Cholesky-KalmanNet: Model-Based Deep Learning with Positive Definite Error Covariance Structure

Minhyeok Ko, Abdollah Shafieezadeh, *Member, IEEE*

***Abstract*—State estimation from noisy observations is crucial across various fields. Traditional methods such as Kalman, Extended Kalman, and Unscented Kalman Filter often struggle with nonlinearities, model inaccuracies, and high observation noise. This paper introduces Cholesky-KalmanNet (CKN), a model-based deep learning approach that considerably enhances state estimation by providing and enforcing transiently precise error covariance estimation. Specifically, the CKN embeds mathematical constraints associated with the positive definiteness of error covariance in a recurrent DNN architecture through the Cholesky decomposition. This architecture enhances statistical reliability and mitigates numerical instabilities. Furthermore, introducing a novel loss function that minimizes discrepancies between the estimated and empirical error covariance ensures a comprehensive minimization of estimation errors, accounting for interdependencies among state variables. Extensive evaluations on both synthetic and real-world datasets affirm CKN's superior performance vis-a-vis state estimation accuracy, robustness against system inaccuracies and observation noise, as well as stability across varying training data partitions, an essential feature for practical scenarios with suboptimal data conditions.**

***Index Terms*—Deep Learning, Kalman Filter, Recurrent Neural Network, State Estimation, Uncertainty Quantification**

I. Introduction

ACCURATE state estimation from noisy observations is essential in many applications [1], [2], [3], [4], [5]. Sensor noise and unmeasurable states often obscure the true state, requiring estimation techniques to reconstruct internal states. The Kalman Filter (KF) [6] provides optimal estimation for linear systems with Gaussian noise. However, the Extended Kalman Filter (EKF) [7] struggles with high nonlinearity and Jacobian computations [8], while the Unscented Kalman Filter (UKF) [9] may suffer from numerical instability [10]. These filters also rely on accurate system models, which are often unavailable in complex, real-world scenarios [11], [12].

KalmanNet (KN) [13], combining KF's structure with the adaptability of Deep Neural Networks (DNNs), addresses nonlinear dynamics and model inaccuracies by learning the Kalman gain directly from data, removing the need to tune noise covariances. Split-KalmanNet (SKN) [14] further improves KN by using two separate DNNs for state and innovation covariance estimation, effectively handling mismatches in process and measurement models. The KN framework has been extended in various ways to tackle different state estimation challenges [15], [16], [17], [18], [19], [20].

Robust Uncertainty Quantification (UQ) is crucial in many applications [21], [22], [23], [24], [25]. In the KN framework, UQ can be derived analytically for linear state-space models with full-column ranked observation matrices [26], while SKN estimates error covariance through DNNs [27]. However, both approaches face limitations, including the requirement for full-rank observation matrices and risks of producing non-positive definite covariances, which may limit their applicability across a wide range of real-world scenarios.

In this paper, we introduce Cholesky-KalmanNet (CKN), which significantly advances the KN framework to enhance the accuracy, robustness, and stability of state estimation. Our key contributions are summarized as follows: (i) *Enabling Precise Error Covariance Estimation*: By enforcing mathematical constraints through Cholesky decomposition within a recurrent DNN architecture, we ensure positive definite error covariance matrices, enabling UQ for estimated states in any problem. Additionally, the proposed enforcement of the mathematical constraints reduces the parameter space estimated by DNNs to nearly half compared to SKN, reducing training time, lowering training data requirements, and mitigating overparameterization; (ii) *Enhanced State Estimation*: The CKN architecture integrates estimated error covariance through a proposed hybrid loss function, improving the accuracy of state estimates. Furthermore, Confidence Intervals (CIs) generated by CKN can capture the transient fluctuation in the error estimates; and (iii) *Rigorous Benchmark Validation*: We demonstrate superior accuracy, robustness, and stability of CKN across diverse scenarios including varying noise levels, different degrees of system model inaccuracies, and model training variance with respect to training sets.

