IEEE Author Portal
IEEE Transactions on Signal Processing

Sponsoring Society:
IEEE Signal Processing Society
IEEE

Regular Paper

Use Extended Kalman Filter to Interpret and Improve Learning Rate

Submission ID 40c62d9c-5b44-4808-b0fc-94a8a57fa0aa

Submission Version Initial Submission

PDF Generation 13 Jul 2025 03:50:18 EST by Atypon ReX

Authors

Mr. Weisheng Chen
Submitting Author

ORCiD
https://orcid.org/0009-0000-1525-5549

Affiliations
- School of Sciences, Harbin Institute of Technology (Shenzhen), Shenzhen, Guangdong, China

Mr. Changan Liu

ORCiD
https://orcid.org/0009-0001-3437-468X

Affiliations
- School of Sciences, Harbin Institute of Technology (Shenzhen), Shenzhen, Guangdong, China

Dr. Yunqi Chen

Affiliations
- School of Sciences, Harbin Institute of Technology (Shenzhen), Shenzhen, Guangdong, China

Prof. Zhibin Yan
Corresponding Author

ORCiD
https://orcid.org/0000-0001-5551-5202

Affiliations
- School of Sciences, Harbin Institute of Technology (Shenzhen), Shenzhen, Guangdong, China

Additional Information

EDICS

ASP ADAPTIVE SIGNAL PROCESSING / 2. ASP-APPL Applications of adaptive filters

2. ASP-APPL Applications of adaptive filters

SPC SIGNAL PROCESSING FOR COMMUNICATIONS / 109. SPC-APPL Applications involving signal processing for communications

109. SPC-APPL Applications involving signal processing for communications

SSP STATISTICAL SIGNAL PROCESSING / 148. SSP-IDEN System identification

148. SSP-IDEN System identification

SSP STATISTICAL SIGNAL PROCESSING / 152. SSP-PARE Parameter estimation

152. SSP-PARE Parameter estimation

Subject Category

ADAPTIVE SIGNAL PROCESSING

DESIGN AND IMPLEMENTATION OF SIGNAL PROCESSING SYSTEMS

MACHINE LEARNING

OPTIMIZATION METHODS FOR SIGNAL PROCESSING

SIGNAL PROCESSING FOR COMMUNICATIONS

STATISTICAL SIGNAL PROCESSING

Is this manuscript a resubmission of, or related to, a previously rejected manuscript, or a previously reviewed and withdrawn manuscript?

No

Is this manuscript an extended version of a conference publication (or conference article accepted for publication)?

No

Is this manuscript related to any other papers of the authors that are either published, accepted for publication, or currently under review, and that are not included among the references cited in the manuscript?

No

Are there any preprints of the manuscript (i.e. preprints that are identical to the submitted manuscript, except for minor differences) that have been posted on the authors' personal website, employer's website or institutional repository, arXiv.org, TechRxiv.org, or on any not-for-profit preprint server approved by the IEEE?

No

Are there any other posted preprints that should not be considered to be prior art?

No

Explain in detail why the contribution of this manuscript is within the scope of the IEEE Transactions on Signal Processing?

1. Theory: New insight on EKF gain as an adaptive learning rate governed by covariance matrices (P,Q,R), an important topic in parameter estimation.

2. Algorithm: We formalize FDEKF via a state-space model and propose the novel RR-FDEKF for high-data-rate task, a key challenge in adaptive filtering.

3. Application: Our proposed algorithm outperforms the Adam optimizer, achieving lower error and faster convergence in PIM cancellation (signal processing for communication).

Why is the contribution significant (What impact will it have)?

1. Theory: We reframe the EKF as an interpretable optimization framework, a theoretically-grounded alternative to heuristic methods.

2. Implementation: The novel RR-FDEKF significantly reduces complexity, making EKF-based approach viable for large-scale, real-time system identification.

3. Application: We solve the critical PIM cancellation challenge in communication, showing substantial engineering value through its superior performance against a strong baseline.

What are the three papers in the published literature most closely related to this paper?

Note: To provide key details within the 500-character limit, citing full titles and DOIs:

1. "Decoupled extended kalman filter training of feedforward layered networks" DOI: 10.1109/IJCNN.1991.155276

2. "Training of convolutional neural networks for image classification with fully decoupled extended kalman filter" DOI: 10.3390/a17060243

3. "Symmetrized Basis Function Approximation Network for Passive Intermodulation Cancellation" DOI: 10.1109/TCOMM.2025.3560338

What is distinctive/new about the current paper relative to these previously published works?

1. Relative to Puskorius '91: We provide a rigorous state-space derivation for the FDEKF and then propose the novel RR-FDEKF with its sequential update mechanism.

2. Unlike Gaytan '24's application, we provide new insight on EKF gain as an adaptive learning rate and propose the novel, superior RR-FDEKF.

3. Complementary to Liu '25: Their work focused on proposing the PIM model architecture. Our contribution is a new, computationally efficient parameter estimation method for their model.

Files for peer review

All files submitted by the author for peer review are listed below. Files that could not be converted to PDF are indicated; reviewers are able to access them online.

Name	Type of File	Size	Page
Use_Extended_Kalman_Filter_to_Interpret_and_Improve_Learning_Rate__submit.pdf	Main Document - PDF	1.5 MB	Page 5

Use Extended Kalman Filter to Interpret and Improve Learning Rate

Weisheng Chen, Changan Liu, Yunqi Chen, Zhibin Yan

***Abstract*—This paper clarifies the physical interpretation of learning rate in nonlinear system identification using the extended Kalman filter (EKF). To handle applications with rapidly arriving data, the fully decoupled extended Kalman filter (FDEKF) is improved into a new one, called Round-Robin fully decoupled extended Kalman filter (RR-FDEKF). This algorithm features a Round-Robin mechanism in parameter update, which utilizes a specially constructed mixed-information state vector to immediately use new measurement information. The effectiveness of both FDEKF and RR-FDEKF are demonstrated using experimental data in passive intermodulation (PIM) cancellation, a challenging real-world task in wireless communication. They achieve superior PIM suppression with significant computational advantages compared to the Adam optimizer, a recently highly recognized estimation strategy in deep learning. These findings highlight the potential of FDEKF and RR-FDEKF as efficient and high-performance solutions for online nonlinear parameter estimation.**

***Index Terms*—System Identification, Extended Kalman Filter, Adam, Deep learning, Passive Intermodulation.**

I. Introduction

SYSTEM identification is the process of constructing mathematical models of dynamic systems, and the subsequent estimation of model parameters from observed data. fundamental challenges across numerous scientific and engineering fields, including control systems, signal processing, econometrics, and machine learning [1], [2]. While the architecture of a model establishes the upper bound on achievable model performance, parameter estimation determines the lower bound, or the actual fidelity of the identified model.

A prominent example demanding advanced system identification arises in wireless communications with the mitigation of passive intermodulation (PIM) distortion. PIM generates unwanted interference signals due to nonlinearities in passive components, significantly degrading receiver sensitivity, especially in co-located transmit/receive systems or multi-carrier scenarios [3], [4]. Digital PIM cancellation technique offers a flexible solution but hinges on accurately identifying the underlying nonlinear dynamics from observed signals. Capturing the complex behavior and memory effects inherent in PIM often necessitates sophisticated, high-dimensional nonlinear models, such as Volterra series or basis function networks [5], [6]. Estimating the large number of parameters in these models efficiently and accurately presents a significant challenge, motivating the search for suitable estimation frameworks.

