computational complexity is $O(N_S \cdot (n+d) \cdot K)$, where N_s is the number of Monte Carlo samples, n is the dimension of the system state, d is the dimension of key measurements, and K is the number of iteration steps.

4. Numerical example

4.1. Gaussian Markovian nonlinear example

The first example verifies the computational performance of KCQF proposed in Section III through a widely used Gaussian and Markovian nonlinear numerical example. This example or its variants have been extensively studied [39, 40], and the model is

$$\begin{cases} x_{k+1} = \dfrac{1}{2}x_k + \dfrac{25x_k}{1+x_k^2} + 8\cos(1.2k) + w_{k+1} \\ y_{k+1} = \dfrac{x_{k+1}^2}{20} + v_{k+1}, \ k = 1, 2, \cdots, K \end{cases}. \tag{4.1}$$

The process noise w_k and measurement noise v_k are assumed to be independent zero mean Gaussian random variables, where $w_k \sim \mathcal{N}(0, 10)$ and $v_k \sim \mathcal{N}(0, 1)$. The initial value is $x_0 \sim N(0, 2)$, and the number of iteration steps is $K = 52$. A MC averaged root mean squared error $E_{\text{rms}}(k)$ is considered for evaluating the accuracy of the estimates. The $E_{\text{rms}}(k)$ is computed over a set of MC runs.

$$E_{\text{rms}}(k) = \sqrt{\frac{1}{M}\sum_{m=1}^{M}(x^m(k) - \hat{x}^m(k))^2}, \tag{4.2}$$

where M is the number of MC runs, and in this example, $M = 50$. $x^m(k)$ and $\hat{x}^m(k)$ represent the actual and estimated states at the time instant k during the m-th MC run. The time averaged error $\overline{E_{\text{rms}}}$ can be calculated by

$$\overline{E_{\text{rms}}} = \frac{1}{K}\sum_{k=1}^{K} E_{\text{rms}}(k), \tag{4.3}$$

To study the impact of the number of key conditions d, we test the performance of the KCQF with $d = 1,\ 2,\ 3$ and 4, and plotted the E_{rms} in Figure 1, where the number of samples $N_{\text{s}} = 50$. For the convenience of discussion, we denote the KCQF with d key conditions as KCQF-d. From Figure 1, it shows that the computational results of KCQF-2 and KCQF-3 are better. When $d = 2$ and 3, the E_{rms} is superior to that when $d = 1$, indicating that considering more key measurements yields a more accurate estimation result than considering only one key measurement. When $d = 2$ and 3, the E_{rms} is superior to that when $d = 4$. This phenomenon corroborates the point made in Section 3.2: If too many measurement values are considered, $p_{\beta_{k+1}}(\beta_{k+1})$ will become too small, leading to the numerator and denominator in (3.14) and (3.15) to become small, thereby increasing numerical errors. Therefore, in practical applications, we recommend initializing with $d = 2$ or 3 and further fine-tuning based on the estimation results of the specific problem.

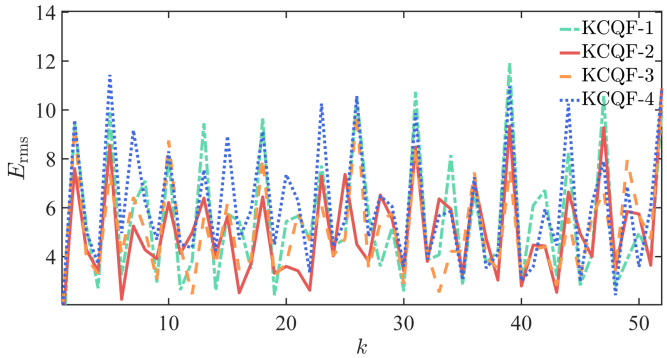

Figure 1. Performance comparison for different numbers of KCQFs.

Next, we will compare the KCQF with some existing filters. This example calculates the $\overline{E_{\text{rms}}}$ of five filters, namely PGM-UT (particle Gaussian mixture with unscented transform), PGM (particle Gaussian mixture) [39], PF-RR (residual resampling) [41], PF-SR (stratified resampling) [42], and UKF filters. We then compare these with the $\overline{E_{\text{rms}}}$ calculated by the KCQF-2 and KCQF-3 methods. The results are plotted in Figure 2. To ensure fairness, the number of samples and the number of particles N_{s} is uniformly set to 50. Furthermore, this example also compares the CPU times taken by

different filters during operation. The results are plotted in Figure 3. Thus, we suggest trading off between estimation accuracy and computational efficiency in practice: Increase N_s for high-precision requirements, and decrease it for real-time applications.

The results shown in Figure 2 indicate that the proposed KCQF-2 and KCQF-3 filters achieve superior estimation accuracy with lower $\overline{E_{\text{rms}}}$ than that of other comparative filters. Furthermore, it is not difficult to see from Figure 2 that for this nonlinear model, KCQF-2 yields the best calculation result. Figure 3 demonstrates that the KCQF-2 and KCQF-3 filters have excellent computational efficiency, with CPU times only slightly lower than the UKF filter and significantly less than other compared filters. We further plot the convergence curves of KCQF-2 and KCQF-3 with different numbers of samples in Figure 4. It is not difficult to see from Figure 4 that when the number of samples $N_s = 50$, the $\overline{E_{\text{rms}}}$ calculated by KCQF-2 is less than 5. As the number of samples gradually increases, the $\overline{E_{\text{rms}}}$ obtained from KCQF-2 and KCQF-3 gradually decrease in an oscillatory manner and converge to $\overline{E_{\text{rms}}} = 4.5$.