II. Problem Formulation and Preliminaries

Dynamic systems are analyzed using discrete-time state-space models, defined at each time step t, given as:

$$\mathbf{x}_t = \mathbf{f}\left(\mathbf{x}_{t-1}\right) + \mathbf{w}_t, \qquad \mathbf{x}_t \in \mathbb{R}^m \tag{1a}$$

$$\mathbf{y}_t = \mathbf{h}\left(\mathbf{x}_t\right) + \mathbf{v}_t, \qquad \mathbf{y}_t \in \mathbb{R}^n \tag{1b}$$

where $\mathbf{x}_t$ is the state vector, evolving through a nonlinear state function $\mathbf{f}(\cdot)$, and $\mathbf{y}_t$ is the observation vector obtained via a nonlinear observation function $\mathbf{h}(\cdot)$. The random processes $\mathbf{w}_t$

This work was supported in part by the U.S. National Science Foundation under grant CMMI-2000156, the Korea Institute of Energy Technology Evaluation and Planning (KETEP) and the Ministry of Trade, Industry & Energy (MOTIE) of the Republic of Korea (No. 20224B10200040), and the Lichtenstein endowment at The Ohio State University. Any opinions, findings, and conclusions are those of the authors and do not necessarily reflect those of the sponsoring organization. (*Corresponding author: Abdollah Shafieezadeh.*)

Minhyeok Ko and Abdollah Shafieezadeh are with the Risk Assessment and Management of Structural and Infrastructural Systems (RAMSIS) Laboratory, Department of Civil, Environmental, and Geodetic Engineering, The Ohio State University, Columbus, OH 43210 USA (e-mail: ko.444@osu.edu; shafieezadeh.1@osu.edu).

and $\mathbf{v}_t$ are white Gaussian with zero mean and covariances $\mathbb{E}\left[\mathbf{w}_t\mathbf{w}_t^T\right]=\mathbf{Q}_t$ and $\mathbb{E}\left[\mathbf{v}_t\mathbf{v}_t^T\right]=\mathbf{R}_t$.

A. Model-based Extended Kalman Filter

The EKF is widely used for state estimation in dynamic systems with nonlinear state and observation models. The main filter equations are as follows:

- Prediction Step:

$$\hat{\mathbf{x}}_{t|t-1}=\mathbf{f}\left(\hat{\mathbf{x}}_{t-1|t-1}\right) \tag{2a}$$

$$\hat{\mathbf{\Sigma}}_{t|t-1}=\mathbf{F}_{t-1}\hat{\mathbf{\Sigma}}_{t-1|t-1}\mathbf{F}_{t-1}^T+\mathbf{Q}_t \tag{2b}$$

- Update Step:

$$\hat{\mathbf{x}}_{t|t}=\hat{\mathbf{x}}_{t|t-1}+\mathbf{K}_t\left(\mathbf{y}_t-\hat{\mathbf{y}}_{t|t-1}\right) \tag{3a}$$

$$\hat{\mathbf{\Sigma}}_{t|t}=\left(\mathbf{I}-\mathbf{K}_t\mathbf{H}_t\right)\hat{\mathbf{\Sigma}}_{t|t-1} \tag{3b}$$

$$\mathbf{K}_t=\hat{\mathbf{\Sigma}}_{t|t-1}\mathbf{H}_t^T\hat{\mathbf{S}}_{t|t-1}^{-1} \tag{3c}$$

where $\mathbf{F}_t=\frac{\partial\mathbf{f}}{\partial\mathbf{x}}|_{\mathbf{x}=\hat{\mathbf{x}}_{t|t}}$, $\mathbf{H}_t=\frac{\partial\mathbf{h}}{\partial\mathbf{x}}|_{\mathbf{x}=\hat{\mathbf{x}}_{t|t-1}}$, $\hat{\mathbf{y}}_{t|t-1}=\mathbf{h}\left(\hat{\mathbf{x}}_{t|t-1}\right)$, and $\hat{\mathbf{S}}_{t|t-1}=\mathbf{H}_t\hat{\mathbf{\Sigma}}_{t|t-1}\mathbf{H}_t^T+\mathbf{R}_t$.

EKF's dependence on the physical model, which may be inaccurate, has led to integrating Deep Learning (DL) techniques within the KF framework for improved performance.

B. Deep Learning-aided Kalman Filters

In KN, the Kalman gain $\mathbf{K}_t(\mathbf{\Theta})$, where $\mathbf{\Theta}$ represents the set of trainable parameters, is estimated using Recurrent Neural Networks (RNNs), particularly the Gated Recurrent Unit (GRU). The update equation for the KN is defined as $\hat{\mathbf{x}}_{t|t}=\hat{\mathbf{x}}_{t|t-1}+\mathbf{K}_t(\mathbf{\Theta})(\mathbf{y}_t-\hat{\mathbf{y}}_{t|t-1})$, where $\mathbf{K}_t(\mathbf{\Theta})$ is adjusted continuously through training. For a comprehensive overview of KN, we refer to [13].