The Kalman filter framework is a cornerstone for state and parameter estimation, providing the optimal estimator in the mean-squared error sense for linear systems subject to Gaussian noise [7]–[9]. Further, as many real-world systems exhibit nonlinear dynamics or measurement relationships, the extended Kalman filter (EKF) was introduced [10]–[12]. The EKF addresses nonlinearity by linearizing the system around the current state estimate at each time step, enabling the application of the prediction-update mechanism in the Kalman filter. Its effectiveness has been proven in diverse applications, from navigation [13] to biological systems [10], [14]–[21].

Despite its widespread use, the practical application of the EKF faces significant hurdles, particularly for high-dimensional systems. When applied to system identification, the number of parameters, denoted by n in following, adds to the dimension of the system. Since the number n can be very large, the identification task suffers from considerable computational complexity, typically scaling quadratically or worse with n due to operations involving the $n \times n$ covariance matrix. Furthermore, it encounters potential numerical instability issues, especially for high nonlinearity or poor initializations [22]. These limitations motivate the development of more efficient and robust schemes of applying the EKF in system identification.

A. Motivations

To overcome the computational and numerical challenges of the EKF, various approximation and simplification strategies have been proposed. A prominent direction involves decoupling, which aims to reduce complexity by simplifying the structure of the error covariance matrix, often by assuming independence or limited correlation between different states or parameters.

The initial drive towards decoupling in the context of neural network training led to the independent EKF (IEKF) philosophy. Pioneering works, such as those by Kollias and Anastassiou [23] and Shah and Palmieri with their MEKA algorithm [24], proposed assigning independent filters to each neuron's weights. These methods prioritized complete computational locality, performing updates locally and eliminating the need for a global coordination matrix, thereby achieving significant speedups. Building upon these early explorations, Puskorius and Feldkamp [25] introduced a different formulation, which they termed the decoupled EKF (DEKF). While acknowledging the efficiency of IEKF, their approach aimed

This work was supported in part by HUAWEI under the project TC20210609011, "PIM digital modeling based on multi-dimensional nonlinear System", and by NSF of China under Grant 62273056.

The authors are with School of Sciences, Harbin Institute of Technology (Shenzhen), Shenzhen, Guangdong, China (e-mail: 23s058018@stu.hit.edu.cn; 22b358003@stu.hit.edu.cn; zbyan@hit.edu.cn; cyq09180@163.com).

for greater theoretical rigor by retaining a degree of indirect global coupling. Their framework, even in its fully decoupled limit, coordinates updates through a shared global scaling matrix that incorporates information from all parameters. This was intended to yield more stable and accurate performance compared to the purely local IEKF approaches.

In this paper, we revisit and formalize the original IEKF philosophy, focusing on its most granular and computationally efficient realization. We adopt a formulation where each individual parameter is updated by a self-contained, scalar EKF. Hereafter, we refer to this specific, fully local variant as the fully decoupled extended Kalman filter (FDEKF), while acknowledging its conceptual distinction from the globally-coordinated version of Puskorius and Feldkamp. The principle of such independent estimation has also been explored in other contexts [26], [27]. The decoupling strategy itself has seen a resurgence in interest, with FDEKF-like approaches being successfully applied in modern, computationally intensive domains like deep neural network training [21]. This historical trajectory highlights a recurring theme: algorithmic approximations developed to address past computational limits often find renewed relevance as problem scales grow and computing power evolves.

The practical success of these decoupling strategies motivates a clear, formal derivation from a state-space perspective. While foundational insights exist in early works [23], [24], our work contributes a self-contained and direct derivation for the FDEKF. This provides a solid basis for systematic analysis and, as a direct extension, allows us to introduce a novel variation called the Round-Robin fully decoupled extended Kalman filter (RR-FDEKF).

B. Contributions

The primary contributions are summarized as follows:

1) **Interpretable Adaptive Learning Rate Mechanism of the EKF:** This work proposes a novel framework for interpreting the EKF's update mechanism, casting the Kalman gain as an interpretable, self-adaptive learning rate. We elucidate how the interplay between the process noise ($\mathbf{Q}_k$), measurement noise ($\mathbf{R}_k$), and the estimated parameter uncertainty ($\mathbf{P}_k$) dynamically tunes the filter's learning rate. This provides a novel contrast to modern optimizers like Adam, whose adaptive mechanisms are primarily heuristic and lack this direct physical grounding.
2) **Formalization of FDEKF and Proposal of RR-FDEKF:** We first establish a rigorous theoretical foundation for the FDEKF by deriving it from state-space model, clarifying its underlying assumptions. Building on this clear foundation, we then introduce the RR-FDEKF, a novel algorithm that improves upon the decoupled extended Kalman filter approach. By updating only a single parameter per measurement in a sequential, round-robin fashion, RR-FDEKF is specifically designed to be advantageous for scenarios where the data arrival rate is high relative to the available computational resources, especially when dealing with a large number of parameters.
3) **High-Performance Digital PIM Cancellation via Efficient System Identification:** We successfully address the challenging task of digital PIM cancellation by developing an efficient identification scheme based on decoupled Kalman filtering. Experimental results using real-world data demonstrate that our approach yields superior PIM suppression and faster convergence compared to the widely-used Adam optimizer, underscoring its practical efficacy for complex, real-world communication systems.

C. Organization

The remainder of this paper is organized as follows. Section II establishes the theoretical foundation of our work. It begins by formulating the parameter estimation problem, then re-examines it from the EKF perspective, and culminates in our novel interpretation of the filter's core covariance matrices as the mechanism behind its adaptive learning rate. Building on this foundation, Section III introduces two computationally efficient algorithms. We first derive the FDEKF based on state-space model and then present our proposed RR-FDEKF, designed for high-efficiency, online applications. Section IV provides a comprehensive experimental validation of the proposed method. We detail the experimental setup, define the performance metrics, and present a comparative analysis of the results. Finally, Section V concludes the paper by summarizing our key findings and suggesting potential avenues for future research.

D. Notations

Throughout this paper, we adhere to the following notational conventions:

- Matrices are represented by boldface capital letters (e.g., $\mathbf{P}$, $\mathbf{Q}$, $\mathbf{R}$). The Kalman gain, $\mathbf{K}_k$, is also denoted this way as it is a matrix in the general case.
- Vectors are column vectors denoted by boldface lowercase letters (e.g., $\boldsymbol{\theta}$, $\mathbf{x}$, $\mathbf{y}$). Row vectors, if not covered by other conventions, may also follow this format or be represented as transposes (e.g., $\mathbf{h}^{\mathrm{T}}$).
- Scalars are represented by normal (non-bold) lowercase letters (e.g., n, p, q, t, k).
- The transpose of a matrix or vector is denoted by the superscript $^{\mathrm{T}}$ (e.g., $\mathbf{P}^{\mathrm{T}}$, $\boldsymbol{\theta}^{\mathrm{T}}$).
- A right subscript indicates the discrete time step (e.g., $\boldsymbol{\theta}_k, \mathbf{y}_k$).
- Estimates are denoted with a hat $\hat{(\cdot)}$. The notation $\hat{\boldsymbol{\theta}}_{k|j}$ represents the estimate of $\boldsymbol{\theta}$ at time k given measurements up to time j. For brevity, the updated estimate $\hat{\boldsymbol{\theta}}_{k|k}$ and its covariance $\mathbf{P}_{k|k}$ are often written simply as $\hat{\boldsymbol{\theta}}_k$ and $\mathbf{P}_k$.
- For the decoupled algorithms, P_k^i denotes the scalar variance of the i-th parameter.
- The innovation or measurement residual is denoted by $\tilde{\mathbf{y}}_k$. For the RR-FDEKF variant, which uses a dual index of a cycle k and an intra-cycle step i, this is written as $\tilde{\mathbf{y}}_{k,i}$.

