

This article investigates the event-triggered feedback particle filter (FPF) problem in general continuous-discrete (CD) Gaussian systems. An event-triggered measurement-assimilation mechanism based on Kullback–Leibler divergence is integrated into the CD-FPF framework, with the trigger criterion approximated in two steps so that the decision can be evaluated at the sensor end. This mechanism aligns naturally with the FPF’s theoretical framework, preserving its original structure and advantages. Updates are omitted when observations are discarded, reducing costs, but proceed normally when retained. Numerical experiments employing the Euler–Maruyama discretization method validate the effectiveness of the proposed approach, which reduces the sensor-side transmission overhead at all triggering rates and reduces the global computational cost when the triggering rate falls below a quantifiable break-even point determined by the regression overhead at the estimator.

I. INTRODUCTION

STATE estimation has long been a fundamental research area in signal processing and control engineering, with numerous filtering techniques developed for this purpose. The theoretical foundation of state estimation dates back to Gauss’s least squares method [1]. The introduction of the Kalman filter (KF) by Kalman established the cornerstone of modern state estimation [2], proving to be optimal under the minimum mean square error criterion. However, practical systems often violate linearity assumptions, making the study of nonlinear system state estimation particularly important. To address nonlinear systems, the extended KF (EKF) was developed [3]. This technique employs a first-order Taylor series expansion linearized at

Received 28 November 2025; revised 6 June 2026; accepted 7 August 2026. Date of publication 12 August 2026; date of current version 14 September 2026.

DOI. No. 10.1109/TAES.2026.3723394

Refereeing of this contribution was handled by C.-F. Huang.

This work was supported by the National Natural Science Foundation of China under Grant 42450242.

Authors’ addresses: Chang Qin, Tong Guo, and Yao Mao are with State Key Laboratory of Optical Field Manipulation Science and Technology, Institute of Optics and Electronics, Chinese Academy of Sciences, Chengdu 610209, China, also with the Key Laboratory of Optical Engineering, Chinese Academy of Sciences, Chengdu 610209, China, and also with University of Chinese Academy of Sciences, Beijing 101408, China, E-mail: (qinchang24@mailsucas.ac.cn; guotong20@mailsucas.ac.cn; maoyao@ioe.ac.cn); Jiayi Kang is with Hetao Institute of Mathematics and Interdisciplinary Sciences (HIMIS), Shenzhen 518000, China, and also with Beijing Institute of Mathematical Sciences and Applications (BIMSA), Beijing 101408, China, E-mail: (kangjiayi@himis-sz.cn); Stephen S.-T. Yau is with Department of Mathematical Sciences, Tsinghua University, Beijing 100084, China, (E-mail: (yau@uic.edu)). (*Corresponding authors: Stephen S.-T. Yau; Jiayi Kang.*)

0018-9251 © 2026 IEEE

the present state estimate to approximate both the nonlinear dynamic and observation models. However, neglecting higher order terms introduces cumulative linearization errors that degrade performance. The unscented KF (UKF) selects a set of deterministic sample points based on the unscented transform to approximate the propagation of the system state's probability density [4]. This avoids explicit Jacobian matrix calculation, offering better numerical stability in certain scenarios. In contrast, the cubature KF employs the spherical-radial cubature rule to generate sample points, achieving a Gaussian-weighted integral approximation of the state's posterior probability density [5]. For continuous-time systems, there exist corresponding filtering approaches, such as the Kalman–Bucy filter [6] and other variants. However, the aforementioned methods based on Kalman filtering typically require linearization and Gaussian assumptions, which may reduce the prediction accuracy for complex systems. In contrast, particle filters (PFs) do not rely on linearization or Gaussian assumptions [14]. By approximating state posterior distributions using empirical particle distributions, they perform better when dealing with low- to mid-dimensional nonlinear systems. However, PFs suffer from particle degeneracy and the curse of dimensionality in high-dimensional systems [26]. Improving performance needs resampling and a large number of particles. Recently, the feedback PF (FPF) [15], proposed for continuous-time systems, eliminates resampling requirements while overcoming sampling issues and dimensionality challenges. As a Monte Carlo method, the FPF, inspired by mean-field optimal control, treats particles as controlled stochastic systems and derives optimal feedback control laws by minimizing the Kullback–Leibler (KL) divergence between the true posterior and the particle distribution. The aforementioned methods apply exclusively to either discrete-time or continuous-time systems. However, continuous-time systems cannot adequately capture intermittent observations, while discrete-time systems inevitably introduce approximation errors when modeling continuous state evolution. Over recent decades, continuous-discrete (CD) systems have gained prominence across various applications, including ballistic missile tracking [7], finance [30], navigation [9], and aerospace engineering [8]. In navigation applications, for instance, target states (such as position and velocity) evolve continuously, whereas sensors provide only discrete intermittent observations [9]. Substantial research exists on state estimation for CD systems. The CD-KF provides the optimal solution for linear systems under the mean squared error criterion [10]. For nonlinear systems, the CD extended KF (CD-EKF) was first proposed in [11]. Building upon CD-KF, the authors in [12] developed the CD-UKF, achieving CD-EKF-level performance with reduced computational complexity. Subsequently, the CD-PF emerged, demonstrating superior performance compared with the CD-EKF [13]. The FPF framework has also been extended to CD systems in [16]. In remote state estimation, resource-constrained sensors communicate with estimators via wireless channels, where frequent transmissions may cause packet losses and delays [17]. The event-triggered

mechanism (ETM) addresses this by permitting data transmission only when specific conditions are satisfied, thereby reducing unnecessary communication while maintaining estimation performance. Event-triggered state estimation has been extensively studied for continuous-time and discrete-time systems [18], [19], [20], [21], [22], [24], [28], [29]. However, research on CD systems remains limited due to the challenge of simultaneously handling continuous dynamics and discrete measurements. Specifically, continuous process noise and discrete measurement noise impact trigger decisions differently, potentially causing false triggers or missed detections. We briefly position the proposed method against three lines of related work. Compared with event-triggered PFs using KL-divergence or variational-Bayes-type approximations [21], which operate in discrete-time settings, this work is the first to instantiate a KL-based ETM within the CD-FPF, where the trigger criterion shares the variational form used in FPF's mean-field optimal-control derivation [15], [16]. Compared with innovation-based and posterior-based triggering schemes [17], [24], which probe, respectively, the residual mean and selected posterior moments, the proposed criterion uses the full prior-to-posterior divergence; in the Gaussian limit this divergence reduces to a Mahalanobis-type quadratic form, so the latter is a special case rather than a distinct paradigm. Compared with the original CD-FPF [16], which assumes that every measurement is assimilated, the present work explicitly accommodates intermittent observations and recovers the original CD-FPF as the always-triggered limit $\gamma \rightarrow 0$. Motivated by the above analysis, this article integrates a KL divergence-based event-triggered measurement-assimilation mechanism into the CD-FPF to reduce unnecessary transmissions while preserving estimation performance. The main contributions of this article are summarized as follows.

