

Convolutional Bayesian Filtering

Wenhan Cao, Shiqi Liu, Chang Liu, Zeyu He,
Stephen S.-T. Yau, *Life Fellow, IEEE*, and Shengbo Eben Li, *Senior Member, IEEE*

Abstract—Bayesian filtering serves as a mainstream framework for state estimation in dynamical systems, yet its accuracy depends on the assumption that the underlying mathematical model perfectly matches the real system. In practice, unmodeled dynamics, disturbances, and structural simplifications often cause significant model mismatch, which conventional filters cannot explicitly handle. To address this limitation, we reformulate the probabilistic foundation of Bayesian filtering by relaxing the exact conditioning assumption in transition and measurement models, both of which are defined through conditional probabilities. By introducing an additional conditioning event that tolerates bounded discrepancies between the virtual and real variables, we reformulate conditional probability into an integral form analogous to convolution, leading to the concept of *convolutional conditional probability*. Building on this idea, we develop a generalized filtering framework termed *convolutional Bayesian filtering* (ConvBF), which modifies both prediction and update steps to explicitly accommodate model mismatch and offers flexible robustness through the choice of distance metrics. For systems with Gaussian noises and Euclidean distance metrics, ConvBF yields a family of robust Kalman filters through modified noise covariance matrices. For general systems with cross-entropy distance, we further propose an efficient exponential density rescaling approach that approximates convolutional effects by rescaling probability densities into fractional powers. Finally, the effectiveness of ConvBF is demonstrated by reformulating classical filtering algorithms, including the Kalman filter, extended Kalman filter, and unscented Kalman filter, into their convolutional counterparts.

Index Terms—Bayesian filtering; conditional probability; convolution; model mismatch

I. INTRODUCTION

Accurately estimating the state value of dynamic systems is a crucial task in science and engineering, such as robotics, power systems, aerospace engineering, and manufacturing. Since the 1960s, Bayesian filtering has become a principled framework for optimal state estimation [1]. The essence of this framework is to find a balance between uncertain system

model and noisy state measurement. Its associated algorithm iteratively updates the probability density function of system state using the prior from the last step and the likelihood of the new observation. Afterward, typical estimation criteria like minimum mean-square error and maximum a posteriori are utilized to acquire the optimal point estimate.

In mathematics, Bayesian filtering relies on two conditional probabilities: transition probability and output probability. The transition probability characterizes the temporal evolution of the system state, and the output probability depicts the relationship between noisy measurement and ground truth state. To incorporate the information of those two probabilities, each iteration of Bayes filter is composed of two steps: prediction and update. The prediction step employs *total probability rule* to integrate the product of transition probability and the state distribution of previous time to obtain the prior distribution. The update step employs *Bayes' law* to calculate the posterior distribution by adding information of the current measurement, where the output probability is used as a likelihood term.

Although Bayesian filtering provides a seemingly rigorous framework for state estimation, its validity depends on the assumption that the mathematical model and the real system are perfectly matched. In practice, however, models are inevitably approximate. Unmodeled dynamics, external disturbances, and simplifications introduced during modeling often create discrepancies between the assumed model and reality. Such model mismatch can significantly degrade estimation performance [2]–[4]. A central challenge, therefore, is to generalize Bayesian filtering so that it can explicitly account for these mismatches, rather than relying on idealized assumptions.

To address this challenge, we propose to relax the strict conditioning assumption in conditional probabilities that underlies standard Bayesian filtering. Instead of enforcing exact adherence to the nominal transition and measurement models, we introduce an additional conditioning event into both transition and output probabilities: the discrepancy between the nominal variables and the real variables must lie within a specified distance metric. This modification transforms conditional probability into a special integral form resembling a convolution operation, which we term *convolutional conditional probability*.

Based on this definition, we reformulate both transition and output probabilities into convolutional forms that directly encode permissible discrepancies between the model and the real system. Collectively, these extensions lead to a generalized framework referred to as *convolutional Bayesian filtering* (ConvBF). Standard Bayesian filtering is recovered as a special case when the distance metric reduces to the Dirac delta function, while robustness against model mismatch is achieved by tailoring the metric appropriately. Within this framework,

This work is supported by National Science and Technology Major Project (No. 2025ZD1606200), NSF China under 52221005, the National Natural Science Foundation of China (No. 42450242), and Tsinghua University Education Foundation. It is also partially supported by Tsinghua University-Toyota Joint Research Center for AI Technology of Automated Vehicle. All correspondence should be sent to S. Eben Li.

Wenhan Cao, Shiqi Liu, and Zeyu He are with the School of Vehicle and Mobility, Tsinghua University, Beijing, 100084, China. Shengbo Eben Li is with the School of Vehicle and Mobility and College of Artificial Intelligence, Tsinghua University, Beijing, 100084, China. (e-mail: wenhan_cao@outlook.com, (liushiqi19, hezeyu1549, lisb04)@gmail.com).

Chang Liu is with the School of Advanced Manufacturing and Robotics, Peking University, Beijing 100871, China. (e-mail: changliu-coe@pku.edu.cn).

Stephen S.-T. Yau is with the Department of Mathematical Sciences, Tsinghua University, Beijing, 100084, China. (e-mail: yau@uic.edu).

classical Bayesian filtering algorithms can be reformulated into their convolutional counterparts. For systems with Gaussian noises, ConvBF admits analytical convolution forms, which naturally give rise to a robust family of Kalman filters. For general systems, we further propose an efficient exponential density rescaling technique that approximates convolutional operations by modulating probability densities into fractional powers.

The remainder of this paper is organized as follows: Section II discusses related works. Section III introduces the definition of convolutional conditional probability. Section IV presents the ConvBF framework. Section V provides simulation results and Section VI concludes the paper.

II. RELATED WORKS

1) *Robust Bayesian filtering*: A substantial body of research has sought to improve the robustness of Bayesian filtering in the presence of model mismatch. One prominent line of work follows the adaptive filtering paradigm, where hierarchical priors are placed on uncertain model parameters, and both the system states and parameters are estimated jointly [5], [6]. This approach offers flexibility in capturing model uncertainties, but the closed-form state estimation of the augmented parameter-state system is generally intractable. Consequently, approximate inference techniques such as factorized variational Bayes are often employed to obtain fixed-point solutions. Compared to our proposed method, these methods generally lack formal robustness guarantees (See Section VI-G in supplementary material) such as the bounded influence functions.

Another important direction reformulates the filtering problem as a robust regression task, employing techniques such as Huber loss [2], Student-t loss [3], or correntropy-based loss functions [4] to mitigate the effect of model discrepancies. Although such methods enhance robustness, they typically sacrifice the conjugacy and computational efficiency of Gaussian filters. Moreover, treating filtering as a regression problem often neglects the uncertainty quantification that lies at the core of Bayesian filtering. In contrast, our proposed method remains the Bayesian problem formulation. By explicitly addressing robustness within a new probabilistic framework, our approach preserves the conjugate structure under Gaussian noises and maintains computational efficiency comparable to standard Gaussian filtering algorithms.

2) *Robust Bayesian inference*: A parallel line of research in the machine learning community has explored robustness against model misspecification within the broader Bayesian inference paradigm [7]. The basic idea is to generalize the Bayesian posterior by replacing the likelihood with alternative discrepancy measures between models and data. Typical methods include β -divergence based posteriors [8], maximum mean discrepancy posteriors [9], kernel Stein discrepancy approaches [10], and α -posteriors [11], [12]. Such approaches are often referred to collectively as generalized Bayesian inference [7]. Although derived from a different startpoint, the update step of our proposed framework can be interpreted as a special case of computing the α -posterior in a sequential setting.