II. EKF versus LMS: interpreting learning rate

A. Problem Formulation

The physical process in the actual system is often highly complex and difficult to describe precisely, but a mathematical model provides a reasonable and effective simplified representation. We describe the parameterized mathematical model for a general causal system, conceptually illustrated in Fig. 1. The input-output relationship at time step k is given by

$$\mathbf{y}_k = f(\boldsymbol{\theta}; \mathbf{x}_k, \mathbf{x}_{k-1}, \ldots) + \mathbf{e}_k,$$

where $\mathbf{y}_k \in \mathbb{R}^q$ is the system output at time step k, $\hat{\mathbf{y}}_k = f(\boldsymbol{\theta}; \mathbf{x}_k, \mathbf{x}_{k-1}, \ldots) \in \mathbb{R}^q$ is the model output, $\boldsymbol{\theta} \in \mathbb{R}^n$ is the vector of n parameters to be identified, $\mathbf{x}_k \in \mathbb{R}^p$ is the system input vector, $f(\cdot)$, an operator parametrized by $\boldsymbol{\theta}$, maps input signal into the predicted value of output, representing the model structure, and $\mathbf{e}_k \in \mathbb{R}^q$ represents the model error.

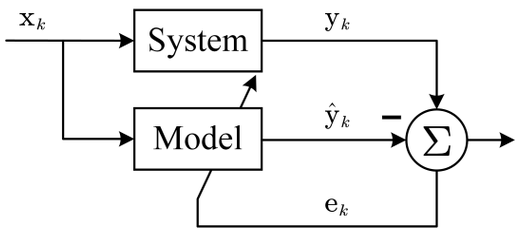

Fig. 1. Conceptual diagram of the parameter estimation process.

The goal of parameter estimation is to find an estimate $\hat{\boldsymbol{\theta}}$ that makes the model output approximate the system output data as closely as possible. A standard formulation for this problem over a batch of k data points is the minimization of the least-squares cost function

$$J(\boldsymbol{\theta}) = \frac{1}{2} \sum_{l=1}^{k} \|\mathbf{y}_l - \hat{\mathbf{y}}_l\|^2. \tag{1}$$

However, in many applications, data arrives sequentially, necessitating an online or recursive algorithm that updates the estimate $\hat{\boldsymbol{\theta}}_{k-1}$ to $\hat{\boldsymbol{\theta}}_k$ using only the new data $(\mathbf{x}_k, \mathbf{y}_k)$. The choice of such an algorithm depends critically on the nature of the operator $f(\cdot)$.

If the model is linear with respect to the parameters the cost function in Eq. (1) has an exact analytical solution that can be computed recursively. This is the celebrated recursive least squares (RLS) algorithm, which recursively computes the optimal solution at each time step [28].

For the more general nonlinear case, the recursive analytical solution is no longer available. A common alternative is to employ gradient-based methods, such as the least mean square (LMS) algorithm [28], [29] or modern variants like Adam [30], [31]. These methods iteratively adjust the parameters based on the instantaneous error, guided by a manually-tuned learning rate μ. While computationally convenient, this learning rate is a heuristic hyperparameter that lacks a clear, interpretable connection to the system's underlying properties.

A fundamentally different paradigm is offered by the EKF. Rather than directly minimizing a cost function, the EKF reframes parameter estimation as a probabilistic inference problem within a state-space model. It "trades" statistical prior knowledge about the system, which is encapsulated in the process and measurement noise covariance matrices $\mathbf{Q}$ and $\mathbf{R}$, for an optimally adaptive and highly interpretable learning gain. This gain is not based on heuristics, but is derived from a rigorous, uncertainty-aware framework. In exchange for this interpretability, the EKF framework entails a significant computational and storage cost of $\mathcal{O}(n^2)$, arising from the manipulation of the full parameter covariance matrix.

B. Extended Kalman Filter for Parameter Estimation

To estimate the parameters $\boldsymbol{\theta}$ in nonlinear systems using a recursive filtering approach, we formulate the problem within a state-space framework. The parameter vector itself is treated as the state, and its dynamics, along with the system's measurement process, are described as follows:

$$\begin{cases} \boldsymbol{\theta}_k = \boldsymbol{\theta}_{k-1} + \mathbf{w}_{k-1} \\ \mathbf{y}_k = f(\boldsymbol{\theta}_k; \mathbf{x}_k, \mathbf{x}_{k-1}, \ldots) + \mathbf{v}_k \end{cases} \tag{2}$$

where $\mathbf{w}_{k-1} \in \mathbb{R}^n$ is the process noise and $\mathbf{v}_k \in \mathbb{R}^q$ is the measurement noise. Both are assumed to be independent, zero-mean Gaussian white noise sequences with covariances $\mathbf{Q}_{k-1}$ and $\mathbf{R}_k$, respectively.

The EKF algorithm provides a recursive solution by iterating through two main stages: prediction and update. Starting with an initial parameter estimate $\hat{\boldsymbol{\theta}}_0$ and its error covariance $\mathbf{P}_0$, the algorithm proceeds as follows for each time step k:

Prediction:

$$\hat{\boldsymbol{\theta}}_{k|k-1} = \hat{\boldsymbol{\theta}}_{k-1}$$
$$\mathbf{P}_{k|k-1} = \mathbf{P}_{k-1} + \mathbf{Q}_{k-1}$$

Update:

$$\mathbf{H}_k = \left. \frac{\partial f(\boldsymbol{\theta}; \mathbf{x}_k, \mathbf{x}_{k-1}, \ldots)}{\partial \boldsymbol{\theta}^T} \right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}_{k|k-1}}$$

$$\mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{H}_k^T(\mathbf{H}_k\mathbf{P}_{k|k-1}\mathbf{H}_k^T + \mathbf{R}_k)^{-1} \tag{3}$$

$$\hat{\boldsymbol{\theta}}_k = \hat{\boldsymbol{\theta}}_{k|k-1} + \mathbf{K}_k \left(\mathbf{y}_k - f(\hat{\boldsymbol{\theta}}_{k|k-1}; \mathbf{x}_k, \mathbf{x}_{k-1}, \ldots) \right) \tag{4}$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_{k|k-1}$$

This prediction-update cycle repeats for each new data point, recursively refining the parameter estimate $\hat{\boldsymbol{\theta}}_k$.

This recursive prediction-update cycle forms the core of the EKF algorithm. However, the true elegance of this framework lies not merely in its procedural steps, but in its underlying principle of optimal, adaptive estimation. The key to this is the Kalman gain, $\mathbf{K}_k$. It is not a static, manually-tuned hyperparameter, but an optimally adaptive learning rate matrix that is dynamically re-computed at each time step.

This gain provides a statistically principled way to fuse new information with existing knowledge. It masterfully balances the confidence in the model's own prediction (quantified by the error covariance $\mathbf{P}_{k|k-1}$) against the reliability of the incoming measurement (quantified by the noise covariance $\mathbf{R}_k$). This ability to intelligently weigh information based on a rigorous model of uncertainty is the source of the EKF's

profound interpretability. To fully appreciate the power of this mechanism, the next section will dive deeper into the statistical meaning of these components and how they enable such an interpretable learning process.