Unlike the conventional KN where a single network estimates the Kalman gain, SKN employs two separate networks. The first network, $\mathcal{G}_t^1(\mathbf{\Theta}_1)$, is designed to learn the prior covariance matrix of the state $\mathbf{\Sigma}_{t|t-1}$ implicitly. The second network, $\mathcal{G}_t^2(\mathbf{\Theta}_2)$, focuses on learning the inverse of the innovation covariance matrix $\mathbf{S}_{t|t-1}^{-1}$ implicitly. The Kalman gain matrix is then constructed as $\mathbf{K}_t(\mathbf{\Theta}_1,\mathbf{\Theta}_2,\mathbf{H}_t)=\mathcal{G}_t^1(\mathbf{\Theta}_1)\mathbf{H}_t^T\mathcal{G}_t^2(\mathbf{\Theta}_2)$. This architectural split enables SKN to effectively handle discrepancies in process and measurement noises. For further details of SKN, we refer to [14].

III. Cholesky-KalmanNet

A. Uncertainty Quantification in DNN-aided KFs

Klein et al. [26] introduced a method for KN to estimate the error covariance, $\hat{\mathbf{\Sigma}}_{t|t}$, through analytical matrix operations, involving an inverse of matrix product, $\tilde{\mathbf{H}}_t=(\mathbf{H}_t\mathbf{H}_t^T)^{-1}$, which requires $\mathbf{H}_t$ to be full-column rank, thus limiting its applicability. Additionally, this approach may result in non-positive definite covariance matrices. SKN, in contrast, uses RNNs to estimate $\mathbf{\Sigma}_{t|t-1}$ and $\mathbf{S}_{t|t-1}$, and subsequently derives the posterior covariance $\hat{\mathbf{\Sigma}}_{t|t}$ [27]. However, the covariances estimated from RNNs may not consistently be positive definite, leading to inaccurate or even invalid UQ.

To address these issues, we propose integrating a new neural layer within the RNN architecture, which enforces positive definiteness in covariance matrices, significantly enhancing stability and applicability across diverse contexts.

B. Positive Definite Enforcing Layer for Error Covariance

The proposed CKN integrates a Positive Definite Enforcing Layer (PDEL) within the RNN architecture to ensure that prior and error residual covariance matrices remain positive definite, as shown in Fig. 1. The PDEL leverages Cholesky decomposition to produce positive definite matrices from any real input [28]. For a symmetric matrix $\mathbf{C}$, the decomposition is $\mathbf{C}=\mathbf{L}\mathbf{L}^T$, where $\mathbf{L}$ is a lower triangular matrix with positive diagonal values, ensuring $\mathbf{C}$ is positive definite.

In CKN, the PDEL transforms a vector $\mathbf{A}_{p\times 1}$ from RNN outputs into a positive definite matrix $\mathbf{C}_{q\times q}$, with the relation $p=q(q+1)/2$. This process involves reshaping $\mathbf{A}$ into a lower triangular matrix $\mathbf{L}'$ as:

$$\mathbf{A}=\begin{bmatrix}a_1\\a_2\\\vdots\\a_p\end{bmatrix}\rightarrow\mathbf{L}'=\begin{bmatrix}p(a_1)&&&\\a_2&p(a_3)&&\mathbf{0}\\\vdots&\vdots&\ddots&\\\cdots&\cdots&\cdots&p(a_p)\end{bmatrix} \tag{4}$$

where the function $p(\cdot)$ represents any strictly positive function, such as exponential, square, or ReLU. In this work, we empirically selected the absolute function, defined as $p(a)=|a|+\varepsilon$ where $\varepsilon>0$ is a small positive constant ensuring positive diagonal terms. This function was chosen empirically, as it provided smoother and more effective training compared to other functions such as exponential or ReLU, leading to more accurate results. The positive definite matrix is then obtained as $\mathbf{C}=\mathbf{L}'\mathbf{L}'^T$. In addition, PDEL reduces the number of entries in the covariance matrix by nearly 50%, allowing efficient training on smaller datasets, which is beneficial in real-world scenarios with limited data.