III. CONVOLUTIONAL CONDITIONAL PROBABILITY

As discussed before, Bayesian filtering is built upon two pillars: total probability rule (1a) and Bayes' law (1b). Both

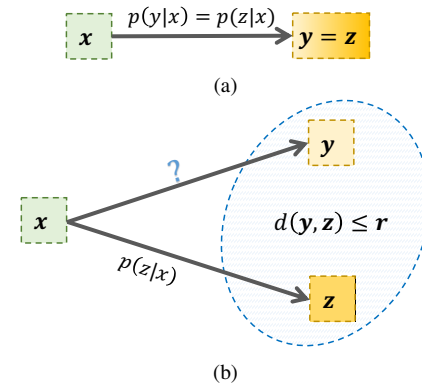


Fig. 1. (a) y and z are identical ($y = z$). (b) The distance between y and z is bounded ($d(y, z) \leq r$).

of them rely on how to handle the conditional probability $p(y|x)$, which is the probability distribution of y when x is known to be a particular value. For the intuition regarding discrete random variables, it describes the probability of the event $\{y = y\}$ occurring, given the event $\{x = x\}$. According to this interpretation, the total probability rule is

$$p(y) = \int p(y|x)p(x) dx, \quad (1a)$$

and the Bayes' law is

$$p(x|y) = \frac{p(x)p(y|x)}{\int p(x)p(y|x) dx}. \quad (1b)$$

Note that we use boldface to denote a random variable, such as x , and normal font to denote the realization of this variable, such as x . Previously, the explicit form of the conditional probability $p(y|x)$ is assumed known directly in Bayes filter design. One may be interested in what will happen if the conditional probability $p(y|x)$ is unknown. This question leads us to conceive a specific form of conditional probability which models uncertainty via inequality constraints between random variables.

Given three random variables x , y , and z , where the information of $p(z|x)$ is known and certain constraints exist between y and z , our objective is to compute $p(y|x)$ by leveraging these constraints. In the case that y and z are equal, i.e., $y = z$, we have $p(y|x) = p(z|x)$, as shown in Fig. 1(a). Conversely, if y is not equal to z and their difference is bounded by an inequality function, we can define a convolutional version of conditional probability, as shown in Fig. 1(b).

Definition 1 (Convolutional Conditional Probability):

Given $p(z|x)$, if y and z are conditionally independent given x , then the convolutional conditional probability is defined as

$$p_c(y|x) \triangleq p(y|x, d(y, z) \leq r).$$

In this definition, $d(\cdot, \cdot)$ is a distance metric between two random variables, and r is a threshold random variable with known cumulative distribution function $F(r)$. Here, r is assumed to be independent of x, y , and z , and its distribution is supported on the interval $[0, +\infty)$.

The definition of $p_c(y|x) \triangleq p(y|x, d(y, z) \leq r)$ can be rigorously formulated through measure-theoretic principles, in particular by constructing conditional expectations via

the Radon–Nikodym theorem and, when absolute continuity holds, identifying the corresponding density. More details can be found in Section VI-A of the supplementary material. The calculation of convolutional conditional probability $p_c(y|x)$ is summarized in the subsequent proposition:

Proposition 1 (Convolutional Conditional Probability):
The convolutional conditional probability satisfies

$$p_c(y|x) = \frac{p_{\text{pri}}(y|x) \int_z (1 - F(d(y, z))) p(z|x) dz}{\int_y p_{\text{pri}}(y|x) \int_z (1 - F(d(y, z))) p(z|x) dz dy} \quad (2)$$

$$\propto p_{\text{pri}}(y|x) \int_z (1 - F(d(y, z))) p(z|x) dz.$$

Here, $p_{\text{pri}}(y|x)$ is the prior of $p(y|x)$.

Proof: According to Bayes' law, we have

$$p_c(y|x) = \frac{P(d(\mathbf{y}, \mathbf{z}) \leq \mathbf{r}|x, y) p_{\text{pri}}(y|x)}{\int_y P(d(\mathbf{y}, \mathbf{z}) \leq \mathbf{r}|x, y) p_{\text{pri}}(y|x) dy}. \quad (3)$$

The likelihood term $P(d(\mathbf{y}, \mathbf{z}) \leq \mathbf{r}|x, y)$ in (3) is simplified as

$$\begin{aligned} & P(d(\mathbf{y}, \mathbf{z}) \leq \mathbf{r}|x, y) \\ &= P(d(y, \mathbf{z}) \leq \mathbf{r}|x, y) \\ &= \int_z (1 - F(d(y, z))) p(z|x, y) dz \quad (4) \\ &= \int_z (1 - F(d(y, z))) p(z|x) dz. \end{aligned}$$

Substituting (4) into (3), we have (2). Note that the final equation of (4) holds due to the conditional independence of y and $\mathbf{z}$, given $\mathbf{x}$. ■

Proposition 1 provides an explicit formulation for computing the convolutional conditional probability. Nevertheless, specifying a prior distribution for $p(y|x)$ remains a challenge. In practice, all information about $p(y|x)$ is implicitly encoded in the known distribution $p(z|x)$ and the inequality constraints between $\mathbf{y}$ and $\mathbf{z}$, which motivates the introduction of convolutional conditional probability. Beyond this, no additional prior information is typically available.

A natural choice when prior information is limited or intentionally excluded is to adopt a noninformative prior, following the objective Bayesian perspective [13]. A classical choice is $p_{\text{pri}}(y|x) = \text{Const} > 0$, which cancels out the prior term in Proposition 1, thereby simplifying the computation to

$$p_c(y|x) = \frac{\int_z (1 - F(d(y, z))) p(z|x) dz}{\int_y \int_z (1 - F(d(y, z))) p(z|x) dz dy}. \quad (5)$$

One may recognize that such a uninformative prior does not integrate to 1, and lead to the worry that it is not a proper probability distribution. Nevertheless, this issue can be resolved by showing that the resulting posterior distribution is proper and can be interpreted as the limiting case of posteriors derived from proper priors. To clarify this further, consider a Gaussian prior $\mathcal{N}(y; 0, \sigma^2 I)$, which is independent of x . Under this prior, the posterior becomes

$$\begin{aligned} p_c(y|x) &= \frac{P(d(\mathbf{y}, \mathbf{z}) \leq \mathbf{r}|x, y) \mathcal{N}(y; 0, \sigma^2 I)}{\int_y P(d(\mathbf{y}, \mathbf{z}) \leq \mathbf{r}|x, y) \mathcal{N}(y; 0, \sigma^2 I) dy} \\ &= \frac{P(d(\mathbf{y}, \mathbf{z}) \leq \mathbf{r} | x, y) \cdot \exp(-\|y\|^2/(2\sigma^2))}{\int_y P(d(\mathbf{y}, \mathbf{z}) \leq \mathbf{r} | x, y) \cdot \exp(-\|y\|^2/(2\sigma^2)) dy}. \end{aligned}$$

As $\sigma^2 \rightarrow \infty$, the Gaussian prior becomes increasingly flat, and its influence on the posterior diminishes. In this limit, the exponential term $\exp(-\|y\|^2/(2\sigma^2))$ approaches 1, and the posterior simplifies to

$$\lim_{\sigma^2 \rightarrow \infty} p_c(y|x) = \frac{P(d(\mathbf{y}, \mathbf{z}) \leq \mathbf{r} | x, y)}{\int P(d(\mathbf{y}, \mathbf{z}) \leq \mathbf{r} | x, y) dy},$$

which is exactly the form obtained in (5) when assuming a constant prior.