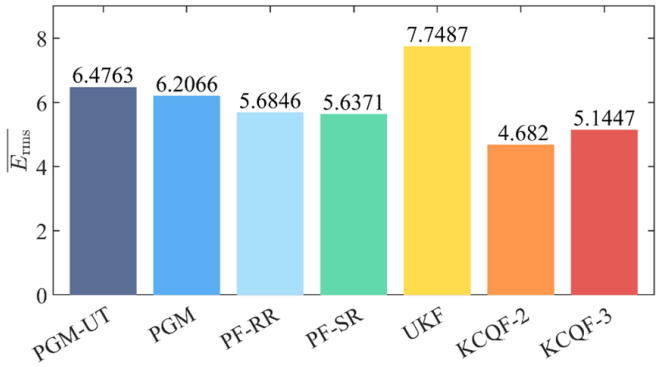

Figure 2. $\overline{E_{\text{rms}}}$ comparison for different filters.

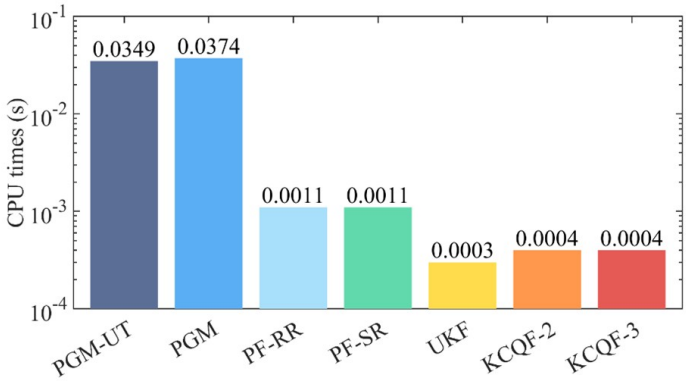

Figure 3. CPU times comparison for different filters.

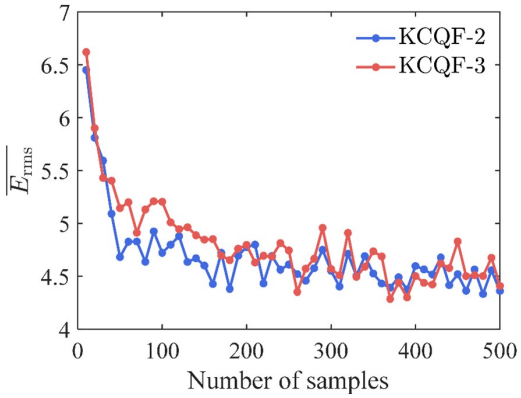

Figure 4. Convergence curves of KCQF-2 and KCQF-3.

4.2. Non-Gaussian, non-Markovian nonlinear example

The second example continues to employ the state and observation models described by (4.1) in Subsection 4.1 and retains the initial conditions and observation noise used in that subsection. However, the difference lies in the process noise, which is no longer Gaussian white noise but is assumed to be a non-Gaussian with a mean of $\bar{w} = 0$ and a variance of 10. Utilizing the Karhunen-Loeve (K-L) expansion [43] to establish the process noise, w_{k+1} can be expressed as:

$$w_{k+1} = \bar{w} + \sum_{n=1}^{M} \xi_n \sqrt{\lambda_n} f_{n,k+1}, \tag{4.4}$$

where, $\xi_n \in \left[-\sqrt{30},\ \sqrt{30}\right]$ is a uniform random variable and $M = 6$ is the expansion order of the K-L expansion. $f_{n,k+1}$ and λ_n are the $n-$ th eigenfunctions and eigenvalues of the autocorrelation function $rho\,(i,\ j)$, satisfying:

$$\begin{gathered} \rho\,(i,j)\, f_{n,j} = \lambda_n f_{n,j},\ 1 \le i,\ j,\ n \le K,\ \lambda_1 > \lambda_2 > \cdots > \lambda_6 \\ \sum_{i=1}^{K} f_{m,i} f_{n,i} = \delta_{m,n},\quad \sum_{i=1}^{K} f_{m,i} \rho\,(i,j)\, f_{n,i} = \lambda_n \delta_{m,n} \end{gathered}, \tag{4.5}$$

in which $\delta_{m,n}$ is the Kronecker delta function. In this example, the autocovariance function $\rho\,(i,\ j)$ is defined as:

$$\rho\,(i,\ j) = \exp\left[-\left(\frac{i-j}{15}\right)^2\right], \tag{4.6}$$

To illustrate the non-Gaussian and non-Markovian characteristics of the problem, we conduct random sampling on ξ 1000000 times , and based on the results, plotted the PDF of w_{26}, as depicted by the red solid line in Figure 5(a). The blue dashed line in Figure 5(a) represents the Gaussian PDF plotted based on the mean and variance of w_{26}. From Figure 5(a) , it is shown that w_{26} does not follow a Gaussian distribution. To further illustrate that the problem is not a Markov process, we use the same method to plot the conditional PDF $p_{x_2|x_1}\,(x_2|x_1)$ and $p_{x_2|x_1,x_0}\,(x_2|x_1,x_0)$ under conditions $x_0 = -0.5$ and $x_1 = -0.2$, respectively. As shown in Figure 5(b), it is evident from Figure 5(b) that $p_{x_2|x_1}\,(x_2|x_1) \neq p_{x_2|x_1,x_0}\,(x_2|x_1,x_0)$, indicating that the state x_k is a non-Markovian random process.