- 1) Integrated a KL divergence-based ETM into the CD-FPF framework, evaluating measurement information via particle density divergence without modifying the FPF structure; the criterion shares the variational form used in FPF's own derivation.
- 2) Developed a two-step particle-then-regression approximation that serves as a practical mechanism enabling sensor-side triggering decisions under communication constraints, reducing the sensor-side transmission overhead at all triggering rates and reducing global computational cost when the triggering rate falls below the break-even point characterized in Remark 3.
- 3) Simulations show the KL-based ETM maintains accuracy at low communication costs, with the event-triggered FPF consistently outperforming the event-triggered PF under the same KL-based criterion.

The rest of this article is organized as follows. Section II briefly introduces the CD system and the FPF. Section III provides a detailed description of the event-triggered CD

FPF with intermittent observation and the two-step approximation method for the KL divergence. In Section IV, numerical simulations are conducted to validate the effectiveness of the proposed algorithm. Finally, Section V concludes this article.

II. BACKGROUND

A. Continuous-Discrete System Model

We model the target state in continuous time via a stochastic differential equation (SDE)

$$dX_t = A(X_t)dt + \sigma_B dB_t \quad (1)$$

where $X_t \in \mathbb{R}^{n \times 1}$ represents the system state vector at time t , and $\{B_t\}$ is an n -dimensional standard Brownian motion, assumed to be independent of the initial state X_0 . The function $A : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is known and satisfies certain continuity conditions. The constant $\sigma_B \in \mathbb{R}^+$ is known and time-invariant. The observations obey the following discrete-time equation:

$$Y_{t_k} = H(X_{t_k}) + \sigma_W W_k \quad (2)$$

where $Y_{t_k} \in \mathbb{R}^{m \times 1}$ is the observation at time $t = t_k$, $X_{t_k} \in \mathbb{R}^{n \times 1}$ is the state vector at time $t = t_k$, and $\{W_k\}$ is a sequence of independent and identically distributed standard normal random variables, also independent of the state process $\{X_t\}$. The function $H : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is known and satisfies certain continuity conditions. The constant $\sigma_W \in \mathbb{R}^+$ is known and time-invariant. The corresponding probability space is denoted as $(\Omega, \mathcal{F}, \mathbb{P})$, and the σ -algebra is defined as $\mathcal{F}_t := \sigma(Y_{t_k} : t_k \leq t)$ which represents the observation history up to time t .

B. Feedback Particle Filter

According to recent research on FPF for CD systems, the objective of filtering is to compute the conditional probability density of X_t based on the accumulated measurement data $\mathcal{Z}_t := \sigma\{Y_{t_i} | \forall i \text{ s.t. } t_i \in [0, t]\}$, i.e., the posterior probability density function $p^*(x, t)$. The fundamental principle of FPF aligns with that of various PF methods: at every time instant t , the true posterior distribution is approximated by an empirical distribution formed by a particle swarm $\{X_t^i : 1 \leq i \leq N\}$, where $X_t^i \in \mathbb{R}^{n \times 1}$ represents the state of the i th particle at time t [16].

During the Continuous Time Interval: The continuous evolution of the posterior distribution $p^*(x, t)$ is governed by

$$\frac{\partial p^*}{\partial t}(x, t) = \mathcal{L}^* p^*(x, t) \quad (3)$$

where $\mathcal{L}^* p^* = -\nabla \cdot (p^* A) + \frac{1}{2} \sigma_B^2 \Delta p^*$ and the initial condition $p^*(x, t_{k-1})$ is known. Clearly, (3) can be derived from the Kolmogorov forward equation. Define $p^*(x, t_k^-) := \lim_{t \nearrow t_k} p^*(x, t)$ as the conditional probability density of X_{t_k} given $\mathcal{Z}_{t_{k-1}} := \sigma\{Y_{t_i} | \forall i \text{ s.t. } t_i \in [0, t_{k-1}]\}$, i.e., the prior probability density. Correspondingly, the i th particle evolves according to the following SDE:

$$dX_t^i = A(X_t^i)dt + \sigma_B dB_t^i \quad (4)$$

where $\{B_t^i\}$ are mutually independent n -dimensional standard Brownian motions and independent of the known initial state $X_{t_{k-1}}^i$, which represents the state vector of the i th particle at time $t = t_{k-1}$. In addition, the samples X_0^i are drawn i.i.d. from $p^*(x, 0)$. Denote

$$X_{t_k^-}^i := \lim_{t \nearrow t_k} X_t^i.$$

When a New Observation is Received at Discrete Time: The posterior distribution $p^*(x, t_k)$ is obtained by applying the Bayes' rule. To transform discrete observations into a continuous framework, a pseudo-time parameter is introduced, and two homotopy mappings are constructed

$$q_k(x, \tau) := \ln q^*(x, t_k^-) + \frac{\tau}{\sigma_W^2} H^T(x) \left(Y_{t_k} - \frac{1}{2} H(x) \right)$$

$$p_k^*(x, \tau) := \frac{\exp q_k(x, \tau)}{\int \exp q_k(x', \tau) dx'}$$

where $\tau \in [0, 1]$ serves as a pseudo-time variable, $p_k^*(x, 0) = p^*(x, t_k^-)$ and $p_k^*(x, 1) = p^*(x, t_k)$. Then, the posterior $p_k^*(x, \tau)$ evolves continuously according to

$$\frac{\partial p_k^*}{\partial \tau}(x, \tau) = \frac{p_k^*(x, \tau)}{\sigma_W^2} \left[[H(x) - \hat{H}(\tau)]^T Y_{t_k} - \frac{1}{2} H^T(x) H(x) + \frac{1}{2} H^T(x) \widehat{H}(\tau) \right] \quad (5)$$

where

$$\hat{H}(\tau) := \int p_k^*(x, \tau) H(x) dx$$

$$H^T(x) \widehat{H}(\tau) := \int p_k^*(x, \tau) H^T(x) H(x) dx.$$

Correspondingly, the i th particle evolves according to the following SDE:

$$\frac{dS_k^i}{d\tau}(\tau) = U_k^i(\tau) = \mathcal{K}_k(S_k^i(\tau), \tau) Y_{t_k} + u_k(S_k^i(\tau), \tau) \quad (6)$$

where $S_k^i(0) = X_{t_k^-}^i$. Let $p_k(x, \tau)$ denote the distribution of $S_k^i(\tau)$, the objective of FPF is to design $\mathcal{K}_k(\cdot)$ and $u_k(\cdot)$ such that $p_k(x, 1) = p_k^*(x, 1) = p^*(x, t_k)$ and $X_{t_k}^i = S_k^i(1)$. According to the Kolmogorov's forward equation, it can be derived that

$$\frac{\partial p_k}{\partial \tau}(x, \tau) = -\nabla \cdot (p_k \mathcal{K}_k) Y_{t_k} - \nabla \cdot (p_k u_k).$$