Remark 1: The notion of a noninformative prior remains a topic of ongoing debate within the statistical community. It sits at the core of long-standing discussions between subjective and objective Bayesian perspectives. A deeper exploration of these issues can be found in [13].

Remark 2: The calculation of convolutional conditional probability $p_c(y|x)$ resembles convolution operation. In the convolution operation, a kernel is applied over an input space to generate a modified output. Here, $1 - F(d(y, z))$ serves as the kernel function, which is a weighting coefficient of $p(z|x)$ based on the distance between $\mathbf{y}$ and $\mathbf{z}$.

The convolutional conditional probability $p_c(y|x)$ represents a specific form of conditional probability, which measures the probability of the event $\{\mathbf{y} = y\}$, conditioned on two events $\{\mathbf{x} = x\}$ and $\{d(\mathbf{y}, \mathbf{z}) \leq \mathbf{r}\}$. Compared to standard definition, convolutional conditional probability has the third event which describes the upper bound between two random variables. Under this new definition, one can substitute $p(y|x)$ with $p_c(y|x)$ in (1a) and (1b) to construct two new rules:

$$p_c(y) = \int_x p_c(y|x) p(x) dx, \quad (6a)$$

$$p_c(x|y) = \frac{p(x) p_c(y|x)}{\int_x p(x) p_c(y|x) dx}. \quad (6b)$$

Here, $p_c(y) \triangleq p(y|d(\mathbf{y}, \mathbf{z}) \leq \mathbf{r})$ and $p_c(x|y) \triangleq p(x|y, d(\mathbf{y}, \mathbf{z}) \leq \mathbf{r})$ denote the probability distributions that implicitly condition on the event $\{d(\mathbf{y}, \mathbf{z}) \leq \mathbf{r}\}$. In this way, (6a) and (6b) can be interpreted as generalized versions of (1a) and (1b). The classical rules in (1) correspond to the special case where this event reduces to $\{\mathbf{y} = \mathbf{z}\}$. The subsequent proposition demonstrates that, under the limiting condition, the generalized rules converge to the standard total probability rule (1a) and Bayes' law (1b).

Proposition 2 (Limiting Property): Choose $\mathbf{r}$ as the exponential distribution, i.e., $\mathbf{r} \sim \text{Exp}(\lambda)$ and $d(y, z) = \|y - z\|^2/\sigma^2$, where $\lambda > 0$ is exponential parameter, and $\sigma > 0$ is scale parameter. As $\sigma \rightarrow 0$, $p_c(y|x)$ reduces to $p(y|x)$, (6a) reduces to total probability rule (1a), and (6b) reduces to Bayes' law (1b).

The proof of this proposition can be found in Section VI-B of the supplementary material. Proposition 2 elucidates that the kernel function converges to a Dirac delta function at y as the scale parameter tends to zero. As a result, the convolutional conditional probability reduces to its standard version. For finite values of scale parameter, this new definition considers a controllable amount of uncertainty governed by the scale parameter, offering an extension to the previous one.

IV. CONVOLUTIONAL BAYESIAN FILTERING

In this section, we demonstrate how model mismatches in filtering problem can be explicitly addressed using convolutional conditional probability. This is achieved by conditioning

on an additional event representing the error bound between the system model and the real system. Further, by substituting the total probability rule and Bayes' law with (6a) and (6b) in the Bayes filter, we can establish a more generalized filtering framework called ConvBF.

A. Uncertain Hidden Markov Model with Mismatch

The essence of Bayesian filtering is to find a balance between the stochasticities of state transition and state observation. The stochasticity of the former comes from inherent randomness in the environment dynamics while that of the latter comes from sensor noises. These two stochastic processes are typically represented by hidden Markov model (HMM):

$$\mathbf{x}_0 \sim p_0(\mathbf{x}_0), \quad \mathbf{x}_t \sim p(\mathbf{x}_t|\mathbf{x}_{t-1}), \quad \mathbf{y}_t \sim p(\mathbf{y}_t|\mathbf{x}_t). \quad (7)$$

Here, $\mathbf{x}_t \in \mathbb{R}^n$ is the system state and $\mathbf{y}_t \in \mathbb{R}^m$ is the corresponding measurement. Besides, p_0 denotes the probability distribution of the initial state $\mathbf{x}_0$, $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ represents the transition probability, and $p(\mathbf{y}_t|\mathbf{x}_t)$ is the output probability.

The standard HMM implicitly assumes that the real system is perfectly modelled, i.e., (7) is an exact description of system dynamics and measurement sensors. However, perfect information about transition or output probabilities is often unattainable due to parametric variation, unmodeled dynamics or external disturbances. In other words, there must be some model errors in engineering practice. This error can lead to significant accuracy degradation in state estimation if not properly considered. To build an HMM with model mismatch, we have to distinguish two kinds of states: the real state and the virtual state. The former is an accurate yet unattainable description of the system. The latter is an artificial construct generated by nominal models and does not exist in the physical world. We use $\bar{\mathbf{x}}_t$ to represent the virtual state and $\mathbf{x}_t$ to represent the real state. Likewise, the real measurement is denoted as $\mathbf{y}_t$, and its virtual counterpart, which is generated by the nominal output probability, is denoted as $\bar{\mathbf{y}}_t$. The HMM with model mismatch is depicted in Fig. 2 and outlined in (8) as follows:

$$\mathbf{x}_0 \sim p_0(\mathbf{x}_0), \quad \bar{\mathbf{x}}_t \sim p(\bar{\mathbf{x}}_t|\mathbf{x}_{t-1}), \quad \bar{\mathbf{y}}_t \sim p(\bar{\mathbf{y}}_t|\bar{\mathbf{x}}_t), \quad (8a)$$

$$d_x(\mathbf{x}_t, \bar{\mathbf{x}}_t) \leq \mathbf{r}_x, \quad d_y(\mathbf{y}_t, \bar{\mathbf{y}}_t) \leq \mathbf{r}_y. \quad (8b)$$

Here, $p(\bar{\mathbf{x}}_t|\mathbf{x}_{t-1})$ and $p(\bar{\mathbf{y}}_t|\bar{\mathbf{x}}_t)$ denote the nominal transition probability and nominal output probability respectively; $\mathbf{r}_x$ and $\mathbf{r}_y$ are the threshold random variables depicting the upper bound of model mismatch, with their distributions characterized by the cumulative distribution functions F_x and F_y , respectively. Importantly, $\mathbf{r}_x$ is assumed to be independent of both $\mathbf{x}_t$ and $\bar{\mathbf{x}}_t$, and a similar independence assumption is made for $\mathbf{r}_y$ relative to $\mathbf{y}_t$ and $\bar{\mathbf{y}}_t$. This new form in (8) is called uncertain hidden Markov model.