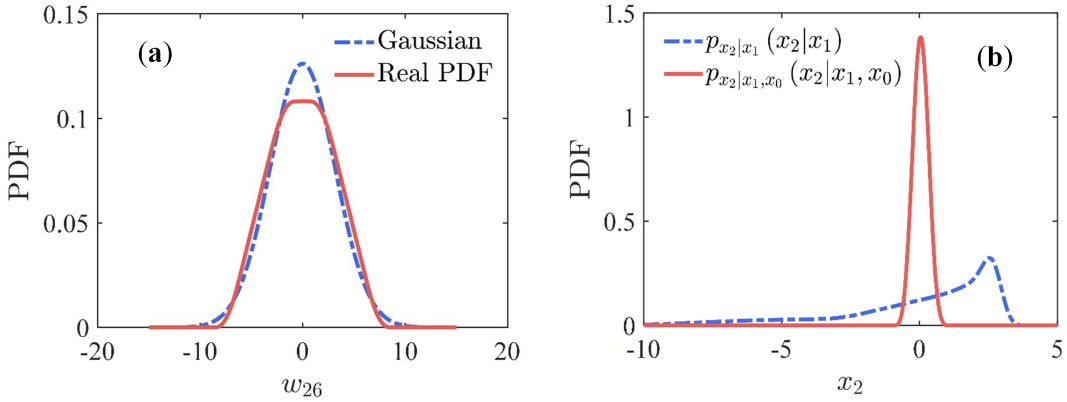

Figure 5. Non-Gaussian and non-Markovian characteristics of the problem: (a) PDF curve of w_{26}; and (b) Conditional PDFs $p_{x_2|x_1}(x_2|x_1)$ and $p_{x_2|x_1,x_0}(x_2|x_1,x_0)$ under conditions $x_0 = -0.5$ and $x_1 = -0.2$.

In this example, the proposed KCQF is used to analyze the nonlinear problem of the non-Gaussian, non-Markovian process. Different numbers of key conditions d are selected to compute $\overline{E_{\text{rms}}}$, with 50 MC runs being conducted, and the results are listed in Table 1. From Table 1, it can be observed that different d have a certain impact on the accuracy of KCQF. As d increases, $\overline{E_{\text{rms}}}$ shows a trend of decreasing first and then increasing, and $\overline{E_{\text{rms}}}$ of KCQF-3 is the best among all filters. When $d = 2$ and 3, E_{rms} is superior to that when $d = 1$, indicating that considering more key measurements leads to better estimation accuracy than considering only one key measurement. When $d \geq 4$, $\overline{E_{\text{rms}}}$ starts to increase slightly. This phenomenon again corroborates the point made in Section 3.2: If too many measurement values are considered, $p_{\beta_{k+1}}(\beta_{k+1})$ will become too small, leading to the numerator and the denominator in (3.14) and (3.15) becoming small, thereby increasing the error in numerical computation.

Some outstanding filters, such as the EKF, UKF, CKF, and the PF, are widely utilized. EKF, UKF, and CKF typically assume that the process noise is Gaussian; the PF generally presumes that the system states follow a Markov process. The problem addressed in this example involves a random process that is neither Gaussian nor Markovian, and neglecting these characteristics will lead to a significant decrease in prediction accuracy. To illustrate this point, we employ seven types of filters: EKF, UKF, CKF, PF-LVR, PF-STR, PGM-UT, and PGM, as comparative filters for computation. The results of these filters are also presented in Table 1. From Table 1, it can be observed that due to the neglect of the non-Gaussian and non-Markovian characteristics of the problem by EKF, UKF, and CKF, their estimation results are inaccurate, with very large $\overline{E_{\text{rms}}}$ values. Among the three methods, UKF has the best accuracy, but its $\overline{E_{\text{rms}}}$ is significantly larger than the $\overline{E_{\text{rms}}}$ of the proposed KCQF. PF-LVR, PF-STR, PGM-UT, and PGM can take into account the non-Gaussian characteristic of the problem. However, due to the neglect of the non-Markovian characteristics, the calculated $\overline{E_{\text{rms}}}$ for these methods is much larger than that of the proposed KCQF. We also compare the computation times of different filters, as shown in Table 1, where the computation times are the averages obtained from performing 50 MC runs. It can be observed that the computation times of different filters do not differ significantly. Summarizing from Figure 6 and Table 1, the proposed KCQF can take into account the non-Gaussian and non-Markovian characteristics of the filtering problem, achieving more accurate state estimations.

Table 1. Comparison of $\overline{E_{\text{rms}}}$ and CPU times calculated by different filters.

Filter	KCQF-1	KCQF-2	KCQF-3	KCQF-4	KCQF-5	KCQF-6	KCQF-7
$\overline{E_{\text{rms}}}$	4.7965	2.1213	1.8797	2.2884	2.3061	2.4685	2.5653
CPU times (s)	8.81E−04	8.92E−04	8.64E−04	8.87E−04	8.74E−04	9.70E−04	1.10E−03
Filter	PF-RR	PF-SR	PGM-UT	PGM	EKF	UKF	CKF
$\overline{E_{\text{rms}}}$	5.1707	5.5060	6.0641	6.2181	22.2849	8.7809	22.83
CPU times (s)	1.50E−03	1.50E−03	3.80E−02	3.49E−02	9.50E−04	8.08E−04	1.10E−03

We further plot the convergence curves of KCQF-2, KCQF-3, and KCQF-4 with different numbers of samples in Figure 6. It is not difficult to see from Figure 6 that when $N_{\text{s}} = 50$, $\overline{E_{\text{rms}}}$ calculated by KCQF-3 is less than 2. As the number of samples gradually increases, $\overline{E_{\text{rms}}}$ obtained from KCQF-2, KCQF-3, and KCQF-4 gradually decrease in an oscillatory manner and converge to $\overline{E_{\text{rms}}} = 1.75$.