C. Interpretable Adaptive Learning Rate

In signal processing, the choice of learning rate significantly impacts the convergence speed and stability of parameter updates. The traditional LMS algorithm employs a fixed learning rate, which, while simple to implement, may lead to slow convergence or instability when dealing with non-stationary signals or complex environments. In contrast, the EKF dynamically adjusts its learning rate through a formula based on covariance matrices and noise estimation, enabling more accurate tracking of system parameter variations and achieving more robust and effective updates.

In the LMS algorithm, the parameter update

$$\hat{\boldsymbol{\theta}}_k = \hat{\boldsymbol{\theta}}_{k-1} + \mu \mathbf{H}_k^{\mathrm{T}} \left(\mathbf{y}_k - f(\hat{\boldsymbol{\theta}}_{k-1}; \mathbf{x}_k, \ldots) \right)$$

relies on a fixed learning rate μ [28]. This fixed learning rate strategy often struggles to balance convergence speed and stability when faced with changing system dynamics or data statistics.

In contrast, the parameter update for the extended Kalman filter is performed using the rule defined in (4). The Kalman Gain $\mathbf{K}_k$in this equation, calculated via (3), acts as an adaptive learning rate. Unlike a fixed value, it is a matrix dynamically computed based on the statistical information of the system. This adaptability stems from the three core covariance matrices: $\mathbf{P}$ (state error covariance), $\mathbf{Q}$ (process noise covariance), and $\mathbf{R}$ (measurement noise covariance). Understanding the statistical meaning of these three matrices is key to recognizing the adaptive power of the extended Kalman filter.

- **State Error Covariance Matrix ($\mathbf{P}_k$):** The matrix $\mathbf{P}_k$ is the covariance of the parameter estimation error, defined as $\mathrm{E}[(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}})(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}})^{\mathrm{T}}]$. A larger $\mathbf{P}_k$ signifies greater uncertainty, leading the filter to trust new measurements more (via a larger Kalman gain $\mathbf{K}_k$). Conversely, a smaller $\mathbf{P}_k$ indicates higher confidence, causing the filter to rely more on its own prediction. In practice, when the initial state of the parameter is highly uncertain, the covariance matrix $\mathbf{P}_0$ is generally initialized as a diagonal matrix with diagonal elements set to variances significantly larger than the expected range of the state variables. This setup indicates significant initial uncertainty, enabling the filter to prioritize early measurements and accelerate parameter estimation improvement.
- **Process Noise Covariance Matrix ($\mathbf{Q}_k$):** The matrix $\mathbf{Q}_k$, defined as $\mathrm{E}[\boldsymbol{w}_k \boldsymbol{w}_k^{\mathrm{T}}]$, represents the covariance of random, unpredictable changes in the true parameters between time steps, capturing these small perturbations in the actual system. A larger $\mathbf{Q}_k$ indicates greater system disturbances, leading to more significant parameter variations. In particular, $\mathbf{Q}_k = 0$ implies that the system is time-invariant, with parameters assumed to remain constant over time. In practice, real-world physical systems are invariably influenced by environmental disturbances, making $\mathbf{Q}_k$ almost always non-zero.
- **Measurement Noise Covariance ($\mathbf{R}$):** The matrix $\mathbf{R}_k$ is the covariance of the measurement noise, defined as $\mathrm{E}[\boldsymbol{v}_k \boldsymbol{v}_k^{\mathrm{T}}]$. The vector $\mathbf{v}_k$ in the system model represents the combined effects of physical measurement error and inaccuracies in the model function $f(\cdot)$. A large value for $\mathbf{R}_k$ indicates that measurements are noisy or unreliable, compelling the filter to down-weight them with a smaller Kalman gain. This value can often be determined from sensor specifications or estimated from offline data analysis.

In summary, the EKF provides a profoundly interpretable learning mechanism. Through the interplay of $\mathbf{P}_k$, $\mathbf{Q}_k$, and $\mathbf{R}_k$, the Kalman gain $\mathbf{K}_k$ achieves an optimal, real-time balance between trusting the model's existing knowledge and embracing new, incoming data. This intelligent, data-driven adjustment enables superior performance compared to fixed-learning-rate methods, especially in dynamic systems.

III. Improving FDEKF Into RR-FDEKF

The preceding section established the EKF as a theoretically powerful framework for parameter estimation, celebrated for its profound interpretability and statistically optimal and adaptive learning. In an ideal setting, it would be the definitive tool for the problems addressed in this paper.

However, a critical obstacle severely limits its practical application: its prohibitive computational cost. The core of the EKF algorithm requires the storage and manipulation of the full $n \times n$ error covariance matrix $\mathbf{P}_k$, leading to a computational complexity that scales as $\mathcal{O}(n^2)$ with the number of parameters, n. This quadratic scaling renders the EKF impractical for the high-dimensional estimation challenges prevalent in modern complex systems.

A. FDEKF for Parameter Estimation

To alleviate the computational burden, a natural simplification is to ignore some interdependence between the system parameters; this approach is referred to as decoupled extended Kalman filter (DEKF) [25], [28]. While various DEKF formulations exist, we adopt a fully decoupled, local implementation that aligns with the independent EKF (IEKF) philosophy [23], [24]. This specific formulation, known as the FDEKF, prioritizes computational efficiency and is realized by treating each parameter as an independent state.

For the parameter vector $\boldsymbol{\theta}_k = [\theta_k^1, \theta_k^2, \ldots, \theta_k^n]^{\mathrm{T}}$, we formulate n independent scalar state-space models, each corresponding to a single parameter component θ_k^i. The i-th model corresponding to the i-th parameter θ_k^i is given by

$$\begin{cases} \theta_k^i = \theta_{k-1}^i + w_{k-1}^i, \\ \mathbf{y}_k = f_i(\theta_k^i; \hat{\boldsymbol{\theta}}_{k-1}^{-i}, \mathbf{x}_k, \mathbf{x}_{k-1}, \ldots) + \mathbf{v}_k. \end{cases} \tag{5}$$

Here

- $w_{k-1}^i \sim \mathcal{N}(0, Q_{k-1}^i)$ is the scalar process noise, where Q_{k-1}^i is the i-th diagonal element of the overall process noise covariance matrix $\mathbf{Q}_{k-1}$.

- $\hat{\boldsymbol{\theta}}_{k-1}^{-i} = [\hat{\theta}_{k-1}^1, \ldots, \hat{\theta}_{k-1}^{i-1}, \hat{\theta}_{k-1}^{i+1}, \ldots, \hat{\theta}_{k-1}^n]^{\mathrm{T}}$ represents the estimated parameter vector except $\hat{\theta}^i$.
- $f_i(\theta_k^i; \hat{\boldsymbol{\theta}}_{k-1}^{-i}, \mathbf{x}_k, \ldots)$ represents the measurement function from the perspective of the i-th model.
- $\mathbf{v}_k \sim \mathcal{N}(0, \mathbf{R}_k)$ is the measurement noise, independent of w_k^i.

Applying the EKF framework to each model (5) independently yields each parameter estimation.