Therefore, the optimal $\mathcal{K}_k(\cdot)$ and $u_k(\cdot)$ are as follows:

$$\mathcal{K}_k(x, \tau) = [\nabla \lambda_1^T, \nabla \lambda_2^T, \dots, \nabla \lambda_m^T] \quad (7)$$

$$u_k(x, \tau) = -\frac{1}{2} \mathcal{K}_k(x, \tau) (H(x) + \hat{H}) + \frac{1}{2} \Omega_k(x, \tau) \quad (8)$$

where $\nabla \lambda_i$ is the solution of equation $\nabla \cdot (p_k \nabla \lambda_j) = -\frac{1}{\sigma_W^2} (H_j - \hat{H}_j) p_k$ with condition $\int_{\mathbb{R}^n} \lambda_j(x, \tau) p_k(x, \tau) dx = 0$, and $\Omega_k = \nabla \phi$ is the solution of equation $\nabla \cdot (p_k \nabla \phi) = -\frac{1}{\sigma_W^2} (G - \hat{G}) p_k$ with condition $\int_{\mathbb{R}^n} \phi(x, \tau) p_k(x, \tau) dx = 0$ and $G := \sum_{j=1}^m \nabla \lambda_j \cdot \nabla H_j^T$, $\hat{G} := \int p_k(x, \tau) G(x) dx$.

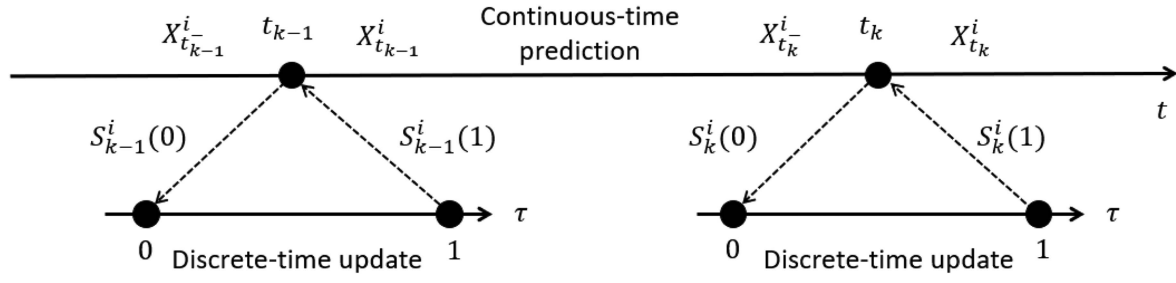


Fig. 1. Evolution process of FPF in CD system.

III. EVENT-TRIGGERED CONTINUOUS-DISCRETE FEEDBACK PARTICLE FILTER

A. Motivation

The authors in [16] proposed an FPF for CD systems that approximates the posterior distribution through particle evolution equations (4) and (6), with the two-step process illustrated in Fig. 1. However, practical limitations arise under resource constraints. First, frequent sensor transmission depletes battery power, as communication consumes more energy than local computation [31]. Second, when observation intervals in system (1) and (2) are small, single updates yield minimal improvement while the pseudo-time update step incurs substantial computation, resulting in inefficient resource utilization. This section addresses these issues.

B. Intermittent Observation Trigger Mechanism

Intermittent observations are common occurrences, such as packet loss, etc., but these are passively generated. Inspired by intermittent observation, however, we will focus on a type of artificial, proactive intermittent observation behavior. By introducing a trigger mechanism, the observation sequence can be reconstructed through sparsification at estimator end. This approach eliminates unnecessary measurement transmissions and update, improving communication efficiency and conserving computational resources while maintaining acceptable estimation accuracy. ETMs are commonly designed in the following form [21], [24]:

$$\zeta_k = \begin{cases} 0, & f_k(Y_{t_k}) < \gamma \\ 1, & f_k(Y_{t_k}) \geq \gamma. \end{cases} \quad (9)$$

Here, ζ_k is a binary decision variable where $\zeta_k = 1$ if the latest measurement Y_{t_k} is retained and sent, and $\zeta_k = 0$ otherwise. $f_k(\cdot) : \mathbb{R}^m \mapsto \mathbb{R}^+$ is the test function and $\gamma \in \mathbb{R}^+$ is the threshold. By designing the test function and the threshold, we can achieve a tradeoff between performance, communication cost, and computational cost. Therefore, the primary objective is to integrate an event-triggered measurement-assimilation mechanism into the CD-FPF framework under intermittent observations, and to analyze the resulting communication–estimation tradeoff and computational implications.

C. Construction of the Test Function

Trigger mechanisms should align with the underlying filtering paradigm. Filtering algorithms operate through two approaches: 1) Bayesian density modeling or 2) innovation-based stochastic methods. For density-based methods, divergence metrics naturally measure distributional “distance” between prior and posterior; for innovation-based methods, the innovation itself serves as the test function. Since FPF is density-based and derived using KL divergence, employing KL divergence as the test function is inherently consistent. Specifically, the FPF control law was obtained in [15] by minimizing the KL divergence between the particle distribution and the true posterior, and Lemma 1 guarantees that the resulting pseudo-time flow makes these two distributions coincide; using the same divergence between prior and posterior as the trigger criterion is therefore structurally consistent with FPF’s variational origin, rather than constituting an extrinsic add-on. Moreover, while KL-based triggering shares mean-shift sensitivity with innovation-based approaches, it critically incorporates covariance changes. This joint sensitivity enables robust detection of phenomena including abrupt process noise increases, modeling errors, and drifting measurement noise variance, effectively reducing false triggers and missed detections. The test function is thus defined as

$$f_k(Y_{t_k}) = D_{KL}(p(X_{t_k} | \mathcal{Z}_{t_{k-1}}, Y_{t_k}) \parallel p(X_{t_k} | \mathcal{Z}_{t_{k-1}})) \quad (10)$$

where $\mathcal{Z}_t := \sigma\{\zeta_i Y_{t_i} | \forall i \text{ s.t. } t_i \in [0, t]\}$ and

$$\begin{aligned} D_{KL}(p(X_{t_k} | \mathcal{Z}_{t_{k-1}}, Y_{t_k}) \parallel p(X_{t_k} | \mathcal{Z}_{t_{k-1}})) \\ = \int p(X_{t_k} | \mathcal{Z}_{t_{k-1}}, Y_{t_k}) \log \frac{p(X_{t_k} | \mathcal{Z}_{t_{k-1}}, Y_{t_k})}{p(X_{t_k} | \mathcal{Z}_{t_{k-1}})} dX_{t_k}. \end{aligned}$$