It is crucial to differentiate between the concepts of system stochasticity (see (8a)) and model mismatch (see (8b)). The distinction hinges on the presence of known mathematical forms. System stochasticity can be accurately modeled using explicit distributions with associated parameters. In contrast, model mismatch refers to the inherent limitations and uncertainties in a model's ability to represent the real system. Typically, this can only be quantified by an upper bound that reflects the extent to which the model deviates from reality. This distinction is pivotal in understanding and building uncertain HMM.

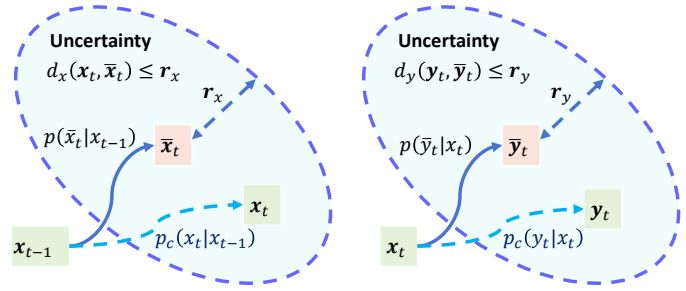


Fig. 2. Illustration of uncertain HMM with model mismatch. The nominal transition probability in HMM projects the real state in the previous time, denoted as $\mathbf{x}_{t-1}$, to a model-predicted virtual state $\bar{\mathbf{x}}_t$. However, due to model mismatch, $\mathbf{x}_{t-1}$ transitions to the real state $\mathbf{x}_t$ in the physical world. The transition from $\bar{\mathbf{x}}_t$ to $\mathbf{x}_t$ is inaccessible; instead, their distance $d_x(\mathbf{x}_t, \bar{\mathbf{x}}_t)$ are bounded by a threshold random variable $\mathbf{r}_x$, i.e., $d_x(\mathbf{x}_t, \bar{\mathbf{x}}_t) \leq \mathbf{r}_x$. Analogously, $d_y(\mathbf{y}_t, \bar{\mathbf{y}}_t) \leq \mathbf{r}_y$ reflects the measurement model mismatch.

Remark 3: When the real system is perfectly modelled, i.e., $\mathbf{x}_t = \bar{\mathbf{x}}_t$, $\mathbf{y}_t = \bar{\mathbf{y}}_t$, we have $p(\bar{\mathbf{x}}_t|\mathbf{x}_{t-1}) = p(\mathbf{x}_t|\mathbf{x}_{t-1})$ and $p(\bar{\mathbf{y}}_t|\bar{\mathbf{x}}_t) = p(\mathbf{y}_t|\mathbf{x}_t)$. In this case, the uncertain HMM (8) reduces to the standard HMM (7).

Remark 4: The two nominal probabilities $p(\bar{\mathbf{x}}_t|\mathbf{x}_{t-1})$ and $p(\bar{\mathbf{y}}_t|\bar{\mathbf{x}}_t)$ can be written to state space model (SSM):

$$\bar{\mathbf{x}}_{t+1} = \bar{f}(\mathbf{x}_t, \bar{\xi}_t), \quad \bar{\mathbf{y}}_t = \bar{g}(\mathbf{x}_t, \bar{\zeta}_t).$$

Here, $\bar{f}$ is nominal transition model, $\bar{g}$ is nominal measurement model, $\bar{\xi}_t$ is virtual process noise, and $\bar{\zeta}_t$ is virtual measurement noise. More discussion about connection between HMM and SSM can be found in Section VI-C in supplementary material.

B. Filtering Algorithm

When the system is perfectly modelled as in (7), Bayesian filtering serves as an ideal framework to calculate the posterior of system state by iteratively performing (9a) and (9b):

$$p(\mathbf{x}_t|\mathbf{y}_{1:t-1}) = \int p(\mathbf{x}_t|\mathbf{x}_{t-1})p(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1})d\mathbf{x}_{t-1}, \quad (9a)$$

$$p(\mathbf{x}_t|\mathbf{y}_{1:t}) = \frac{p(\mathbf{x}_t|\mathbf{y}_{1:t-1})p(\mathbf{y}_t|\mathbf{x}_t)}{\int p(\mathbf{x}_t|\mathbf{y}_{1:t-1})p(\mathbf{y}_t|\mathbf{x}_t)d\mathbf{x}_t}. \quad (9b)$$

Here, $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ is the posterior, $p(\mathbf{x}_t|\mathbf{y}_{1:t-1})$ is the prior. As a tradition, (9a) is called prediction and (9b) is called update. These two equations originate from total probability rule and Bayes' law respectively. In fact, almost all existing Bayesian filtering algorithms adhere to this framework, with their differences lying in how these two equations are calculated.

When considering the HMM with model mismatch (8), we need to shift the mathematical foundation to convolutional rules (6). First, let us redefine some core probability distributions in Bayesian filtering, including posterior distribution, prior distribution, transition probability, and output probability.

The redefinition relies on the uncertain HMM (8):

$$\begin{aligned} p_c(x_t|y_{1:t}) &\triangleq p(x_t|y_{1:t}, d_x(\mathbf{x}_i, \bar{\mathbf{x}}_i) \leq \mathbf{r}_x, d_y(\mathbf{y}_i, \bar{\mathbf{y}}_i) \leq \mathbf{r}_y), \\ p_c(x_t|y_{1:t-1}) &\triangleq p(x_t|y_{1:t-1}, d_x(\mathbf{x}_i, \bar{\mathbf{x}}_i) \leq \mathbf{r}_x, d_y(\mathbf{y}_j, \bar{\mathbf{y}}_j) \leq \mathbf{r}_y), \\ p_c(x_t|x_{t-1}) &\triangleq p(x_t|x_{t-1}, d_x(\mathbf{x}_t, \bar{\mathbf{x}}_t) \leq \mathbf{r}_x), \\ p_c(y_t|x_t) &\triangleq p(y_t|x_t, d_y(\mathbf{y}_t, \bar{\mathbf{y}}_t) \leq \mathbf{r}_y), \\ i &= 1, 2, \dots, t, j = 1, 2, \dots, t-1. \end{aligned} \quad (10)$$

Here, $p_c(x_t|y_{1:t})$, $p_c(x_t|y_{1:t-1})$, $p_c(x_t|x_{t-1})$ and $p_c(y_t|x_t)$ are convolutional distributions, each corresponding to their respective meanings. Then, we will illustrate how to use these definitions in (10) to derive a new Bayesian filtering framework to handle model mismatch. We begin with the assumption of conditional independence:

Assumption 1 (Conditional Independence): $\mathbf{x}_t$ and $\bar{\mathbf{x}}_t$ are conditionally independent given $\mathbf{x}_{t-1}$, i.e., $p(\bar{\mathbf{x}}_t|x_t, x_{t-1}) = p(\bar{\mathbf{x}}_t|x_{t-1})$. Besides, $\mathbf{y}_t$ and $\bar{\mathbf{y}}_t$ are conditionally independent given $\mathbf{x}_t$.