Prediction: For each parameter $i = 1, \ldots, n$:

$$\begin{aligned}\hat{\theta}_{k|k-1}^i &= \hat{\theta}_{k-1}^i \\ P_{k|k-1}^i &= P_{k-1}^i + Q_{k-1}^i.\end{aligned}$$

Update: For each parameter $i = 1, \ldots, n$:

$$\begin{aligned}\tilde{\mathbf{y}}_k^i &= \mathbf{y}_k - f_i(\hat{\theta}_{k|k-1}^i; \hat{\boldsymbol{\theta}}_{k-1}^{-i}, \mathbf{x}_k, \ldots) \\ \mathbf{S}_k^i &= \mathbf{h}_k^i P_{k|k-1}^i (\mathbf{h}_k^i)^{\mathrm{T}} + \mathbf{R}_k \\ \mathbf{K}_k^i &= P_{k|k-1}^i (\mathbf{h}_k^i)^{\mathrm{T}} (\mathbf{S}_k^i)^{-1} \\ \hat{\theta}_k^i &= \hat{\theta}_{k|k-1}^i + \mathbf{K}_k^i \tilde{\mathbf{y}}_k^i \\ P_k^i &= (1 - \mathbf{K}_k^i \mathbf{h}_k^i) P_{k|k-1}^i,\end{aligned}$$

where the Jacobian $\mathbf{h}_k^i$ is computed as

$$\mathbf{h}_k^i = \left. \frac{\partial f_i(\theta^i; \hat{\boldsymbol{\theta}}_{k-1}^{-i}, \mathbf{x}_k, \mathbf{x}_{k-1}, \ldots)}{\partial \theta^i} \right|_{\theta^i = \hat{\theta}_{k|k-1}^i}.$$

The overall FDEKF algorithm is summarized in Algorithm 1.

Algorithm 1 FDEKF Algorithm

Initialization: For $i = 1, \ldots, n$, initialize $\hat{\theta}_0^i$ and P_0^i.
for each time step $k = 1, 2, \ldots$ **do**
  for each parameter $i = 1, \ldots, n$ **do**
    Predict:
      $\hat{\theta}_{k|k-1}^i = \hat{\theta}_{k-1}^i$
      $P_{k|k-1}^i = P_{k-1}^i + Q_{k-1}^i$
    Update:
      $\mathbf{h}_k^i = \left. \frac{\partial f_i(\theta^i; \hat{\boldsymbol{\theta}}_{k-1}^{-i}, \mathbf{x}_k, \ldots)}{\partial \theta^i} \right|_{\theta^i = \hat{\theta}_{k|k-1}^i}$
      $\tilde{\mathbf{y}}_k^i = \mathbf{y}_k - f_i(\hat{\theta}_{k|k-1}^i; \hat{\boldsymbol{\theta}}_{k-1}^{-i}, \mathbf{x}_k, \ldots)$
      $\mathbf{S}_k^i = \mathbf{h}_k^i P_{k|k-1}^i (\mathbf{h}_k^i)^{\mathrm{T}} + \mathbf{R}_k$
      $\mathbf{K}_k^i = P_{k|k-1}^i (\mathbf{h}_k^i)^{\mathrm{T}} (\mathbf{S}_k^i)^{-1}$
      $\hat{\theta}_k^i = \hat{\theta}_{k|k-1}^i + \mathbf{K}_k^i \tilde{\mathbf{y}}_k^i$
      $P_k^i = (1 - \mathbf{K}_k^i \mathbf{h}_k^i) P_{k|k-1}^i$
  end for
end for

For a system with n parameters and an q-dimensional measurement vector, the computational complexity of FDEKF is primarily determined by the n independent update calculations. Each such update requires operations on $q \times q$ matrices, such as forming and inverting the innovation covariance $\mathbf{S}_k^i$, contributing $\mathcal{O}(q^3)$ per parameter. Consequently, the dominant complexity for the FDEKF update stage is $\mathcal{O}(nq^3)$. This contrasts sharply with the EKF, which typically incurs a complexity of $\mathcal{O}(n^2)$ due to operations involving the full $n \times n$ covariance matrix. Therefore, when the measurement dimension q is small relative to the parameter dimension n (specifically, when $q^3 \ll n$), the FDEKF provides substantial computational savings. This renders the FDEKF particularly advantageous for high-dimensional parameter estimation problems.

B. RR-FDEKF for Parameter Estimation

The FDEKF processes a single measurement $\mathbf{y}_k$ to update all n parameter estimates at one time step. Sometimes, these n number of estimates can not be arranged completely in parallel. This is typical when the model architecture involves multilayered blocks which are cascaded. Therefore a significant challenge emerges when we need to deal with a continuous stream of measurements arriving at a high frequency and the real-time online running is essential for engineering requirement. The processing time required for the FDEKF to update all n parameters exceeds the extremely short interval between consecutive measurements in a rapid stream.

To reduce computational complexity, we propose RR-FDEKF, which only updates a single parameter θ^i using per measurement. To formalize this, we structure the algorithm around logical processing cycles, indexed by $k = 1, 2, \ldots$. Within each cycle, parameters are updated sequentially from $i = 1$ to n. We establish a direct mapping from the original continuous data stream, indexed by time $t = 1, 2, \ldots$, to our cyclical framework (k, i). The measurement $\mathbf{y}_{k,i}$ and input history starting with $\mathbf{x}_{k,i}$ used in the i-th step of the k-th cycle correspond to the data from the physical time step $t = (k-1)n + i$.

This process is illustrated in Fig. 2.

The dynamics of each parameter $\theta^i$and its relationship to the corresponding measurement $\mathbf{y}_{k,i}$ within this cyclical framework are described by the following state-space model:

$$\begin{cases}\theta_k^i = \theta_{k-1}^i + w_{k-1}^i \\ \mathbf{y}_{k,i} = f_i(\theta_k^i; \boldsymbol{\xi}_k^i, \mathbf{x}_{k,i}, \mathbf{x}_{k,i-1}, \ldots) + \mathbf{v}_{k,i}\end{cases}$$

where $w_{k-1}^i \sim \mathcal{N}(0, Q_{k-1}^i)$ is the process noise accumulating between cycles, and $\mathbf{v}_{k,i} \sim \mathcal{N}(0, \mathbf{R}_{k,i})$ is the measurement noise associated with the specific measurement $\mathbf{y}_{k,i}$. The vector $\boldsymbol{\xi}_k^i$ denotes the "mixed-cycle" information vector used for the i-th update, composed of the most recent estimates of all other parameters, and is defined as $\boldsymbol{\xi}_k^i = [\hat{\theta}_k^1, \ldots, \hat{\theta}_k^{i-1}, \hat{\theta}_{k-1}^{i+1}, \ldots, \hat{\theta}_{k-1}^n]^{\mathrm{T}}$.

The complete algorithm based on this cyclical structure is described in Algorithm 2.