However, computing KL divergence faces three challenges: 1) it requires the unknown posterior density that filtering aims to estimate; 2) it lacks closed-form expressions and requires integration; and 3) for event-triggered decisions at the sensor, the measurement information resides locally, yet KL divergence computation traditionally occurs at the remote estimator, preventing sensor-side evaluation. We propose a two-step approximation to address these issues. First, KL divergence is approximated via particles. Second, the estimator employs sampling and regression-based methods for approximation. Finally, only regression coefficients

are transmitted to the sensor, enabling local event-triggered decisions. The approximation process proceeds as follows.

Particle Approximation: According to the Bayes rule

$$p(X_{t_k} | \mathcal{Z}_{t_{k-1}}, Y_{t_k}) = \frac{p(Y_{t_k} | X_{t_k})p(X_{t_k} | \mathcal{Z}_{t_{k-1}})}{p(Y_{t_k} | \mathcal{Z}_{t_{k-1}})}$$

we have

$$\begin{aligned} f_k(Y_{t_k}) &= \int p(X_{t_k} | \mathcal{Z}_{t_{k-1}}, Y_{t_k}) \log \frac{p(X_{t_k} | \mathcal{Z}_{t_{k-1}}, Y_{t_k})}{p(X_{t_k} | \mathcal{Z}_{t_{k-1}})} dX_{t_k} \\ &= \int p(X_{t_k} | \mathcal{Z}_{t_{k-1}}, Y_{t_k}) \log \frac{p(Y_{t_k} | X_{t_k})}{p(Y_{t_k} | \mathcal{Z}_{t_{k-1}})} dX_{t_k} \\ &= \int p(X_{t_k} | \mathcal{Z}_{t_{k-1}}, Y_{t_k}) \log p(Y_{t_k} | X_{t_k}) dX_{t_k} \\ &\quad - \log p(Y_{t_k} | \mathcal{Z}_{t_{k-1}}) \int p(X_{t_k} | \mathcal{Z}_{t_{k-1}}, Y_{t_k}) dX_{t_k} \\ &= \int \frac{p(Y_{t_k} | X_{t_k}) p(X_{t_k} | \mathcal{Z}_{t_{k-1}})}{p(Y_{t_k} | \mathcal{Z}_{t_{k-1}})} \log p(Y_{t_k} | X_{t_k}) dX_{t_k} \\ &\quad - \log p(Y_{t_k} | \mathcal{Z}_{t_{k-1}}). \end{aligned}$$

Since

$$\begin{aligned} p(X_{t_k} | \mathcal{Z}_{t_{k-1}}) &\approx \frac{1}{N} \sum_{i=1}^N \delta_{X_{t_k}^i}(x) \\ p(Y_{t_k} | \mathcal{Z}_{t_{k-1}}) &= \int p(Y_{t_k} | X_{t_k}) p(X_{t_k} | \mathcal{Z}_{t_{k-1}}) dX_{t_k} \\ &\approx \frac{1}{N} \sum_{i=1}^N p(Y_{t_k} | X_{t_k}^i). \end{aligned}$$

$f_k(Y_{t_k})$ can be approximated by the following equation:

$$\begin{aligned} f_k(Y_{t_k}) &\approx \sum_{i=1}^N \frac{p(Y_{t_k} | X_{t_k}^i) \log p(Y_{t_k} | X_{t_k}^i)}{\sum_{i=1}^N p(Y_{t_k} | X_{t_k}^i)} \\ &\quad - \log \frac{1}{N} \sum_{i=1}^N p(Y_{t_k} | X_{t_k}^i). \end{aligned} \quad (11)$$

This particle-based approximation circumvents two computational challenges: 1) the posterior density is hard to calculate explicitly and 2) the KL divergence integral is analytically intractable for general nonlinear systems. The empirical distribution $\{X_{t_k}^i\}_{i=1}^N$ provides a consistent estimator of the KL divergence as $N \rightarrow \infty$.

Regression Fitting: We emphasize that the regression representation introduced as follows serves primarily as a practical mechanism enabling sensor-side triggering decisions under communication constraints, rather than as a new statistical criterion; the underlying design principle remains the KL divergence between prior and posterior. Before acquiring the current measurement, the estimator is unable to compute the KL divergence. If the estimator were to transmit all particle information to the sensor for computation of the KL divergence, the communication overhead would be prohibitively high, owing to the potentially large number and high dimensionality of particles. Furthermore,

evaluating the KL divergence entails substantial computational effort, which is impractical for sensors with limited processing capability. To address these challenges, sampling and computation are carried out at the estimator side to generate a set of data pairs $\{Y_k^m, f_k(Y_k^m)\}_{m=1}^M$. These data are then used to perform regression fitting. The estimator calculates regression coefficients $\{\beta_{r,k}^*\}_{r=1}^R$ by minimizing

$$\sum_{m=1}^M \|f_k(Y_k^m) - \boldsymbol{\varphi}(Y_k^m) \boldsymbol{\beta}_k\|^2 \quad (12)$$

where $\boldsymbol{\varphi}(\cdot) = (\varphi_1(\cdot), \dots, \varphi_R(\cdot))$ is a series of basis functions, and $\boldsymbol{\beta}_k = (\beta_{1,k}, \dots, \beta_{R,k})^T$ is the coefficient vector. Then, only the $\{\beta_{r,k}^*\}_{r=1}^R$ are transmitted to the sensor. Based on these information, the sensor can subsequently reconstruct the approximate KL divergence

$$\hat{f}_k(Y_k) = \sum_{r=1}^R \beta_{r,k}^* \varphi_r(Y_k) \quad (13)$$

using the current measurement Y_k and perform event-triggered decisions.