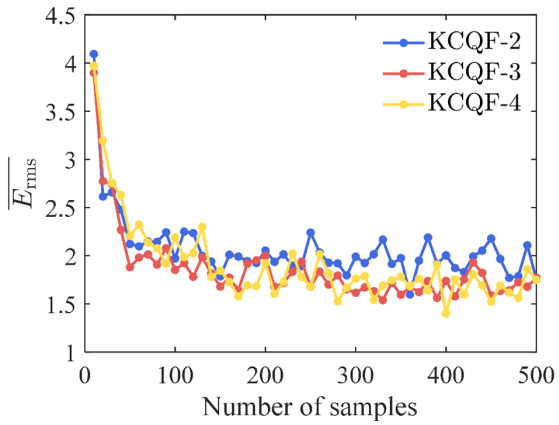

Figure 6. Convergence curves of KCQF-2, KCQF-3, and KCQF-4.

4.3. Time-varying noise navigation example

In practical applications, noise characteristics often exhibit significant time-varying behavior due to evolving system dynamics and environmental conditions [44, 45]. To investigate time-varying noise scenarios, we employ the continuous white noise acceleration model in a two-dimensional Cartesian coordinate system [46, 47] to validate the performance of the KCQF algorithm, where process noise and observation noise exhibit temporal slow-variation characteristics. The state vector $\mathbf{X}_k = [x_k \quad y_k \quad \dot{x}_k \quad \dot{y}_k]^{\text{T}}$ represents target position and velocity. The kinematic process is assumed to be monitored in real-time by the Global Navigation Satellite System (GNSS), with position signals serving as observations. The state and measurement equations are as follows:

$$\begin{cases} \mathbf{X}_k = \mathbf{F}_k \mathbf{X}_{k-1} + \omega_k \\ \mathbf{Y}_k = \mathbf{H}_k \mathbf{X}_k + \mathbf{v}_k \end{cases}, \tag{4.7}$$

where $\mathbf{F}_k$ and $\mathbf{H}_k$ denote the state transition matrix and the observation matrix, respectively, with complete definitions provided in [46]. The process noise and observation noise obey the following statistical characteristics: $\omega_k \sim \mathcal{N}(0, \mathbf{Q}_k^{\text{nominal}})$ and $v_k \sim \mathcal{N}(0, \mathbf{R}_k^{\text{nominal}})$. Considering the slow

temporal variation of noise characteristics, the time-varying process noise covariance matrix (PNCM) and measurement noise covariance matrix (MNCM) are defined as:

$$\begin{aligned}\mathbf{Q}_k^{\text{ture}} &= (3.5 + 0.5\cos(\frac{\pi k}{K}))\sigma_\omega^2 \begin{bmatrix} \frac{\Delta t^2}{3}\mathbf{I}_2 & \frac{\Delta t^2}{2}\mathbf{I}_2 \\ \frac{\Delta t^2}{2}\mathbf{I}_2 & \Delta t\mathbf{I}_2 \end{bmatrix}, \\ \mathbf{R}_k^{\text{ture}} &= (0.2 + 0.01\cos(\frac{\pi k}{K}))\sigma_v^2 \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}\end{aligned} \tag{4.8}$$

where the noise parameters are configured as $\sigma_\omega^2 = 3$, $\sigma_v^2 = 64\,\text{m}^2$; $\mathbf{I}$ with subscript denotes the identity matrix of specified dimensions; The simulation duration is $K = 2000\,\text{s}$ with sampling interval $\Delta t = 1\,\text{s}$. Furthermore, the nominal PNCM and MNCM are set as $\mathbf{Q}_k^{\text{nominal}} = \alpha\mathbf{I}_4$ and $\mathbf{R}_k^{\text{nominal}} = \beta\mathbf{I}_2$, respectively, during simulation, where α and β represent prior confidence parameters for tuning initial fixed noise covariance.

To evaluate the precision of the KCQF, we utilize the root mean square error E_{rms} for position and velocity and the averaged error $\overline{E_{\text{rms}}}$, as defined in Subsection IV.A.

Case1. Gaussian and Markovian system. To validate the accuracy of the KCQF algorithm in addressing time-varying noise, comparative evaluations are conducted against: (1) The Kalman filter using true PNCM and MNCM (KFTCM), (2) the Kalman filter using nominal PNCM and MNCM (KFNCM), and (3) other advanced adaptive filtering algorithms with nominal PNCM and MNCM, including VBAFK [48], N-VBAFK [46], and ST-VBAFK [49]. KFTCM, which utilizes true PNCM and MNCM, is generally adopted as the baseline solution. The prior confidence parameters are configured as $\alpha = 4$ and $\beta = 36$. Considering inherent sensor stochastic errors, 500 Monte Carlo runs ensure statistical reliability.

Figure 7 illustrates the E_{rms} of position and velocity for different filtering algorithms. Table 2 presents E_{rms} between algorithmic estimates and ground truth values, along with relative errors compared to the KFTCM baseline. As shown in the figure, compared to VBAFK and KFNCM, the E_{rms} values of the other three algorithms for position and velocity converge rapidly to levels close to those of KFTCM. Notably, the KCQF algorithm achieves the smallest positional E_{rms} among all methods. Overall, the KCQF algorithm demonstrates superior accuracy in handling time-varying noise. Compared to KFNCM and VBAFK, KCQF improves positional $\overline{E_{\text{rms}}}$ by 14.48% and 6.16%, and enhances velocity $\overline{E_{\text{rms}}}$ by 14.23% and 8.40%. These results conclusively demonstrate that the KCQF algorithm exhibits high precision and robust performance in addressing time-varying noise problems.