The matrix $(\mathbf{I}-\mathbf{K}_t\mathbf{H}_t)$, in (3b), is essential for estimating $\hat{\mathbf{\Sigma}}_{t|t}$ by adjusting the prior covariance $\hat{\mathbf{\Sigma}}_{t|t-1}$ with new measurement information. This matrix is typically non-symmetric and not inherently positive definite. Nevertheless, when $\mathbf{H}_t$ and $\mathbf{R}_t$ are such that $\mathbf{K}_t$ remains well-bounded, it generally serves as a stable transformation matrix.

C. Training to Predict Error Covariance

The additional PDEL enables the estimation of the positive definite posterior covariance $\hat{\mathbf{\Sigma}}_{t|t}$. However, training based on (6), focused solely on state estimation, does not ensure that $\hat{\mathbf{\Sigma}}_{t|t}$ accurately reflects the true error covariance. To address this, we propose an alternative loss function as:

$$\mathcal{L}^{\text{total}}=(1-\beta)\mathcal{L}^{\ell_2}+\beta\mathcal{L}^{cov} \tag{5}$$

where

$$\mathcal{L}^{\ell_2}=\frac{1}{N}\sum_{i=1}^{N}\frac{1}{T_i}\sum_{t=1}^{T_i}\left\|\mathbf{x}_t^{(i)}-\hat{\mathbf{x}}_{t|t}^{(i)}\right\|_2^2,\text{and} \tag{6}$$

$$\mathcal{L}^{cov}=\frac{1}{N}\sum_{i=1}^{N}\frac{1}{T_i}\sum_{t=1}^{T_i}\sum_{j=1}^{m}\sum_{k=1}^{m}\left|\left[\mathbf{e}_t^{(i)}\mathbf{e}_t^{(i)T}\right]_{j,k}-\left[\hat{\mathbf{\Sigma}}_{t|t}^{(i)}\right]_{j,k}\right|. \tag{7}$$

The superscript (i) denotes the i-th data sample; for example, $\mathbf{x}_t^{(i)}$ represents the true state at time step t for the i-th sample. Additionally, $\mathbf{e}_t^{(i)}=\mathbf{x}_t^{(i)}-\hat{\mathbf{x}}_{t|t}^{(i)}$ represents the estimation errors,

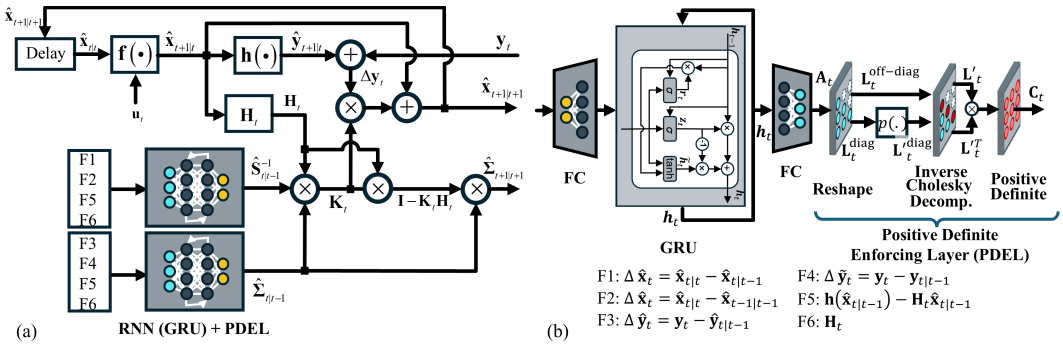

Fig. 1. Block diagrams for CKN: (a) Split-GRU architecture used to estimate the innovation covariance $\hat{\mathbf{S}}_{t|t-1}$ and the state covariance $\hat{\mathbf{P}}_{t|t-1}$. The architecture includes GRU layers combined with the Positive Definite Enforcing Layer (PDEL). (b) Detailed CKN architecture, comprising a fully connected (FC) input layer, followed by a GRU layer, another FC output layer, and ending with the PDEL, which reshapes the output, applies Cholesky decomposition, and enforces positive definiteness in the output covariance matrix.

and $\beta \in [0, 1]$ balances the emphasis between $\mathcal{L}^{\ell_2}$ and $\mathcal{L}^{cov}$. The first term $\mathcal{L}^{\ell_2}$ minimizes the MSE between true and estimated states, ensuring precise state tracking. The second term $\mathcal{L}^{cov}$ minimizes discrepancies between actual residuals and the estimated covariance $\hat{\mathbf{\Sigma}}_{t|t}$, capturing interdependencies among state variables by incorporating both diagonal and off-diagonal elements. Together, these terms enable the network to provide accurate state estimates and reliable error covariance estimates. Furthermore, this total loss function explicitly guides training to produce the posterior covariances that are not only aligned with the observed data discrepancies but also meet the fundamental statistical requirements for valid covariances.