D. Event-Triggered Continuous-Discrete Feedback Particle Filter

Under the CD system (1) and (2) and employing the KL divergence-based ETM, the event-triggered FPF with intermittent observation is as follows:

$t \in [t_{k-1}, t_k)$:

$$dX_t^i = A(X_t^i)dt + \sigma_B dB_t^i \quad (14)$$

$t = t_k : \text{if } \zeta_k = 1$

$$\begin{aligned} \frac{dS_k^i(\tau)}{d\tau} &= \mathcal{K}_k(S_k^i(\tau), \tau) \left(Y_{t_k} - \frac{H(S_k^i(\tau)) + \hat{H}}{2} \right) \\ &\quad + \frac{1}{2} \Omega_k(S_k^i(\tau), \tau) \end{aligned} \quad (15)$$

1) $\hat{f}_k(Y_k) \geq \gamma$, i.e., $\zeta_k = 1 : X_{t_k}^i = S_k^i(1)$; and 2) $\hat{f}_k(Y_k) < \gamma$, i.e., $\zeta_k = 0 : X_{t_k}^i = S_k^i(0)$. The proposed event-triggered CD-FPF operates as a standard CD filtering problem when triggered ($\zeta_k = 1$); whereas when not triggered ($\zeta_k = 0$), it locally reduces to a prediction-only mode, constituting a subobservation system with fewer observation nodes. This framework can be viewed as an extension of the work in [25] to an active ETM. Adaptive switching between prediction and filtering modes is driven by evaluating the KL divergence between the prior and posterior distributions, enabling seamless transitions across subsystems of varying observational complexity.

LEMMA 1 (SEE [23]) Given $p_k(x, 0) = p_k^*(x, 0)$ and taking $\mathcal{K}_k(\cdot)$ and $u_k(\cdot)$ from (7) and (8), then for any $\tau \in [0, 1]$, we have $p_k(x, \tau) = p_k^*(x, \tau)$.

REMARK 1 Lemma 1 demonstrates that the control input $U_k^i(\tau)$ (or $\{\mathcal{K}_k, u_k\}$) is optimal if it is designed such that the posterior distribution of $S_k(1)$ equals to the actual posterior distribution.

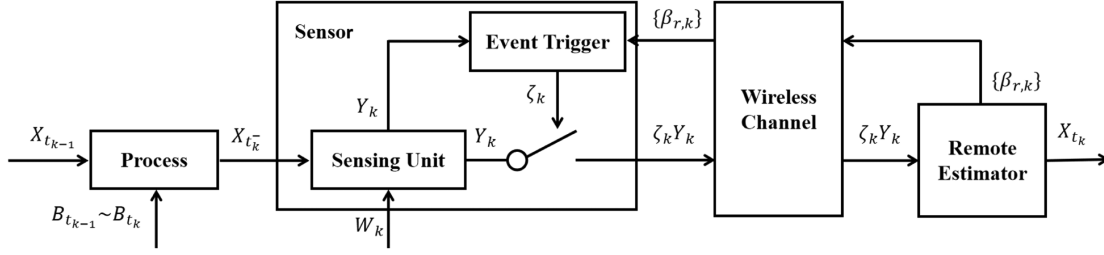


Fig. 2. Event-triggered FPF in CD system.

REMARK 2 When measurements are retained, pseudo-time iterative updates are performed; when discarded, updates are omitted and prediction follows the continuous-time dynamics. This event-triggered CD-FPF effectively selects a reduced model within the original framework, achieving dual objectives: reducing communication costs while preserving filtering performance. The inherent CD structure is maintained throughout.

REMARK 3 (COST DECOMPOSITION) The per-observation computational and communication cost distributes as follows. At the sensor side, evaluating $\hat{f}_k(Y_k) = \sum_{r=1}^R \beta_{r,k}^* \varphi_r(Y_k)$ costs $O(R)$ scalar multiplications, plus the transmission of Y_k when $\zeta_k = 1$. At the estimator side, the regression fit of Section III-C costs $O(MR^2 + R^3)$ per observation and is paid *whether or not* a trigger fires; the pseudo-time FPF update of complexity $O(N \cdot N_\tau)$ is paid *only when a trigger fires*; in addition, the estimator broadcasts the R coefficients $\{\beta_{r,k}^*\}$ to the sensor at every step. Since the estimator is not power-constrained and R is small, the binding communication cost is the battery-limited sensor uplink, i.e., the transmission of Y_k . The sensor-side transmission overhead is therefore reduced at all triggering rates, while the global computational cost decreases only when the average triggering rate falls below the break-even point at which the saved pseudo-time updates dominate the regression overhead. For the nonlinear system considered in Section IV, the break-even event-triggered rate is approximately 0.83, which is below unity. This indicates that the event-triggered implementation yields a net computational saving when the average triggering rate remains below this value and the wall-clock cost decreases monotonically with the triggering threshold.

E. Summary of the Proposed Algorithm

This section provides an overview of the proposed event-triggered FPF algorithm for CD systems. The detailed procedure of the algorithm is illustrated in Fig. 2. To implement the event-triggered FPF under a general CD Gaussian system, certain approximation processes are also required. First, the Euler–Maruyama discretization method is applied to both t and τ to implement the event-triggered CD-FPF corresponding to (14) and (15). In addition, at pseudo-time τ , the posterior mean and covariance can be approximated

by particles, i.e.,

$$\mathcal{K}_k(\tau) = \frac{1}{N} \sum_{i=1}^N S_k^i(\tau) (H(S_k^i(\tau)) - \hat{H}(\tau))^T \quad (16)$$

$$\Omega_k(\tau) = \frac{1}{N} \sum_{i=1}^N S_k^i(\tau) (G(S_k^i(\tau)) - \hat{G}(\tau))^T \quad (17)$$

where

$$\begin{aligned} \hat{H}(\tau) &= \frac{1}{N} \sum_{i=1}^N H(S_k^i(\tau)) \\ \hat{G}(\tau) &= |\hat{H}(\tau)|^2 - |\hat{H}|^2(\tau) \\ |\hat{H}|^2(\tau) &= \frac{1}{N} \sum_{i=1}^N H^T(S_k^i(\tau)) H(S_k^i(\tau)). \end{aligned}$$

In summary, the specific procedures of the proposed algorithm are outlined in Algorithm 1.

IV. SIMULATION

In this section, we will compare the FPF using KL-based ETM with the PF using KL-based ETM in both CD linear Gaussian systems and nonlinear Gaussian systems. It is important to mention that the following simulation was conducted on a laptop equipped with an Intel Core i7-10710 U with integrated graphics card 1.10 GHz CPU.