This assumption suggests that the virtual state can be inferred directly from the past state, without additional information from the current state. Also, the virtual measurement can be inferred directly from the state, without additional information from the real measurement. This assumption originates from the philosophical belief that the physical world and the modeling of a system are mutually exclusive at any given moment; that is, the act of modeling does not affect the system in the physical world, nor does the physical system influence its nominal model. Under the assumption of conditionally independence, we can obtain the main result of the paper:

Theorem 1 (Convolutional Bayesian Filtering): Under Assumption 1, the ConvBF is calculated recursively by

$$p_c(x_t|y_{1:t-1}) = \int p_c(x_t|x_{t-1})p_c(x_{t-1}|y_{1:t-1})d\mathbf{x}_{t-1}, \quad (11a)$$

$$p_c(x_t|y_{1:t}) = \frac{p_c(x_t|y_{1:t-1})p_c(y_t|x_t)}{\int p_c(x_t|y_{1:t-1})p_c(y_t|x_t)d\mathbf{x}_t}. \quad (11b)$$

Here, the convolutional transition probability is

$$p_c(x_t|x_{t-1}) = \frac{\int_{\bar{\mathbf{x}}_t} (1 - F_x(d_x(x_t, \bar{\mathbf{x}}_t))) p(\bar{\mathbf{x}}_t|x_{t-1}) d\bar{\mathbf{x}}_t}{\int_{\mathbf{x}_t} \int_{\bar{\mathbf{x}}_t} (1 - F_x(d_x(x_t, \bar{\mathbf{x}}_t))) p(\bar{\mathbf{x}}_t|x_{t-1}) d\bar{\mathbf{x}}_t d\mathbf{x}_t}, \quad (12a)$$

and the convolutional output probability is

$$p_c(y_t|x_t) = \frac{\int_{\bar{\mathbf{y}}_t} (1 - F_y(d_y(y_t, \bar{\mathbf{y}}_t))) p(\bar{\mathbf{y}}_t|x_t) d\bar{\mathbf{y}}_t}{\int_{\mathbf{y}_t} \int_{\bar{\mathbf{y}}_t} (1 - F_y(d_y(y_t, \bar{\mathbf{y}}_t))) p(\bar{\mathbf{y}}_t|x_t) d\bar{\mathbf{y}}_t d\mathbf{y}_t}. \quad (12b)$$

Proof: The transition and output probabilities in (7) are standard conditional probabilities. By leveraging the definition of convolutional conditional probability, we can derive the convolutional forms of (7) as in (12). Consequently, their convolutional counterparts are referred to as the convolutional transition probability (12a) and the convolutional output probability (12b), respectively. Then by utilizing (6), we can deduce ConvBF (11). ■

Remark 5: In the iterative process, (11a) and (11b) resemble the prediction and update steps of Bayesian filtering, respectively. The only difference is that all the probability distributions are transformed into their convolutional counterparts. Therefore, we refer to this iterative process as ConvBF.

C. Analytical Form in Gaussian Cases

A major challenge in ConvBF is the difficulty in computing the integrals in (12a) and (12b), as their analytical solutions generally do not exist. However, an exceptional case arises when distance metrics are represented as quadratic forms, threshold distributions are chosen as exponential distributions, and virtual noises are characterized as additive Gaussian. In this specific case, we have an analytical version of ConvBF.

Corollary 1: Consider the following nominal system model

$$\begin{aligned} p(\bar{\mathbf{x}}_t|x_{t-1}) &= \mathcal{N}(\bar{\mathbf{x}}_t; f(x_{t-1}), Q), \\ p(\bar{\mathbf{y}}_t|x_t) &= \mathcal{N}(\bar{\mathbf{y}}_t; g(x_t), R). \end{aligned} \quad (13)$$

If $d_x(\mathbf{x}, \bar{\mathbf{x}}) = \|\mathbf{x} - \bar{\mathbf{x}}\|^2$, $d_y(\mathbf{y}, \bar{\mathbf{y}}) = \|\mathbf{y} - \bar{\mathbf{y}}\|^2$, $\mathbf{r}_x \sim \text{Exp}(\alpha)$ and $\mathbf{r}_y \sim \text{Exp}(\beta)$ with $\alpha, \beta > 0$ being exponential coefficients, we have

$$p_c(x_t|x_{t-1}) = \mathcal{N}(x_t; f(x_{t-1}), Q + 1/(2\alpha) \cdot I_{n \times n}), \quad (14a)$$

$$p_c(y_t|x_t) = \mathcal{N}(y_t; g(x_t), R + 1/(2\beta) \cdot I_{m \times m}). \quad (14b)$$

The proof of this corollary can be found in Section VI-D of the supplementary material. This corollary shows that by using quadratic distance metrics and choosing exponential threshold variables, the covariance matrix of the convolutional transition probability for system (13) essentially becomes the nominal covariance matrix plus a constant matrix related to the exponential coefficient. As the exponential coefficient increases, the exponential distribution becomes more concentrated, with its mean and variance tending towards zero. This implies that the uncertain HMM becomes increasingly deterministic. When the exponential coefficient becomes infinity, the effect of model mismatch diminishes, and the convolutional transition probability $p_c(x_t|x_{t-1})$ reduces to the nominal transition probability $p(x_t|x_{t-1})$. This analysis is equally applicable to convolutional output probability and nominal output probability.

For linear Gaussian case, where the system in (13) satisfies $f(x_{t-1}) = Ax_{t-1}$ and $g(x_t) = Cx_t$, standard Bayesian filtering simplifies to the canonical Kalman filter. By using Corollary 1, the canonical Kalman filter can be transformed into its convolutional version by only replacing the covariance matrix of process noise Q with $Q + 1/(2\alpha)I$, and the covariance matrix of measurement noise R with $R + 1/(2\beta)I$.

Remark 6: The resulting method is an outlier-robust Kalman filter (KF). Unlike the robust regression KF [2]–[4], it preserves the original structure of KF, maintaining the conjugate nature of Gaussian distributions without increasing the computational burden. Besides, our method is in alignment with the well-established results for engineering practice of the KF, as discussed in Chapter 6.1 of [14] and Theorem 7.6 of [15]: if the modeling of noise covariance is imprecise, it is common practice to opt for a larger covariance in the KF. This treatment is proven to preserve stability, albeit resulting in a more conservative filter.

D. Approximation via Exponential Density Rescaling

In general, the ConvBF recursion does not admit closed-form solutions, except in special cases such as Gaussian noise models (see Corollary 1). This analytical intractability motivates the development of an exponential density rescaling technique to approximate the computation of ConvBF. The key idea is to replace the nominal transition and measurement probabilities with suitably rescaled versions. Intuitively,

raising a density function to a fractional power and then renormalizing it flattens the distribution while preserving its overall shape.

We consider a setting where the distance metric is quantified by cross entropy between the empirical distribution and the nominal distribution:

$$\begin{aligned} d_x(\mathbf{x}_t, \bar{\mathbf{x}}_t) &= D_{\text{CE}}(\hat{p}_{\mathbf{x}_t} \| p(\bar{\mathbf{x}}_t | x_{t-1})), \\ d_y(\mathbf{y}_t, \bar{\mathbf{y}}_t) &= D_{\text{CE}}(\hat{p}_{\mathbf{y}_t} \| p(\bar{\mathbf{y}}_t | x_t)), \end{aligned}$$

where $\hat{p}_{\mathbf{x}_t}(x) = \delta(x - \mathbf{x}_t)$ and $\hat{p}_{\mathbf{y}_t}(y) = \delta(y - \mathbf{y}_t)$ denote empirical distributions. The threshold random variables are assumed to follow exponential distributions $\mathbf{r}_x \sim \text{Exp}(\alpha)$ and $\mathbf{r}_y \sim \text{Exp}(\beta)$ with $\alpha, \beta > 0$.