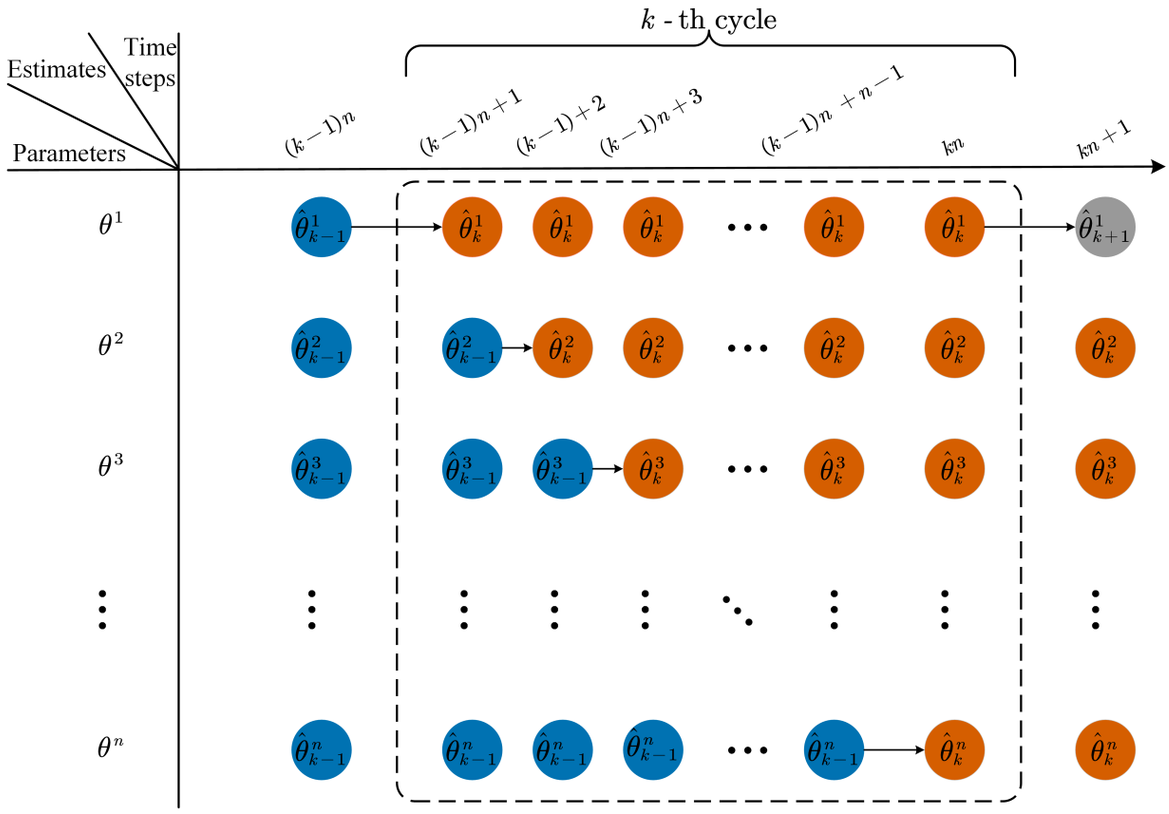

Fig. 2. Conceptual illustration of RR-FDEKF within a processing k-th cycle.

Algorithm 2 RR-FDEKF Algorithm

Initialization: For $i = 1, \ldots, n$, initialize $\hat{\theta}_0^i$ and P_0^i.
for each logical processing cycle $k = 1, 2, \ldots$ **do**
  for each parameter index $i = 1, \ldots, n$ **do**
    Let $(\mathbf{y}_{k,i}, \mathbf{x}_{k,i}, \ldots)$ be the data from the physical time step $t = (k-1)n + i$.
    Predict:
      $\hat{\theta}_{k|k-1}^i = \hat{\theta}_{k-1}^i$
      $P_{k|k-1}^i = P_{k-1}^i + Q_{k-1}^i$
    Update:
      Construct the mixed-cycle vector
      $\boldsymbol{\xi}_k^i = [\hat{\theta}_k^1, \ldots, \hat{\theta}_k^{i-1}, \hat{\theta}_{k-1}^{i+1}, \ldots, \hat{\theta}_{k-1}^n]^{\mathrm{T}}$
      $\mathbf{h}_{k,i} = \left.\frac{\partial f_i(\theta^i; \boldsymbol{\xi}_k^i, \mathbf{x}_{k,i}, \ldots)}{\partial \theta^i}\right|_{\theta^i = \hat{\theta}_{k|k-1}^i}$
      $\tilde{\mathbf{y}}_{k,i} = \mathbf{y}_{k,i} - f_i(\hat{\theta}_{k|k-1}^i; \boldsymbol{\xi}_k^i, \mathbf{x}_{k,i}, \ldots)$
      $\mathbf{S}_{k,i} = \mathbf{h}_{k,i} P_{k|k-1}^i (\mathbf{h}_{k,i})^{\mathrm{T}} + \mathbf{R}_{k,i}$
      $\mathbf{K}_{k,i} = P_{k|k-1}^i (\mathbf{h}_{k,i})^{\mathrm{T}} (\mathbf{S}_{k,i})^{-1}$
      $\hat{\theta}_k^i = \hat{\theta}_{k|k-1}^i + \mathbf{K}_{k,i} \tilde{\mathbf{y}}_{k,i}$
      $P_k^i = (1 - \mathbf{K}_{k,i} \mathbf{h}_{k,i}) P_{k|k-1}^i$
  end for
end for

The elegance of the RR-FDEKF lies in its sequential update strategy, which maximizes data utility at minimal cost. At any given time step, the single measurement vector $\mathbf{y}_{k,i}$ is used to refine only one parameter, θ^i. To make this single update as effective as possible, the algorithm constructs the mixed-cycle vector $\boldsymbol{\xi}_k^i$, which leverages the most current information available. By incorporating the freshly updated estimates $(\hat{\theta}_k^1, \ldots, \hat{\theta}_k^{i-1})$ from within the same cycle, information from earlier updates immediately cascades to benefit later ones. This intra-cycle information propagation ensures that the algorithm converges efficiently, making each measurement's contribution count to the fullest extent.

IV. Experimental Validation

In this section, we evaluate the performance of the FDEKF and RR-FDEKF for parameter estimation in complex nonlinear systems. We test these algorithms on a practical scenario involving PIM cancellation, using real-world measurement data. We compare FDEKF and RR-FDEKF against the Adam optimizer, a widely used benchmark algorithm, focusing on estimation accuracy and computational efficiency.

A. Experimental Setup

PIM cancellation is a critical challenge in modern wireless communication systems, requiring accurate parameter estimation for effective cancellation. The conceptual block diagram of the simultaneous transmit-receive communication system is shown in Fig. 3. The real-world measurement data for this study are collected from the laboratory setup. To model the nonlinear PIM generation, we employed the Symmetrized

Basis Function Approximation Network (BFAN) as presented in [32]. This model maps the input signals to a single scalar output (i.e., $q = 1$) representing the estimated PIM interference, using a set of $n = 232$ parameters ($\boldsymbol{\theta}$) that need to be identified. The key objective of our work is to accurately estimate the parameter of the model.

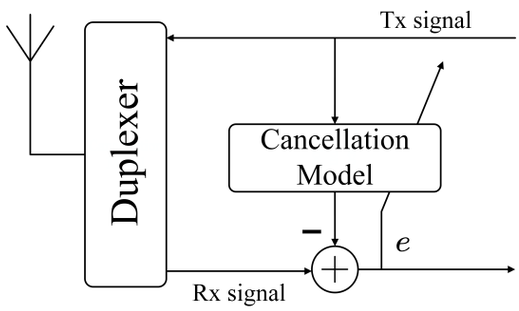

Fig. 3. Block diagram of the digital PIM cancellation system setup.

B. Evaluation Setup

To ensure the reproducibility of these experiments, we detail the experimental setup for evaluating the parameter estimation algorithms. All methods are implemented in Python using PyTorch to ensure a consistent environment. The experiments are conducted on a workstation equipped with an Intel(R) Xeon(R) CPU @ 2.20GHz. The initial parameters for all algorithms are initialized from a $\mathcal{N}(0, 0.01)$ distribution. We process the dataset of 64,800 samples one by one, where one full pass through the dataset defines a single training epoch. To account for the effects of random initialization and ensure statistical validity, each experiment was repeated 20 times with different random seeds.