A. Simulation Setup

In the experiments, in order to compare the performances, we introduce the root mean squared error (RMSE) defined as follows:

$$\text{RMSE} = \sqrt{\frac{1}{K+1} \sum_{k=0}^K (X_{k\Delta t} - \hat{X}_{k\Delta t})^2}$$

where $K = \frac{T}{\Delta t}$. In addition, the event-triggered rate refers to the ratio between the number of transmitted observations and the total number of observations. According to the definition, it is readily seen that as the threshold increases, the triggering rate decreases accordingly. Since the test function f_k is a (nonnegative) KL divergence, any threshold $\gamma \leq 0$ yields the always-triggered baseline, i.e., the original CD-FPF. In the second-step approximation of the KL divergence, a radial basis function (RBF) ridge regression fitting is used. The basis function adopts a Gaussian kernel

Algorithm 1: Event-Triggered FPF for Continuous-Discrete Systems.

```

1: Initialization:
2: Set  $k = 0$ 
3: Sample  $X_0^i$  from  $p^*(x, 0)$ 
4: Set  $t = 0$ 
5:  $k = k + 1$ 
6: Loop 1: Continuous-time prediction
7: Set  $t = t_{k-1}$ 
8: while  $t \leq t_k$  do
9:    $dX_t^i = A(X_t^i) dt + \sigma_B dB_t^i$ 
10:   $t = t + \Delta t$ 
11: end while
12: Event-trigger condition:
13: Set  $\tau = 0$ 
14: Set  $S_k^i(0) = X_t^i$ 
15: Generate data pairs  $\{Y_k^m, f_k(Y_k^m)\}_{m=1}^M$ 
16: Calculate  $\{\beta_{r,k}^*\}_{r=1}^R$ 
17: Remote estimator sends  $\{\beta_{r,k}^*\}_{r=1}^R$  to sensor
18: Sensor calculates  $\hat{f}_k(Y_k) = \sum_{r=1}^R \beta_{r,k}^* \varphi_r(Y_k)$ 
19: if  $\hat{f}_k(Y_k) < \gamma$  then
20:   Sensor doesn't send  $Y_k$  to remote estimator
21:   Set  $X_t^i = S_k^i(0)$ 
22: else
23:   Sensor sends  $Y_k$  to remote estimator
24:   Loop 2: Discrete-time update
25:   while  $\tau \leq 1$  do
26:      $\frac{dS_k^i}{d\tau} = \mathcal{K}_k[Y_k - \frac{1}{2}(H + \hat{H})] + \frac{1}{2}\Omega_k$ 
27:      $\tau = \tau + \Delta\tau$ 
28:   end while
29:   Set  $X_t^i = S_k^i(1)$ 
30: end if
31:  $k = k + 1$ 
32: if  $t < T$  then
33:   Go to Loop 1
34: else
35:   return  $\{X_t^i\}$  and stop
36: end if

```

form

$$\varphi_r(y) = \exp\left(-\frac{1}{2} \frac{\|y - c_r\|_2^2}{\sigma_{\text{rbf}}^2}\right), \quad r = 1, \dots, R$$

where c_r represents the center of the r th RBF, which is selected from the set of test points, and σ_{rbf} is the kernel width. In addition, the number of basis functions is R , the ridge regression regularization coefficient is λ_{ridge} , and the test points are generated as follows:

$$Y_k^m = H(X_{t_k}^{i_m}) + \sigma_W W_k^m, \quad m = 1, \dots, M$$

where $W_k^m \sim \mathcal{N}(0, I_m)$, the number of test points is M , and $X_{t_k}^{i_m}$ is randomly selected from $\{X_{t_k}^i\}_{i=0}^N$. The general parameters for the numerical simulations are configured as follows. The whole time interval is $T = 10$, and the discrete observation instant is $t_k = k\Delta t_{\text{true}}$ with $\Delta t_{\text{true}} = 0.1$. Use

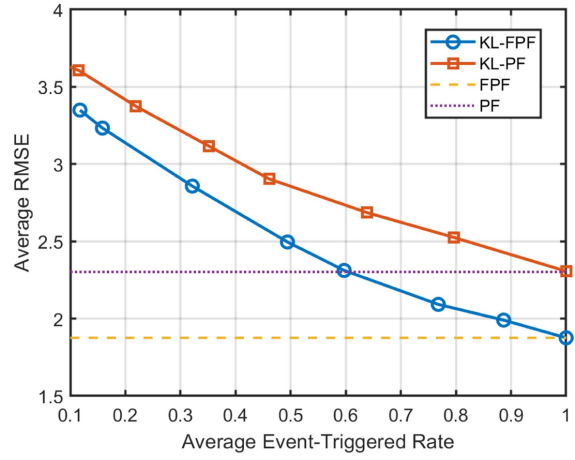


Fig. 3. Average RMSE of CD-FPF under KL-based ETM and CD-PF under KL-based ETM.

the Euler's method in time discretization with real time step $\Delta t = 0.01$. In addition, the number of Monte Carlo simulations is $N_{\text{MC}} = 60$.

B. Linear System

Consider the CD linear Gaussian system with independent noises

$$\begin{cases} dX_t = AX_t dt + \sigma_B dB_t \\ Y_k = HX_{t_k} + \sigma_W W_k \end{cases} \quad (18)$$

where $X_0 \sim \mathcal{N}(0, I_{10})$, the process noise is modeled by a standard Brownian motion $\{B_t\}$, the measurement noise $\{W_k\}$ is a standard Gaussian white noise sequence, $\sigma_B = 1$, $\sigma_W = 1$, $H = I_{10 \times 10}$, and $A = [a_{ij}(t)]_{10 \times 10}$ with elements

$$a_{ij}(t) = \begin{cases} 0.1, & \text{if } i + 1 = j \\ -0.3, & \text{if } i = j \\ 0, & \text{otherwise.} \end{cases}$$

Use the Euler's method in time discretization with pseudo-time step $\Delta\tau = 0.05$. The number of particles is $N = 20$. And threshold γ is selected from $\{0, 0.003, 0.01, 0.03, 0.05, 0.1, 0.2, 0.3\}$. In addition, the kernel width is $\sigma_{\text{rbf}} = 1$, the number of basis functions is $R = 20$, the ridge regression regularization coefficient is $\lambda_{\text{ridge}} = 10^{-4}$, $W_k^m \sim \mathcal{N}(0, I_{10})$, and the number of test points is $M = 60$. Fig. 3 shows the relationship between the average RMSE and the average event-triggered rate. Results indicate that the event-triggered CD-FPF exhibits acceptable estimation performance at each event-triggered rate, and consistently outperforms the event-triggered CD-PF. As the average event-triggered rate increases, its performance progressively approaches that of the original CD-FPF and essentially coincides with it near full triggering. In addition, when the triggering rate exceeds 60%, the event-triggered CD-FPF already achieves a performance level comparable to that of the original CD-PF.