Table 2. $\overline{E_{\text{rms}}}$ and relative errors of KFTCM for position and velocity across algorithms.

Filter	KFTCM	KFNCM	VBAFK	N-VBAKF	ST-VBAKF	KCQF
Pos $\overline{E_{\text{rms}}}$	4.312	5.179/20.1%	4.802/11.4%	4.686/7.9%	4.654/7.9%	4.524/4.9%
Vel $\overline{E_{\text{rms}}}$	4.735	5.739/21.2%	5.446/15.0%	4.956/4.7%	5.035/6.3%	5.024/6.1%

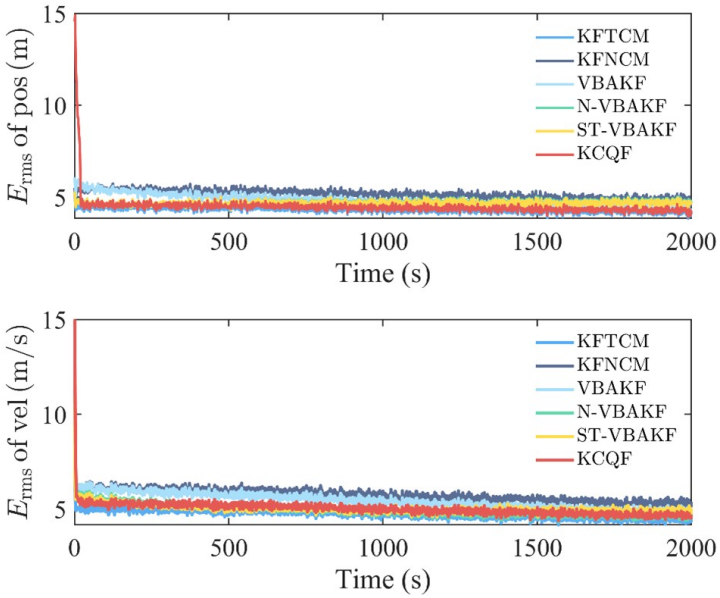

Figure 7. E_{rms} of position and velocity for different filtering algorithms.

Case2. Non-Gaussian and non-Markovian system. Conventional navigation filtering algorithms typically assume that process noise is a Markovian white noise process. This case involves actual wind field disturbances, which are characterized by significant temporal correlation and spatial coherence, belonging to a typical non-Markov process [50]. Turbulence, manifested as three-dimensional random vortex motions embedded within the background wind field, has statistical properties that serve as key parameters for characterizing the dynamic structure of the wind field. To more realistically simulate the dynamic behavior of moving vehicles within turbulent environments, we introduce an atmospheric turbulence model based on the Kaimal spectrum. This model generates physically realistic non-Markovian and non-Gaussian process noise.

The assumed process noise utilizes the Kaimal spectrum model to simulate non-Markovian noise induced by wind field disturbances. The Kaimal spectrum model is a statistical model employed to characterize wind speed time series under meteorological conditions, and its power spectral density (PSD) expression is given by:

$$S(f) = 4\sigma^2 L/U/(1 + 6fL/U)^{5/3}, \tag{4.9}$$

where $S(f)$ is the longitudinal wind speed power spectral density, f denotes frequency, $\sigma = 2.5$ is the root mean square turbulence intensity of the longitudinal wind speed, $L = 80\,\text{m}$ represents the longitudinal turbulence integral scale, and $U = 1.2\,\text{m/s}$ is the mean wind speed at hub height. The wind velocity field can be synthesized using the following formula [51]:

$$\begin{aligned} \Phi_{ij}(x, y, t) &= \cos\left[2\pi f_i t - \frac{2\pi f_i}{U}\left(x\cos\theta_j + y\sin\theta_j\right) + \varepsilon_{ij}\right] \\ \eta(x, y, t) &= \sum_{i=1}^{n}\sum_{j=1}^{m}\sqrt{2S(f_i)\Delta f}\,\Phi_{ij}(x, y, t) \end{aligned}, \tag{4.10}$$

where θ and ε represent directional angle component and a random phase, respectively. Considering the credibility of the results and the computational time, the Monte Carlo simulation frequency is set to 20 times. Moreover, since most of the filtering algorithms are constrained by the Markovian and Gaussian assumptions, only KFTCM is selected as the baseline solution to demonstrate the performance of the KCQF algorithm.

Figure 8 shows the E_{rms} of position and velocity for the KFTCM and KCQF algorithms. It can be seen that under non Markovian and non-Gaussian process noise, the E_{rms} of position and velocity for the KCQF algorithm can quickly converge to values close to those of KFTCM, and the results are similar to those of the KFTCM algorithm with real noise. Table 3 also lists the E_{rms} of position and velocity for the KFTCM and KCQF algorithms, as well as the relative errors compared to KFTCM. As can be seen from the table, the KCQF algorithm can maintain high accuracy in dealing with time-varying noise problems under non-Markovian and non-Gaussian process noise conditions.