To illustrate the idea, we examine the update step, noting that the prediction step can be treated analogously. Under this formulation and following (4), the convolutional output probability $p_c(y_t | x_t)$ satisfies

$$\begin{aligned} p_c(y_t | x_t) &\propto P(d_y(\mathbf{y}_t, \bar{\mathbf{y}}_t) \leq \mathbf{r}_y | y_t, x_t) \\ &= \int_{\bar{\mathbf{y}}_t} (1 - F_y(d_y(y_t, \bar{\mathbf{y}}_t))) p(\bar{\mathbf{y}}_t | x_t) d\bar{\mathbf{y}}_t \\ &= \int_{\bar{\mathbf{y}}_t} e^{-\beta D_{\text{CE}}(\hat{p}_{\mathbf{y}_t} \| p(\bar{\mathbf{y}}_t | x_t))} p(\bar{\mathbf{y}}_t | x_t) d\bar{\mathbf{y}}_t \\ &= \mathbb{E}_{\bar{\mathbf{y}}_t \sim p(\bar{\mathbf{y}}_t | x_t)} \left\{ e^{-\beta D_{\text{CE}}(\hat{p}_{\mathbf{y}_t} \| p(\bar{\mathbf{y}}_t | x_t))} \right\}. \end{aligned}$$

Substituting the empirical distribution simplifies the cross-entropy as

$$\begin{aligned} D_{\text{CE}}(\hat{p}_{y_t} \| p(\cdot | x_t)) &= - \int \delta(y - y_t) \log p(y | x_t) dy \\ &= - \log p(y_t | x_t), \end{aligned}$$

Here, $p(y_t | x_t)$ is not the true output probability; rather, it is the value of the nominal output probability $p(\bar{\mathbf{y}}_t | x_t)$ evaluated at the real measurement y_t . Consequently, we have $p_c(y_t | x_t) \propto p(y_t | x_t)^\beta$. Thus, the exponential weighting amounts to raising the nominal density $p(y_t | x_t)$ to the power β (up to normalization).

A similar derivation applies to the state transition probability, yielding the approximations:

$$\begin{aligned} p_c(x_t | x_{t-1}) &\approx \frac{p(x_t | x_{t-1})^\alpha}{\int p(x_t | x_{t-1})^\alpha dx_t} \triangleq p_e(x_t | x_{t-1}), \\ p_c(y_t | x_t) &\approx \frac{p(y_t | x_t)^\beta}{\int p(y_t | x_t)^\beta dy_t} \triangleq p_e(y_t | x_t). \end{aligned}$$

Thus, the convolutional transition and output densities can be approximated by the exponentially rescaled versions of the nominal transition and measurement densities.

The effect of this rescaling becomes particularly clear in linear Gaussian models. Suppose the nominal output distribution is $p(\bar{\mathbf{y}}_t | x_t) = \mathcal{N}(\bar{\mathbf{y}}_t; Cx_t, R)$. After rescaling, one obtains $p_e(y_t | x_t) = \mathcal{N}(y_t; Cx_t, 1/\beta \cdot R)$, indicating that the covariance matrix is scaled from R to R/β . The parameter β thus controls the sharpness of the distribution: larger β yields a more concentrated density, while smaller β broadens it. This behavior aligns with the analytical Gaussian result in Corollary 1.

A practical question is how to choose the parameters α and β automatically. To this end, we design two adaptive schemes that adjust the exponents based on the mismatch in the transition and measurement models. The key idea is to use

the (pseudo-)innovation as an indicator of model inconsistency. For measurement-model mismatch, we select

$$\beta = (1 + \|y_t - g(\hat{x}_{t|t-1})\|_{R^{-1}})^{-1} \leq 1. \quad (15)$$

Under this rule, a large normalized innovation reduces β , thereby downweighting unreliable measurements.

Similarly, the transition exponent α is adapted using the pseudo-innovation $v_{x,t} \triangleq \hat{x}_{t|t} - f(\hat{x}_{t-1|t-1})$, which measures the deviation from the nominal transition model. An analogous update rule is adopted:

$$\alpha = (1 + \|v_{x,t}\|_{Q^{-1}})^{-1} \leq 1. \quad (16)$$

When the dynamical residual $v_{x,t}$ is large, α decreases and transition probability flattens, corresponding to inflating the process covariance from Q to Q/α . Overall, these adaptive rules provide a unified mechanism for adjusting (α, β) based on real-time evidence of model mismatch using the exponential density rescaling principle. When the innovations are small, the exponents remain close to one and the method reduces to the standard Bayesian update. When significant mismatches occur, the exponents shrink toward zero, yielding more robust behavior. The pseudocode for the convolutional Kalman family using this adaptive scheme is provided in Algorithm 1 - 3 in Section VI-F of supplementary material.

Remark 7: Adaptive filtering techniques such as Bayesian updates of unknown noise covariances, maximum-likelihood estimation of parameters, and covariance-matching methods, primarily aim to adapt or learn noise statistics directly from the data [16]. They are designed under the assumption that the underlying Bayesian model is still correct. In contrast, ConvBF aims to provide a theoretically grounded mechanism for handling model mismatch. More importantly, we are able to prove that the resulting convolutional Kalman filter possesses a bounded influence function, whereas the standard Kalman filter does not (See Section VI-G in supplementary material), providing a formal robustness guarantee not shared by adaptive filters.

V. SIMULATIONS

In this section, we evaluate our proposed framework across two benchmark systems to demonstrate its applicability to classic filtering algorithms under various types of model mismatch. For each setting, we conduct $N = 200$ Monte Carlo experiments with $M = 200$ time steps. The performance metric throughout the simulations is the root mean square error (RMSE), defined as $\sqrt{\frac{1}{Mn} \sum_{i=1}^M \|x_i - \hat{x}_i\|^2}$, where x_i and $\hat{x}_i$ denote the true and estimated state at the i -th time step.

A. Linear Wiener Velocity Model

The Wiener velocity model is a well-known standard linear model in the field of target tracking, where the velocity is modeled as the Wiener process [17]. The state $\mathbf{x} = [p_x, p_y, v_x, v_y]^\top$ consists of system positions p_x, p_y and velocities v_x, v_y . The nominal model is given by

$$\begin{aligned} \bar{\mathbf{x}}_{t+1} &= \begin{bmatrix} 1 & 0 & 0.1 & 0 \\ 0 & 1 & 0 & 0.1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}_t + \bar{\xi}_t, \\ \bar{\mathbf{y}}_t &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \mathbf{x}_t + \bar{\zeta}_t. \end{aligned}$$

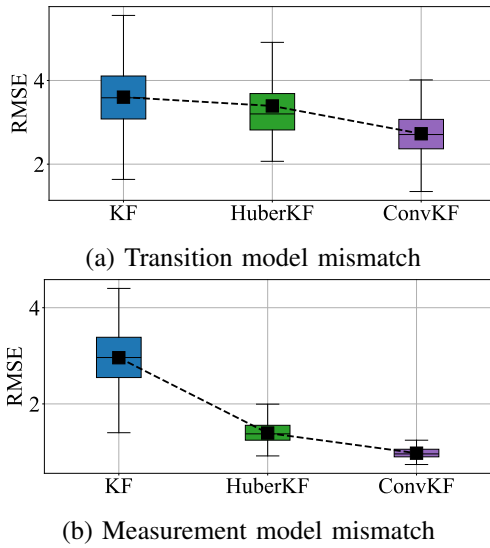


Fig. 3. Box plot of RMSE for KF, HuberKF and ConvKF for wiener velocity model. The black square “■” represents the average RMSE.