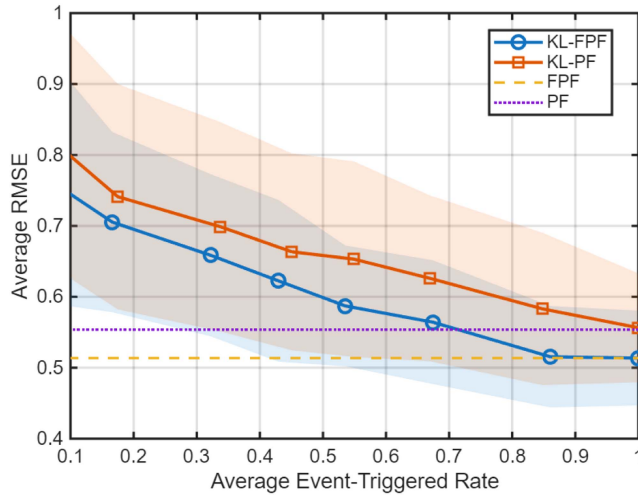


Fig. 4. Average RMSE of CD-FPF under KL-based ETM and CD-PF under KL-based ETM.

C. Nonlinear System

Consider the CD nonlinear Gaussian system with independent noises

$$\begin{cases} dX_t = A(X_t)dt + \sigma_B dB_t \\ Y_{t_k} = H(X_{t_k}) + \sigma_W W_k \end{cases} \quad (19)$$

where $X_0 \sim \mathcal{N}(0, I_4)$, the process noise is modeled by a standard Brownian motion $\{B_t\}$, the measurement noise $\{W_k\}$ is a standard Gaussian white noise sequence, $\sigma_B = 0.2$, $\sigma_W = 1$, and

$$\begin{aligned} H(x) &= x + 0.1x^{\odot 3} \\ A(x) &= Ax + 0.2\sin(x) \end{aligned}$$

where $x^{\odot 3}$ denotes the elementwise cube of the vector and

$$A = \begin{bmatrix} -0.3 & 0.1 & 0 & 0 \\ 0 & -0.3 & 0.1 & 0 \\ 0 & 0 & -0.3 & 0.1 \\ 0 & 0 & 0 & -0.3 \end{bmatrix}.$$

The number of pseudo-time steps is $N_t = 150$, the number of particles is $N = 50$, and the threshold γ is selected from $\{-0.1, 0.01, 0.03, 0.05, 0.07, 0.1, 0.18, 0.35\}$. In addition, the kernel width is $\sigma_{rbf} = 1$, the number of basis functions is $R = 8$, the ridge regression regularization coefficient is $\lambda_{\text{ridge}} = 10^{-4}$, $W_k^m \sim \mathcal{N}(0, I_4)$, and the number of test points is $M = 35$.

Fig. 4 shows the relationship between the average RMSE and the average event-triggered rate. Results indicate that the event-triggered CD-FPF also exhibits acceptable estimation performance at each event-triggered rate, and consistently outperforms the event-triggered CD-PF in the nonlinear system. When the average event-triggered rate exceeds 65.0%, its performance becomes very close to that of the original CD-FPF. For example, with a 67% event-triggered rate, only 9.8% accuracy degradation occurs. In addition, when the triggering rate exceeds 70%, the event-triggered CD-FPF already achieves a performance

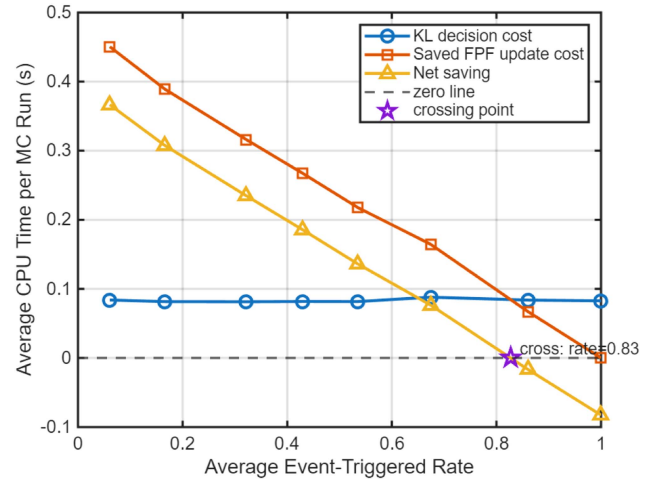


Fig. 5. Average CPU time per MC run of CD-FPF under KL-based ETM, decomposed into the KL-decision cost, the saved FPF update cost, and the net saving, with the break-even crossing point.

level comparable to that of the original CD-PF. Fig. 5 shows the average CPU time per Monte Carlo run of the CD-FPF under the KL-based ETM. The KL decision cost remains nearly constant because the trigger criterion is evaluated at every observation time, whereas the saved FPF update cost decreases as the event-triggered rate increases. The net saving is positive when the triggering rate is below approximately 83%, indicating that the skipped pseudo-time updates can compensate for the KL decision overhead in this range. Beyond this crossing point, the additional KL overhead dominates the saved update cost. Moreover, even in linear systems, the PF suffers from the curse of dimensionality, and this effect becomes more significant in nonlinear systems, whereas the FPF can overcome it [26]. A comparative performance analysis between the FPF and PF demonstrates that the PF requires a significantly larger number of particles than the FPF [27].

V. CONCLUSION

An event-triggered FPF for general CD Gaussian systems has been investigated in this article. An ETM based on KL divergence has been integrated into the CD-FPF framework and approximated in two steps so that the trigger decision can be evaluated at the sensor end while the regression coefficients required by this evaluation are computed at the estimator. The proposed method provides an integrative extension of CD-FPF to event-triggered measurement assimilation while preserving the original prediction-update structure. When the latest observation is not retained for transmission, the update is truncated, reducing the sensor-side transmission overhead at all triggering rates and reducing the global computational cost whenever the triggering rate falls below the break-even point characterized in Remark 3. Numerical experiments demonstrate the effectiveness of the proposed method, showing that when the average event-triggered rate exceeds 0.7, the performance of the event-triggered CD-FPF closely approximates that of the original filter.