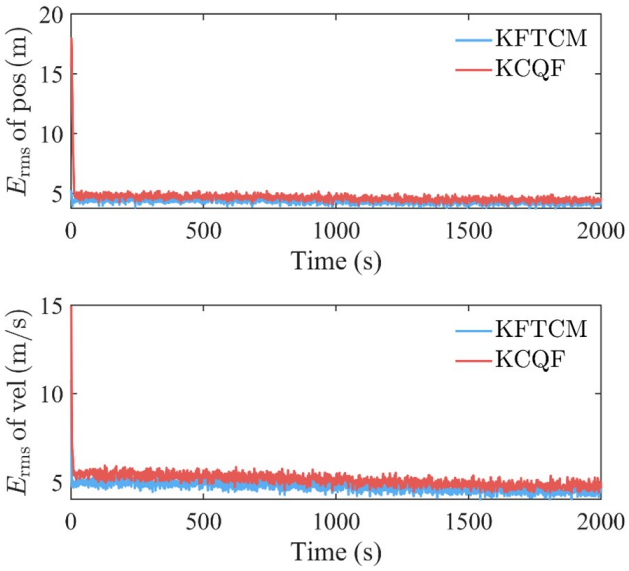

Figure 8. E_{rms} of position and velocity for different filtering algorithms.

Table 3. $\overline{E_{\text{rms}}}$ of position and velocity for KFTCM and KCQF algorithms and relative errors compared to KFTCM.

Filter	KFTCM	KCQF
Position $\overline{E_{\text{rms}}}$	4.306	4.634/7.6%
Velocity $\overline{E_{\text{rms}}}$	4.728	5.128/8.5%

5. Conclusion and outlook

In this paper, we study the estimation of state given measurement conditions. We first theoretically observe, without involving approximations such as Gaussian distributions or Markov processes, that when considering all measurement conditions, the numerator and denominator of the state estimation

quotient-form expression tend to zero simultaneously over time, making numerical calculations unstable. This observation motivates us to propose the idea of estimating the state based on key measurement conditions rather than all measurement conditions. According to this idea, using the principle of probability conservation, we have derived the corresponding integral quotient-form expressions for the conditional PDF, mean, and variance of states based on key measurement conditions, and employed the MC method to calculate these expressions, thereby constructing a key conditional quotient filter (KCQF). KCQF uses key conditions to estimate states, avoiding the numerical difficulty that the numerator and denominator tend to zero as time increases. Three numerical examples were given to demonstrate the superior estimation performance of KCQF, compared to other filters.

In the future, we plan to: 1) Extend the concept of key measurement conditions to other filters that require sampling, such as particle filter; 2) attempt to apply the proportional-integral filter to the KCQF framework; 3) and extend our research to address filtering problems involving uncertain noises whose PDF is unknown and only its interval bounds are known. In some scenarios, the lack of prior statistical knowledge makes it difficult to accurately determine the probability density information of uncertain parameters in detection and navigation systems [52, 53]. Therefore, we will extend the KCQF to the interval framework and investigate multi source navigation filtering algorithms that can effectively evaluate and handle the above uncertainty factors.

Author contributions

Yue Zeng: validation, writing–original draft; Yuelin Zhao: validation, writing–original draft; Feng Wu: Supervision, conceptualization, Writing – review & editing; Li Zhu: validation. All authors have read and agreed to the published version of the manuscript.

Use of Generative-AI tools declaration

The authors declare they have not used Artificial Intelligence(AI) tools in the creation of this article.

Acknowledgments

This work was supported by Excellent Research Group Project of the National Natural Science Foundation of China (No. 62388101) and the National Natural Science Foundation of China (Nos. 12372190,12432005).

Conflict of interest

The authors declare no conflicts of interest in this paper.

References

1. H. Ye, W. Wu, Y. Zhang, K. Li, S. Wu, Receiving filter design for the distant sidelobe suppression of RMWFPA radar, *IEEE T. Aero. Elec. Sys.*, **61** (2025), 13333–13347. https://doi.org/10.1109/TAES.2025.3578060

2. M. J. Lindenfeld, Mismatched filters for incoherent pulse compression in laser radar, *IEEE T. Aero. Elec. Sys.*, **57** (2021), 1252–1260. https://doi.org/10.1109/TAES.2020.3038256

3. H. Liu, X. Sun, J. Yang, M. Xu, S. Bai, Skewed unscented Kalman filter using Gaussian sum, *IEEE T. Aero. Elec. Sys.*, **61** (2025), 3917–3935. https://doi.org/10.1109/TAES.2024.3501237

4. T. Badar, S. Särkkä, Z. Zhao, A. Visala, Rao–blackwellized particle filter using noise adaptive Kalman filter for fully mixing state-space models, *IEEE T. Aero. Elec. Sys.*, **60** (2024), 6972–6982. https://doi.org/10.1109/TAES.2024.3409644

5. L. Guo, S. Hu, J. Zhou, X. R. Li, Recursive nonlinear filtering via Gaussian approximation with minimized kullback–leibler divergence, *IEEE T. Aero. Elec. Sys.*, **60** (2024), 965–979. https://doi.org/10.1109/TAES.2023.3330952

6. L. Zhu, Y. Zhao, F. Wu, A key condition quotient method for heat conduction conditional uncertainty propagation, *Int. Commun. Heat Mass*, **171** (2026), 110207. https://doi.org/10.1016/j.icheatmasstransfer.2025.110207

7. S. Wei, S. Chen, X. Dong, Z. Peng, W. Zhang, Parametric identification of time-varying systems from free vibration using intrinsic chirp component decomposition, *Acta Mech. Sin.*, **36** (2020), 188–205. https://doi.org/10.1007/s10409-019-00905-7

8. Y. Zhao, F. Wu, Key conditional quotient of random finite element model under measurement conditions, *Comput. Method. Appl. M.*, **440** (2025), 117943. https://doi.org/10.1016/j.cma.2025.117943

9. R. E. Kalman, A new approach to linear filtering and prediction problems, *J. Basic Eng.*, **82** (1960), 35–45. https://doi.org/10.1115/1.3662552