The hyperparameter configurations for each algorithm are carefully selected based on standard practices and preliminary experiments to ensure a fair and robust comparison. We adjust all settings to ensure more accurate model estimations, minimizing the error (measured by MSE).

- **Adam (Baseline):** The Adam optimizer [30], a widely-used gradient-based baseline, is configured with a learning rate $\alpha = 0.001$, exponential decay rates $\beta_1 = 0.9$ and $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$, based on engineering experience.
- **FDEKF:** Key tuning parameters are set based on physical interpretations and empirical adjustments. The initial parameter error variances P_0^i are set to 0.01, onsistent with the variance of the parameter initialization distribution. The process noise variance Q_k^i follows a dynamic schedule: initially zero for 10 epochs to accelerate convergence, then increases to 10^{-6} to prevent Kalman gain vanishing, and subsequently decays by a factor of 0.7 every 50 epochs to a minimum of 10^{-12}. The measurement noise variance R_k is fixed at 10^{-8}, calculated from the inherent noise in the receiving channel.
- **RR-FDEKF:** For a direct and fair comparison, the RR-FDEKF is configured with the same parameters as the FDEKF. Specifically, it shared the same initial error variances P_0^i, measurement noise variance R_k, and the dynamic schedule for the process noise variance Q_k^i.

In real-world scenarios, the optimal model parameters $\boldsymbol{\theta}_{\text{true}}$ for interference cancellation are unknown due to complex signal environments. To evaluate algorithm effectiveness, two practical metrics are used. The primary metric, residual PIM power (in dB), measures the power of uncancelled intermodulation distortion after cancellation. The second metric, computational efficiency, is assessed as the average execution time required to process each data sample and update the model parameters.

C. Results and Analysis

We compare FDEKF, RR-FDEKF, and the baseline Adam optimizer for PIM cancellation, showing that both decoupled Kalman filter methods significantly outperform Adam. The Power Spectral Density (PSD) plot in Fig. 4 shows that FDEKF and RR-FDEKF achieve much lower residual error than Adam. Table I provides the precise results: after 400 epochs, FDEKF and RR-FDEKF attained residual PIM of 2.56 dB and 2.61 dB, respectively, compared to Adam's 4.77 dB.

The residual error plot in Fig. 5 underscores the superior performance of the decoupled Kalman filter methods. Their adaptive learning rate mechanism drives a remarkably rapid decline in residual error during early iterations, while also achieving lower final residual error compared to Adam. Additionally, the Kalman filter-based methods exhibit markedly improved stability. As presented in Table I, the final standard deviations for FDEKF at 0.19 dB and RR-FDEKF at 0.25 dB are over three times lower than Adam's 0.83 dB, demonstrating more consistent and dependable error cancellation performance.

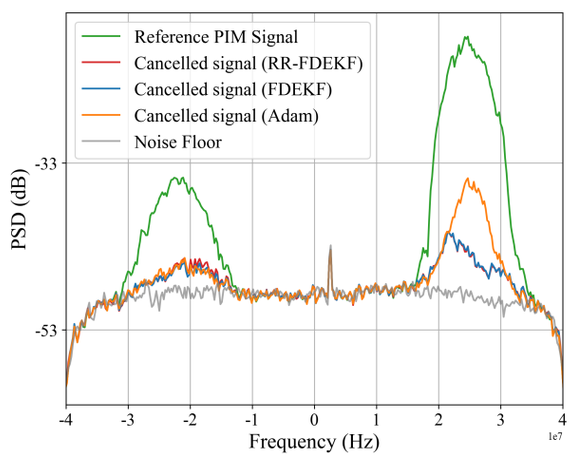

Fig. 4. Power spectral density of the reference PIM signal and the residual signals after cancellation by FDEKF, RR-FDEKF, and Adam.

Before evaluating practical computational efficiency, we analyze the theoretical complexity for our specific experimental case of a single-output system ($q = 1$). The FDEKF adopts a parallel decoupling strategy, enabling independent updates of n parameters. This eliminates the expensive matrix operations

TABLE I
PIM CANCELLATION PERFORMANCE (RESIDUAL PIM POWER [DB], MEAN ± STD) AFTER DIFFERENT EPOCHS. ORIGINAL PIM LEVEL: ≈19.13 DB.

Algorithm	Residual PIM Power (dB) at Epoch (Mean ± Std)					
	5	25	50	100	200	400
Adam	7.78 ± 0.33	7.52 ± 0.53	6.80 ± 0.52	5.82 ± 0.61	4.91 ± 0.86	4.77 ± 0.83
FDEKF	7.16 ± 0.32	6.51 ± 0.18	5.74 ± 0.43	3.68 ± 0.29	2.71 ± 0.20	**2.56 ± 0.19**
RR-FDEKF	7.20 ± 0.31	6.60 ± 0.20	6.20 ± 0.51	4.47 ± 0.65	3.06 ± 0.41	2.61 ± 0.25

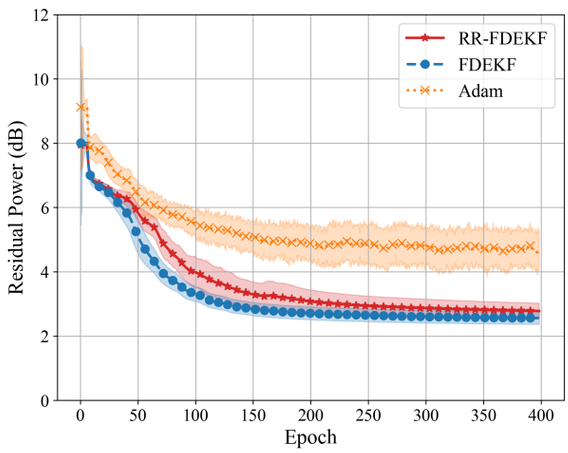

Fig. 5. Residual power (dB) versus training epoch for FDEKF, RR-FDEKF, and Adam on the PIM cancellation.

of the standard EKF, reducing computational complexity to $\mathcal{O}(n)$. Likewise, the RR-FDEKF employs a Round-Robin decoupling method, sequentially updating parameters within a single data cycle. This sequential approach also achieves $\mathcal{O}(n)$ complexity. By immediately incorporating information from one parameter's update into the next, the RR-FDEKF is well-suited for low-latency and online applications.

TABLE II
AVERAGE EXECUTION TIME PER EPOCH (S)

Algorithm	Average Execution Time (s)
Adam	70.9513 ± 4.1053
FDEKF	70.7753 ± 3.0392
RR-FDEKF	**60.7785** ± 7.4055

The execution times in Table II confirm the computational efficiency of the proposed designs. The FDEKF achieves a runtime comparable to the highly optimized Adam baseline, while the RR-FDEKF demonstrates a notable advantage, proving to be the most efficient method. Notably, RR-FDEKF proves to be the most efficient, requiring only 60.78 s per epoch. This reflects a 1.16x speedup compared to FDEKF, which takes 70.78 s per epoch, highlighting RR-FDEKF's enhanced efficiency due to its sequential update structure. This empirical performance advantage, combined with RR-FDEKF's architectural suitability for pipelined processing, underscores its implementation advantages for computationally intensive real-time applications.