CHANG QIN 
TONG GUO 
 Chinese Academy of Sciences, Chengdu, China
 University of Chinese Academy of Sciences,
 Beijing, China

JIAYI KANG 
 Hetao Institute of Mathematics and
 Interdisciplinary Sciences (HIMIS), Shenzhen,
 China
 Beijing Institute of Mathematical Sciences and
 Applications (BIMSA), Beijing, China

YAO MAO , Senior Member, IEEE
 Chinese Academy of Sciences, Chengdu, China
 University of Chinese Academy of Sciences,
 Beijing, China

STEPHEN S.-T. YAU , Life Fellow, IEEE
 Tsinghua University, Beijing, China

REFERENCES

- [1] H. W. Sorenson, *Kalman Filtering: Theory and Application*. New York, NY, USA: IEEE Press, 1985.
- [2] R. E. Kalman, "A new approach to linear filtering and prediction problems," *J. Basic Eng.*, vol. 82, no. 1, pp. 35–45, Mar. 1960.
- [3] R. E. Larson, R. M. Dressler, and R. S. Ratner, "Application of the extended Kalman filter to ballistic trajectory estimation," *Stanford Res. Inst.*, Menlo Park, CA, USA, Final Rep. 5188-103, 1967.
- [4] S. J. Julier and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," *Proc. SPIE*, vol. 3068, pp. 182–193, 1997.
- [5] I. Arasaratnam and S. Haykin, "Cubature Kalman filters," *IEEE Trans. Autom. Control*, vol. 54, no. 6, pp. 1254–1269, Jun. 2009.
- [6] R. S. Bucy and P. D. Joseph, *Filtering for Stochastic Processes With Applications to Guidance*. Providence, RI, USA: Amer. Math. Soc., 2005.
- [7] M. Mallick, M. Morelande, and L. Mihaylova, "Continuous-discrete filtering using EKF, UKF, and PF," in *Proc. 15th Int. Conf. Inf. Fusion*, Jul. 2012, pp. 2180–2187.
- [8] G. Y. Kulikov and M. V. Kulikova, "Accurate continuous-discrete unscented Kalman filtering for estimation of nonlinear continuous-time stochastic models in radar tracking," *Signal Process.*, vol. 139, pp. 25–35, Oct. 2017.
- [9] M. S. Grewal, L. R. Weill, and A. P. Andrews, *Global Positioning Systems, Inertial Navigation, and Integration*. Hoboken, NJ, USA: Wiley, 2007.
- [10] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*. New York, NY, USA: Academic Press, 1970.
- [11] P. Frogerais, J.-J. Bellanger, and L. Senhadji, "Various ways to compute the continuous-discrete extended Kalman filter," *IEEE Trans. Autom. Control*, vol. 57, no. 4, pp. 1000–1004, Apr. 2012.
- [12] T. Knudsen and J. Leth, "A new continuous-discrete unscented Kalman filter," *IEEE Trans. Autom. Control*, vol. 64, no. 5, pp. 2198–2205, May 2019.
- [13] Y. Xia, Z. Deng, L. Li, and X. Geng, "A new continuous-discrete particle filter for continuous-discrete nonlinear systems," *Inf. Sci.*, vol. 242, pp. 64–75, May 2013.
- [14] A. Doucet, N. de Freitas, and N. Gordon, Eds., *Sequential Monte Carlo Methods in Practice*. New York, NY, USA: Springer-Verlag, Apr. 2001.
- [15] T. Yang, P. G. Mehta, and S. P. Meyn, "Feedback particle filter," *IEEE Trans. Autom. Control*, vol. 58, no. 10, pp. 2465–2480, Oct. 2013.
- [16] T. Yang, H. A. P. Blom, and P. G. Mehta, "The continuous-discrete time feedback particle filter," in *Proc. Amer. Control Conf.*, Jun. 2014, pp. 648–653.
- [17] Q. Liu, Z. Wang, X. He, and D. Zhou, "Event-based recursive distributed filtering over wireless sensor networks," *IEEE Trans. Autom. Control*, vol. 60, no. 9, pp. 2470–2475, Sep. 2015.
- [18] N. Sadeghzadeh-Nokhodberiz, M. Davoodi, and N. Meskin, "Stochastic event-triggered particle filtering for state estimation," in *Proc. 2nd Int. Conf. Event-Based Control, Commun., Signal Process.*, Krakow, Poland, Jun. 2016, pp. 1–6.
- [19] W. Li, Y. Jia, J. Du, and J. Zhang, "An information aware event-triggered scheme for particle filter based remote state estimation," *Automatica*, vol. 103, pp. 151–158, May 2019.
- [20] J. Ruuskanen and A. Cervin, "Event-based state estimation using the auxiliary particle filter," in *Proc. Eur. Control Conf.*, Jun. 2019, pp. 315–320.
- [21] X. Lv, Z. Wang, D. Zhou, and H. Liu, "Stochastic event-triggered variational Bayesian filtering," *IEEE Trans. Autom. Control*, vol. 68, no. 7, pp. 4321–4328, Jul. 2023.
- [22] K. Zhu, Y. Gu, and J. Xu, "Set-membership filtering for two-dimensional systems with dynamic event-triggered mechanism," *Automatica*, vol. 143, 2022, Art. no. 110416.
- [23] X. Chen, J. Kang, and S. S.-T. Yau, "Continuous discrete optimal transportation particle filter," *Asian J. Math.*, vol. 29, no. 1, pp. 51–76, Jan. 2025.
- [24] Z. Hu, B. Chen, R. Wang, and L. Yu, "Remote state estimation with posterior-based stochastic event-triggered schedule," *IEEE Trans. Autom. Control*, vol. 69, no. 2, pp. 1194–1201, Feb. 2023.
- [25] J. Kang, X. Jiao, and S. S.-T. Yau, "Estimation of the linear system via optimal transportation and its application for missing data observations," *IEEE Trans. Autom. Control*, vol. 70, no. 9, pp. 5644–5659, Sep. 2025.
- [26] A. Taghvaei and P. G. Mehta, "An optimal transport formulation of the ensemble Kalman filter," *IEEE Trans. Autom. Control*, vol. 66, no. 7, pp. 3052–3067, Jul. 2021.
- [27] J. Kang, X. Chen, Y. Tao, and S. S.-T. Yau, "Optimal transportation particle filter for linear filtering systems with correlated noises," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 58, no. 6, pp. 5190–5203, Dec. 2022.
- [28] D. Selvi and G. Battistelli, "Distributed Kalman filtering with adaptive communication," *IEEE Control Syst. Lett.*, vol. 9, pp. 15–20, 2025.
- [29] D. Deng and J. Xiong, "Kalman-like filter for event-triggered remote state estimation over an additive noise channel," *IEEE Control Syst. Lett.*, vol. 8, pp. 205–210, 2024.
- [30] B. Øksendal, *Stochastic Differential Equations: An Introduction With Applications*. Berlin, Germany: Springer, 2013.
- [31] J. M. Reason and J. M. Rabaey, "A study of energy consumption and reliability in a multi-hop sensor network," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 8, no. 1, pp. 84–97, 2004.