10. X. Song, M. Zhang, W. X. Zheng, Z. Liu, Maximum correntropy kalman filter for linear discrete-time systems with intermittent observations and non-gaussian noise, *IEEE T. Circuits-II*, **71** (2024), 3246–3250. https://doi.org/10.1109/TCSII.2024.3357588

11. A. H. Jazwinski, Introduction to filtering theory, In: *Stochastic processes and filtering theory*, Academic Press, (1970), 142–161. https://doi.org/10.1016/S0076-5392(09)60374-X

12. L. D. Avendano-Valencia, L. E. Avendano, J. M. Ferrero, G. Castellanos-Dominguez, Improvement of an extended Kalman filter power line interference suppressor for ECG signals, In: *2007 Computers in Cardiology*, 2007, 553–556. http://doi.org/10.1109/CIC.2007.4745545

13. G. Battistelli, L. Chisci, Stability of consensus extended Kalman filter for distributed state estimation, *Automatica*, **68** (2016), 169–178. https://doi.org/10.1016/j.automatica.2016.01.071

14. H. Hajimolahoseini, M. R. Taban, H. R. Abutalebi, Improvement of extended Kalman filter frequency tracker for nonstationary harmonic signals, In: *2008 international symposium on telecommunications*, 2008, 592–597. https://doi.org/10.1109/ISTEL.2008.4651370

15. S. J. Julier, J. K. Uhlmann, New extension of the Kalman filter to nonlinear systems, In: *Proceedings Volume 3068, Signal Processing, Sensor Fusion, and Target Recognition VI*, 1997. https://doi.org/10.1117/12.280797

16. A. Meyer Sjøberg, O. Egeland, Lie algebraic unscented Kalman filter for pose estimation, *IEEE T. Automat. Contr*, **67** (2022), 4300–4307. https://doi.org/10.1109/TAC.2021.3121247

17. S. J. Julier, The scaled unscented transformation, In: *Proceedings of the 2002 American Control Conference (IEEE Cat. No. CH37301)*, **6** (2002), 4555–4559. https://doi.org/10.1109/ACC.2002.1025369

18. M. R. Morelande, B. Ristic, Reduced sigma point filtering for partially linear models, In: *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, 2006. https://doi.org10.1109/ICASSP.2006.1660584

19. I. Arasaratnam, S. Haykin, Cubature Kalman filters, *IEEE T. Automat. Contr.*, **54** (2009), 1254–1269. https://doi.org/10.1109/TAC.2009.2019800

20. N. J. Gordon, D. J. Salmond, A. F. Smith, Novel approach to nonlinear/non-Gaussian Bayesian state estimation, In: *IEE Proceedings F (Radar and Signal Processing)*, **140** (1993), 107–113. https://doi.org/10.1049/ip-f-2.1993.0015

21. D. Huang, F. Wu, Y. Zhao, J. Yan, H. Zhang, Application of high-credible statistical results calculation scheme based on least squares Quasi-Monte Carlo method in multimodal stochastic problems, *Comput. Method. Appl. M.*, **418** (2024), 116576, https://doi.org/10.1016/j.cma.2023.116576

22. F. Wu, Y. Zhao, K. Zhao, W. Zhong, A multi-body dynamical evolution model for generating the point set with best uniformity, *Swarm Evol. Comput.*, **73** (2022), 101121, https://doi.org/10.1016/j.swevo.2022.101121

23. R. Van Der Merwe, A. Doucet, N. De Freitas, E. Wan, The unscented particle filter, In: *NIPS'00: Proceedings of the 14th International Conference on Neural Information Processing Systems*, **13** (2000), 563–569.

24. F. Wang, J. Zhang, B. Lin, X. Li, Two stage particle filter for nonlinear Bayesian estimation, *IEEE Access*, **6** (2018), 13803–13809. https://doi.org/10.1109/ACCESS.2018.2808922

25. W. Wei, S. Gao, Y. Zhong, C. Gu, G. Hu, Adaptive square-root unscented particle filtering algorithm for dynamic navigation, *Sensors*, **18** (2018), 2337. https://doi.org/10.3390/s18072337

26. M. Bolic, P. M. Djuric, S. Hong, New resampling algorithms for particle filters, In: *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03).*, 2003. https://doi.org/10.1109/ICASSP.2003.1202435

27. A. Daniyan, Y. Gong, S. Lambotharan, An improved resampling approach for particle filters in tracking, In: *2017 22nd International Conference on Digital Signal Processing (DSP)*, 2017. https://doi.org/10.1109/ICDSP.2017.8096095

28. T. Z. Lv, C. X. Zhao, H. F. Zhang, An improved fastslam algorithm based on revised genetic resampling and SR-UPF, *Int. J. Autom. Comput.*, **15** (2018), 325–334. https://doi.org/10.1007/s11633-016-1050-y

29. D. Lin, X. Chen, Y. Zheng, Z. Guo, Q. Zhang, S. Wang, Quaternion information filters with inaccurate measurement noise covariance: A variational Bayesian method, *IEEE T. Signal Proces.*, **73** (2025), 1367–1378. https://doi.org/10.1109/TSP.2025.3549023

30. D. M. Titterington, B. Wang, Convergence properties of a general algorithm for calculating variational Bayesian estimates for a normal mixture model, *Bayesian Anal.*, **1** (2006), 625–650. https://doi.org/10.1214/06-BA121 .