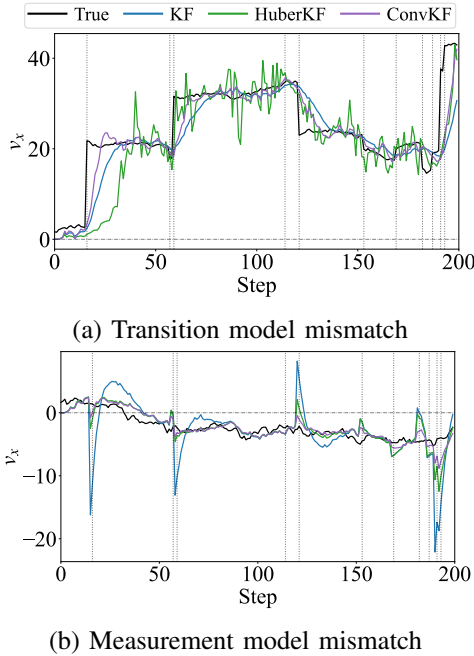


Fig. 4. Curve of the state estimates of KF, HuberKF, ConvKF and the true state for v_x of the wiener velocity model.

Here, the virtual process noise $\bar{\xi}_t$ is modelled by $\bar{\xi}_t \sim \mathcal{N}(0, Q)$ with covariance

$$Q = \begin{bmatrix} 0.1^3/3 & 0 & 0.1^2/2 & 0 \\ 0 & 0.1^3/3 & 0 & 0.1^2/2 \\ 0.1^2/2 & 0 & 0.1 & 0 \\ 0 & 0.1^2/2 & 0 & 0.1 \end{bmatrix},$$

and virtual measurement noise satisfies $\bar{\zeta}_t \sim \mathcal{N}(0, R)$ with $R = \mathbb{I}_{2 \times 2}$. Additionally, the initial state $\mathbf{x}_0$ satisfies $\mathbf{x}_0 \sim \mathcal{N}\left(\begin{bmatrix} 0 & 0 & 1 & 1 \end{bmatrix}^\top, \mathbb{I}_{4 \times 4}\right)$. To evaluate the effectiveness of our designed filter, we consider two cases for model mismatch:

(a) **Transition Model Mismatch:** In the real system, the

process noise is a mixture of Gaussian noises $\xi_t \sim 0.95 \cdot \mathcal{N}(0, Q) + 0.05 \cdot \mathcal{N}(0, 1000Q)$, while the distribution of the measurement noise is just as the virtual noise modeled.

(b) **Measurement Model Mismatch:** The process noise is Gaussian as the virtual one, while the measurement noise in the real system is a mixture of Gaussian noises $\zeta_t \sim 0.95 \cdot \mathcal{N}(0, R) + 0.05 \cdot \mathcal{N}(0, 1000R)$.

We apply the ConvBF framework to KF using exponential density rescaling and the adaptive schemes in (15) and (16). The resulting filter, referred to as ConvKF (see Algorithm 1 in the supplementary material), is compared against the standard KF and the robust HuberKF [2]. Fig. 3 summarizes the RMSE performance. In the transition-model mismatch setting, ConvKF achieves the lowest average RMSE (2.73), compared with 3.60 for KF and 3.39 for HuberKF, corresponding to improvements of roughly 24% and 20%, respectively. In the measurement-model mismatch setting, ConvKF again performs best with an average RMSE of 0.98, outperforming KF (2.96) and HuberKF (1.39) by about 67% and 30%.

In addition to the aggregated RMSE, Fig. 4 visualizes the estimated velocity v_x over time. The dashed vertical lines indicate the moments when outliers occur. KF exhibits large deviations at each outlier and recovers slowly because it does not account for model mismatch. HuberKF attenuates the impact of moderate outliers but still shows overshoot and drift when disturbances are large, partly due to its reliance on a recursive nonlinear optimization step that may introduce instability. In contrast, ConvKF remains close to the true trajectory around the outlier locations and returns to accurate tracking within only a few steps. This stability is a direct consequence of the exponential density rescaling mechanism: when the (pseudo-)innovation becomes abnormally large, the effective likelihood or transition probability automatically flattens, reducing the influence of extreme outliers while preserving the Gaussian conjugate structure.

B. Nonlinear Sequence Forecasting System

Further, we conduct experiments on a nonlinear sequence forecasting system [18]. The nominal model is

$$\begin{aligned} \bar{\mathbf{x}}_{t+1} &= \mathbf{x}_t + \kappa_1 \cdot \begin{bmatrix} -1 & 0.1 \\ 0.1 & -1 \end{bmatrix} \mathbf{x}_t + \kappa_2 \cdot \cos(\mathbf{x}_t) + \bar{\xi}_t, \\ \bar{\mathbf{y}}_t &= \mathbf{x}_t + \sin(\mathbf{x}_t) + \bar{\zeta}_t, \end{aligned}$$

where $\mathbf{x}_0 \sim \mathcal{N}(0, \mathbb{I}_{2 \times 2})$. Both the constants κ_1 and κ_2 are set to 0.1. We assume $Q = R = \mathbb{I}_{2 \times 2}$ for virtual process noise $\bar{\xi}_t \sim \mathcal{N}(0, Q)$ and virtual measurement noise $\bar{\zeta}_t \sim \mathcal{N}(0, R)$, respectively. We apply ConvBF for EKF and UKF following the exponential density rescaling technique and the resulting approaches are named as convolutional EKF (ConvEKF) and convolutional UKF (ConvUKF), respectively. More details of the algorithms can be found in Algorithm 2 and Algorithm 3 in supplementary material. We compare our proposed methods with the standard UKF [19], standard EKF [20] and Huber UKF [21]. The Huber UKF is a robust version of the UKF, which replaces the quadratic loss in the optimization of the update step with Huber loss. Our comparisons consider the following two cases:

(a) **Transition Model Mismatch:** The real process noise follows a Gaussian mixture $\xi_t \sim 0.9 \cdot \mathcal{N}(0, Q) + 0.1 \cdot \mathcal{N}(0, 1000Q)$, while the measurement noise matches the nominal assumptions.

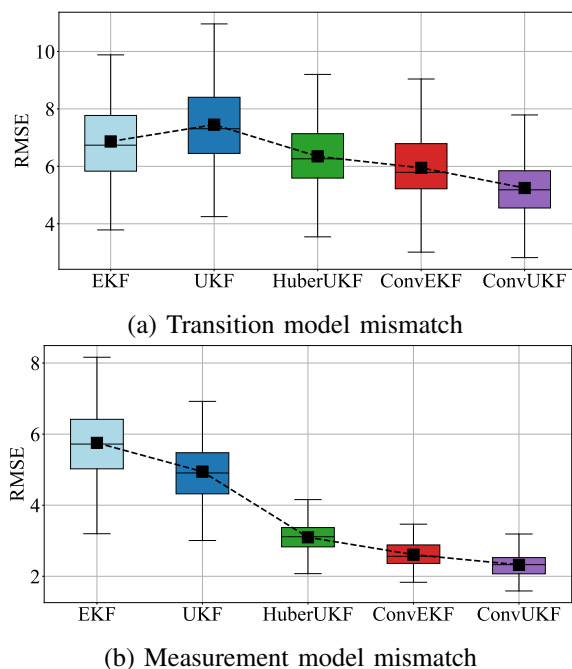


Fig. 5. Box plot of RMSE for EKF, UKF, HuberUKF, ConvEKF, ConvUKF for sequence forecasting model.