V. CONCLUSION

Both the FDEKF and the proposed RR-FDEKF have the virtue of being able to interpret the Kalman gain as a mechanism-clear adaptive learning rate. They offer an alternative with clear theoretical foundation to Adam, a heuristic but highly recognized optimizer in deep learning research field. In digital PIM cancellation, an advanced technique in wireless communication, our results confirm that both algorithms achieve better cancellation performance and faster convergence than Adam. Specifically, the RR-FDEKF's sequential, single-parameter update scheme is well-suited for high-data-rate scenarios with numerous parameters, ensuring high computational efficiency. These findings provide strong evidence for the capability of this class of algorithms, and particularly highlight the considerable potential of the proposed RR-FDEKF for the online identification of complex dynamic systems.

REFERENCES

[1] L. Ljung, "Perspectives on system identification," *Annual Reviews in Control*, vol. 34, no. 1, pp. 1–12, 2010.
[2] T. Söderström, P. Stoica, and B. Friedlander, "An indirect prediction error method for system identification," *Automatica*, vol. 27, no. 1, pp. 183–188, 1991.
[3] M. Z. Waheed, D. Korpi, L. Anttila, A. Kiayani, M. Kosunen, K. Stadius, P. P. Campo, M. Turunen, M. Allén, J. Ryynänen *et al.*, "Passive intermodulation in simultaneous transmit–receive systems: Modeling and digital cancellation methods," *IEEE Transactions on Microwave Theory and Techniques*, vol. 68, no. 9, pp. 3633–3652, 2020.
[4] J. R. Wilkerson, I. M. Kilgore, K. G. Gard, and M. B. Steer, "Passive intermodulation distortion in antennas," *IEEE Transactions on Antennas and Propagation*, vol. 63, no. 2, pp. 474–482, 2014.
[5] F. Mkadem and S. Boumaiza, "Physically inspired neural network model for rf power amplifier behavioral modeling and digital predistortion," *IEEE Transactions on Microwave Theory and Techniques*, vol. 59, no. 4, pp. 913–923, 2011.
[6] D. R. Morgan, Z. Ma, J. Kim, M. G. Zierdt, and J. Pastalan, "A generalized memory polynomial model for digital predistortion of rf power amplifiers," *IEEE Transactions on signal processing*, vol. 54, no. 10, pp. 3852–3860, 2006.
[7] V. G. Asutkar, B. M. Patre, and T. Basu, "Kalman filter approach for identification of linear fast time-varying processes," in *2009 International Conference on Control, Automation, Communication and Energy Conservation*. IEEE, 2009, pp. 1–5.
[8] S. Deepak, R. Aiswarya, C. Aparna, and J. J. Nair, "Optimization of gaussian membership functions using unscented kalman filter," in *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. IEEE, 2018, pp. 957–961.
[9] M. Q. Phan, F. Vicario, R. W. Longman, and R. Betti, "State-space model and kalman filter gain identification by a kalman filter of a kalman filter," *Journal of Dynamic Systems, Measurement, and Control*, vol. 140, no. 3, p. 030902, 2018.
[10] X. Sun, L. Jin, and M. Xiong, "Extended kalman filter for estimation of parameters in nonlinear state-space models of biochemical networks," *PloS one*, vol. 3, no. 11, p. e3758, 2008.
[11] P. Maybeck, J. Negro, S. Cusumano, and M. DePonte, "A new tracker for air-to-air missile targets," *IEEE Transactions on Automatic Control*, vol. 24, no. 6, pp. 900–905, 1979.

[12] D. Simon, *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons, 2006.
[13] Y. Wang, Y. Li, and Z. Zhao, "State parameter estimation of intelligent vehicles based on an adaptive unscented kalman filter," *Electronics*, vol. 12, no. 6, p. 1500, 2023.
[14] T. Yoshimura, K. Konishi, and T. Soeda, "A modified extended kalman filter for linear discrete-time systems with unknown parameters," *Automatica*, vol. 17, no. 4, pp. 657–660, 1981.
[15] A. P. Sage, J. L. Melsa, and W. Steinway, "Estimation theory with applications to communication and control," *IEEE Transactions on Systems, Man, and Cybernetics*, no. 4, pp. 405–405, 1971.
[16] S. Sinha and T. Nagaraja, "Extended kalman filter algorithm for continuous system parameter identification," *Computers & electrical engineering*, vol. 16, no. 1, pp. 51–64, 1990.
[17] M. Mansouri, H. Tolouei, and M. A. Shoorehdeli, "Identification of hammerstein-wiener armax systems using extended kalman filter," in *2011 Chinese Control and Decision Conference (CCDC)*. IEEE, 2011, pp. 1110–1114.
[18] D. Li and Y. Wang, "Parameter identification of a differentiable bouc-wen model using constrained extended kalman filter," *Structural Health Monitoring*, vol. 20, no. 1, pp. 360–378, 2021.
[19] M. Gautier and P. Poignet, "Extended kalman filtering and weighted least squares dynamic identification of robot," *Control Engineering Practice*, vol. 9, no. 12, pp. 1361–1372, 2001.
[20] X. Wang, J. Li, S. Chen, G. Zhang, B. Jiang, X. Wei, and H. Dai, "Online detection of lithium plating onset for lithium-ion batteries based on impedance changing trend identification during charging processes," *IEEE Transactions on Transportation Electrification*, vol. 9, no. 2, pp. 3487–3497, 2022.
[21] A. Gaytan, O. Begovich-Mendoza, and N. Arana-Daniel, "Training of convolutional neural networks for image classification with fully decoupled extended kalman filter," *Algorithms*, vol. 17, no. 6, p. 243, 2024.
[22] H. Jaeger, *Tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the echo state network approach*. Citeseer, 2002, vol. 5, no. 1.
[23] S. Kollias and D. Anastassiou, "An adaptive least squares algorithm for the efficient training of artificial neural networks," *IEEE Transactions on Circuits and Systems*, vol. 36, no. 8, pp. 1092–1101, 1989.
[24] S. Shah and F. Palmieri, "Meka-a fast, local algorithm for training feed-forward neural networks," in *1990 IJCNN International Joint Conference on Neural Networks*. IEEE, 1990, pp. 41–46.
[25] G. V. Puskorius and L. A. Feldkamp, "Decoupled extended kalman filter training of feedforward layered networks," in *IJCNN-91-Seattle International Joint Conference on Neural Networks*, vol. 1. IEEE, 1991, pp. 771–777.
[26] M. Ciołek, M. Niedźwiecki, and A. Gańcza, "Decoupled kalman filter based identification of time-varying fir systems," *IEEE Access*, vol. 9, pp. 74 622–74 631, 2021.
[27] Z. Youmin, P. Quan, Z. Hongcai, and D. Guanzhong, "A parallel decoupled kalman filtering algorithm and systolic architecture," in *Proceedings of 32nd IEEE Conference on Decision and Control*. IEEE, 1993, pp. 3590–3595.
[28] S. S. Haykin, *Adaptive filter theory*. Pearson Education India, 2002.
[29] B. Widrow, J. R. Glover, J. M. McCool, J. Kaunitz, C. S. Williams, R. H. Hearn, J. R. Zeidler, J. E. Dong, and R. C. Goodlin, "Adaptive noise cancelling: Principles and applications," *Proceedings of the IEEE*, vol. 63, no. 12, pp. 1692–1716, 1975.
[30] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
[31] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*. MIT press Cambridge, 2016, vol. 1, no. 2.
[32] C. Liu and Z. Yan, "Symmetrized Basis Function Approximation Network for Passive Intermodulation Cancellation," *IEEE Transactions on Communications*, 2025, doi:10.1109/TCOMM.2025.3560338.