31. B. Yang, H. Wang, Z. Shi, Variational Bayesian and generalized maximum-likelihood based adaptive robust nonlinear filtering framework, *Signal Process.*, **215** (2024), 109271. https://doi.org/10.1016/j.sigpro.2023.109271

32. B. Chen, X. Liu, H. Zhao, J. C. Principe, Maximum correntropy Kalman filter, *Automatica*, **76** (2017), 70–77. https://doi.org/10.1016/j.automatica.2016.10.004

33. S. Peng, P. Cai, D. Lin, Y. Zheng, S. Wang, Cauchy–Gaussian maximum mixture correntropy Kalman filter with component-by-component construction, *Signal Process.*, **234** (2025), 110008. https://doi.org/10.1016/j.sigpro.2025.110008

34. W. Yu, D. Lin, Y. Zheng, S. Wang, Extended H∞ filtering in RKHS for nonlinear systems with uncertainty, *IEEE T. Circuits-II*, **72** (2025), 429–433. https://doi.org/10.1109/TCSII.2024.3522913

35. C. H. Tseng, S. F. Lin, D. J. Jwo, Robust Huber-based cubature Kalman filter for gps navigation processing, *J. Navigation*, **70** (2017), 527–546. https://doi.org/10.1017/S0373463316000692

36. G. Tan, T. Wang, A new result on H∞ filtering criterion based on improved techniques, *Circuits Syst. Signal Process*, **45** (2026), 977–994. https://doi.org/10.1007/s00034-025-03305-4

37. J. Li, J. B. Chen, Probability density evolution method for dynamic response analysis of structures with uncertain parameters, *Comput. Mech.*, **34** (2004), 400–409. https://doi.org/10.1007/s00466-004-0583-8

38. A. Rényi, *Probability theory*, Dover Publications, 2007.

39. D. Raihan, S. Chakravorty, Particle Gaussian mixture filters-I, *Automatica*, **98** (2018), 331–340. https://doi.org/10.1016/j.automatica.2018.07.023

40. M. Sun, M. E. Davies, I. K. Proudler, J. R. Hopgood, Adaptive Kernel Kalman filter, *IEEE T. Signal Process.*, **71** (2023), 713–726. https://doi.org/10.1109/TSP.2023.3250829

41. J. S. Liu, R. Chen, Sequential Monte Carlo methods for dynamic systems, *J. Am. Stat. Assoc.*, **93** (1998), 1032–1044. https://doi.org/10.1080/01621459.1998.10473765

42. G. Kitagawa, Monte Carlo filter and smoother for non-Gaussian nonlinear state space models, *J. Comput. Graph. Stat*, **5** (1996), 1–25. https://doi.org/10.1080/10618600.1996.10474692

43. F. WU, Y. ZHAO, Y. YANG, N. ZHOU, A new discrepancy for sample generation in stochastic response analyses of aerospace problems with uncertain parameters, *Chinese J. Aeronaut.*, **37** (2024), 192–211. https://doi.org/10.1016/j.cja.2024.09.044

44. C. Hide, T. Moore, M. Smith, Adaptive Kalman filtering algorithms for integrating GPS and low cost INS, Tn: *Plans 2004. Position location and navigation symposium (ieee cat. no. 04ch37556)*, 2004. https://doi.org/10.1109/PLANS.2004.1308998

45. M. J. Yu, INS/GPS integration system using adaptive filter for estimating measurement noise variance, *IEEE T. Aero. Elec. Sys.*, **48** (2012), 1786–1792. https://doi.org/10.1109/TAES.2012.6178100

46. Y. Huang, Y. Zhang, Z. Wu, N. Li, J. Chambers, A novel adaptive Kalman filter with inaccurate process and measurement noise covariance matrices, *IEEE T. Automat. Contr.*, **63** (2018), 594–601. https://doi.org/10.1109/TAC.2017.2730480

47. C. Pan, J. Gao, Z. Li, N. Qian, F. Li, Multiple fading factors-based strong tracking variational Bayesian adaptive Kalman filter, *Measurement*, **176** (2021), 109139. https://doi.org/10.1016/j.measurement.2021.109139

48. T. Ardeshiri, E. Özkan, U. Orguner, F. Gustafsson, Approximate Bayesian smoothing with unknown process and measurement noise covariances, *IEEE Signal Proc. Let.*, **22** (2015), 2450–2454. https://doi.org/10.1109/LSP.2015.2490543

49. F. M. Tan, J. J. Zhao, Strong tracking based variational bayesian adaptive kalman filtering algorithm, *Electron. Opt. Control*, **27** (2020), 12–16. In Chinese. https://doi.org/10.3969/j.issn.1671-637X.2020.01.003

50. F. Guo, D. Schlipf, P. W. Cheng, Evaluation of lidar-assisted wind turbine control under various turbulence characteristics, *Wind Energy Sci.*, **8** (2023), 149–171. https://doi.org/10.5194/wes-8-149-2023

51. F. Wu, L. Zhu, Y. Zhao, C. Ai, X. Wang, F. Cai, et al., Wave spectrum fitting with multiple parameters based on optimization algorithms and its application, *Ocean Eng.*, **312** (2024), 119073. https://doi.org/10.1016/j.oceaneng.2024.119073

52. Y. Yang, F. Wu, X. Wei, W. Hu, C. Li, S. Liang, Time-variant uncertainty modeling with limited samples: Interval process model dimension reduction construction method, *Eng. Struct.*, **350** (2026), 121932. https://doi.org/10.1016/j.engstruct.2025.121932

53. F. Wu, Y. Zhao, W. Hu, An interval analysis method based on pdf for unconditional/conditional uncertain structural responses, *Appl. Math. Model.*, **152** (2026), 116577. https://doi.org/10.1016/j.apm.2025.116577

AIMS Press