(b) Measurement Model Mismatch: The process noise matches nominal assumption, but the measurement noise becomes a mixture $\zeta_t \sim 0.9 \cdot \mathcal{N}(0, R) + 0.1 \cdot \mathcal{N}(0, 1000R)$.

As shown in Fig. 5, in the transition model mismatch case, ConvEKF reduces the RMSE by approximately 13.4% compared with standard EKF and by about 6.3% compared with HuberUKF. ConvUKF also improves significantly over the standard UKF, achieving a reduction of about 29.6%, and further lowers the RMSE by 17.3% compared with HuberUKF. In the measurement model mismatch case, the improvement becomes even more substantial. ConvEKF reduces the RMSE by 54.7% compared with standard EKF and by 15.8% compared with HuberUKF. ConvUKF achieves the best overall performance, lowering the RMSE by 53.0% relative to standard UKF and by 25.0% relative to HuberUKF.

VI. CONCLUSION

This paper introduces a ConvBF framework by transforming transition and output probabilities into convolutional forms, broadening the scope of Bayesian filtering. We demonstrate that ConvBF possesses analytical forms of convolution operation in systems with Gaussian noises. For non-Gaussian cases, the transition and output probabilities can be effectively approximated by scaling them into fractional powers, when employing the cross entropy as the distance measure. This leads to an enhanced version of the Kalman filter, which achieves robustness through simple modifications to the noise covariance matrix, while still preserving the conjugate nature of Gaussian distributions. The practical efficacy of ConvBF is demonstrated through its application to Kalman filter family.

REFERENCES

- [1] Y. Ho and R. Lee, "A Bayesian approach to problems in stochastic estimation and control," *IEEE Transactions on Automatic Control*, vol. 9, no. 4, pp. 333–339, 1964.
- [2] M. A. Gandhi and L. Mili, "Robust Kalman filter based on a generalized maximum-likelihood-type estimator," *IEEE Transactions on Signal Processing*, vol. 58, no. 5, pp. 2509–2520, 2009.
- [3] M. Roth, E. Özkan, and F. Gustafsson, "A Student's t filter for heavy tailed process and measurement noise," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5770–5774, IEEE, 2013.
- [4] B. Chen, X. Liu, H. Zhao, and J. C. Principe, "Maximum correntropy Kalman filter," *Automatica*, vol. 76, pp. 70–77, 2017.
- [5] G. Agamennoni, J. I. Nieto, and E. M. Nebot, "Approximate inference in state-space models with heavy-tailed noise," *IEEE Transactions on Signal Processing*, vol. 60, no. 10, pp. 5024–5037, 2012.
- [6] R. Piché, S. Särkkä, and J. Hartikainen, "Recursive outlier-robust filtering and smoothing for nonlinear systems using the multivariate Student-t distribution," in *2012 IEEE International Workshop on Machine Learning for Signal Processing*, pp. 1–6, IEEE, 2012.
- [7] J. Knoblauch, J. Jewson, and T. Damoulas, "An optimization-centric view on Bayes' rule: Reviewing and generalizing variational inference," *Journal of Machine Learning Research*, vol. 23, no. 132, pp. 1–109, 2022.
- [8] F. Futami, I. Sato, and M. Sugiyama, "Variational inference based on robust divergences," in *International Conference on Artificial Intelligence and Statistics*, pp. 813–822, PMLR, 2018.
- [9] B.-E. Chérif-Abdellatif and P. Alquier, "MMD-Bayes: Robust bayesian estimation via maximum mean discrepancy," in *Symposium on Advances in Approximate Bayesian Inference*, pp. 1–21, PMLR, 2020.
- [10] T. Matsubara, J. Knoblauch, F.-X. Briol, and C. J. Oates, "Robust generalised Bayesian inference for intractable likelihoods," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 84, no. 3, pp. 997–1022, 2022.
- [11] M. A. Medina, J. L. M. Olea, C. Rush, and A. Velez, "On the robustness to misspecification of α -posteriors and their variational approximations," *Journal of Machine Learning Research*, vol. 23, no. 147, pp. 1–51, 2022.
- [12] A. Graves, "Practical variational inference for neural networks," *Advances in neural information processing systems*, vol. 24, 2011.
- [13] J. Berger, "The case for objective Bayesian analysis," *Bayesian Analysis*, vol. 1, no. 3, pp. 385 – 402, 2006.
- [14] B. D. Anderson and J. B. Moore, *Optimal filtering*. Courier Corporation, 2012.
- [15] A. H. Jazwinski, *Stochastic processes and filtering theory*. Courier Corporation, 2007.
- [16] R. Mehra, "Approaches to adaptive filtering," *IEEE Transactions on automatic control*, vol. 17, no. 5, pp. 693–698, 2003.
- [17] Z. Zhang, X. Si, C. Hu, and Y. Lei, "Degradation data analysis and remaining useful life estimation: A review on Wiener-process-based methods," *European Journal of Operational Research*, vol. 271, no. 3, pp. 775–796, 2018.
- [18] Y. Tao and S. S.-T. Yau, "Outlier-robust iterative extended Kalman filtering," *IEEE Signal Processing Letters*, 2023.
- [19] S. J. Julier and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," in *Signal processing, Sensor Fusion, and Target Recognition VI*, vol. 3068, pp. 182–193, Spie, 1997.
- [20] G. L. Smith, S. F. Schmidt, and L. A. McGee, *Application of statistical filter theory to the optimal estimation of position and velocity on board a circumlunar vehicle*, vol. 135. National Aeronautics and Space Administration, 1962.
- [21] Z. Bing, C. Lubin, J. Xu, *et al.*, "Huber-based adaptive unscented Kalman filter with non-gaussian measurement noise," *Circuits Systems and Signal Processing*, vol. 37, no. 12, pp. 1–21, 2018.
- [22] R. Durrett, *Probability: theory and examples*, vol. 49. Cambridge University Press, 2019.
- [23] S. E. Li, *Reinforcement learning for sequential decision and optimal control*. Springer, 2023.
- [24] W. Cao, C. Liu, Z. Lan, Y. Piao, and S. E. Li, "Generalized moving horizon estimation for nonlinear systems with robustness to measurement outliers," in *2023 American Control Conference (ACC)*, pp. 1614–1621, IEEE, 2023.
- [25] N. Tishby, F. C. Pereira, and W. Bialek, "The information bottleneck method," *arXiv preprint physics/0004057*, 2000.
- [26] W. Bialek, I. Nemenman, and N. Tishby, "Predictability, complexity, and learning," *Neural Computation*, vol. 13, no. 11, pp. 2409–2463, 2001.
- [27] A. A. Alemi, "Variational predictive information bottleneck," in *Symposium on Advances in Approximate Bayesian Inference*, pp. 1–6, PMLR, 2020.
- [28] P. J. Huber, "Robust statistics," in *International Encyclopedia of Statistical Science*, pp. 1248–1251, Springer, 2011.