IV. Numerical Examples

In this section, unless stated otherwise, the noise covariances, $\mathbf{Q}_t$ and $\mathbf{R}_t$, are defined as $\sigma_w^2\mathbf{I}$ and $\sigma_v^2\mathbf{I}$, respectively. We compare EKF, KN, SKN, and CKN. To model the variability of the noise, we define a noise ratio $\nu = \sigma_v^2/\sigma_w^2$. The code for CKN has been made available via GitHub.[1]

A. Synthetic linear state and nonlinear observation models

The first benchmark example consists of linear state evolution and nonlinear observation models as [14]:

$$\mathbf{x}_{t+1} = \begin{bmatrix} \cos(\frac{\pi}{18}) & -\sin(\frac{\pi}{18}) \\ \sin(\frac{\pi}{18}) & \cos(\frac{\pi}{18}) \end{bmatrix} \mathbf{x}_t + \mathbf{w}_t \qquad \mathbf{x}_t \in \mathbb{R}^2 \tag{8a}$$

$$\mathbf{y}_t = \begin{bmatrix} \|\mathbf{x}_t\|_2 & \text{atan2}\,(\mathbf{x}_t) \end{bmatrix}^T + \mathbf{v}_t \qquad \mathbf{y}_t \in \mathbb{R}^2 \tag{8b}$$

where $\mathbf{x}_t = [x_1\ x_2]^T$. In this example, we want to examine the DNN-aided KFs under varying noise ratio ν=0 to 50 [dB], while fixing the process variance σ_w^2=10^{-3}.

For each noise ratio, data sets comprising 1,000 training time sequences and 300 testing time sequences were generated synthetically. Each set features different realizations of measurement and process noise, with training and testing data having $T_{\text{train}} = 30$ and $T_{\text{test}} = 50$.

Fig. 2 presents the MSE comparison for a range of ν. CKN consistently achieves the lowest MSE across all ν values, making it the most robust method among those evaluated.

[1][Online]. Available: https://github.com/RAMSIS-Lab/ckn-spl-public

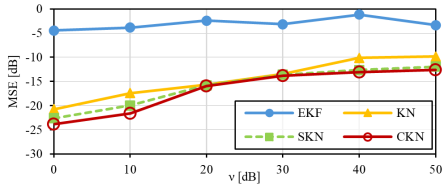

Fig. 2. MSE comparison for different method as varying ν [dB] for IV-A

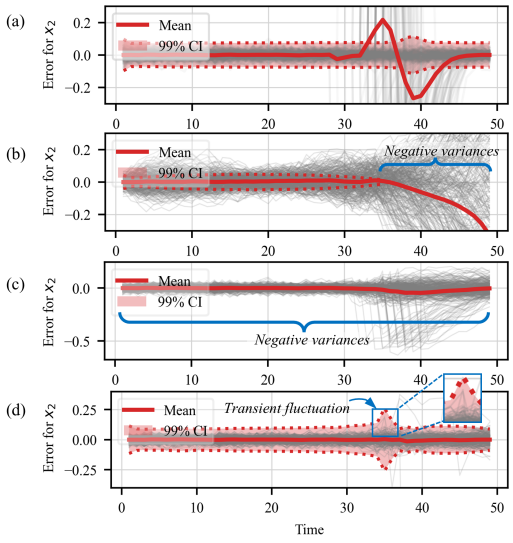

Fig. 3. Comparison for Error and 99% CI for $\nu = 0$ for IV-A: (a) EKF, (b) KN, (c) SKN, and (d) CKN

Figs. 3 present the error plots for x_2 with 99% CIs for a case of ν=0. The gray lines represent the error time series for the 300 individual state estimates. Both EKF and KN exhibit large deviations. Additionally, the observed negative variances for KN is a clear sign of instability in the estimation process. SKN also produces negative variance estimates, which is statistically invalid. In contrast, CKN demonstrates improved accuracy and avoids negative variances observed in KN and SKN. Moreover, the CKN's CIs effectively capture the transient fluctuation around t=35, showcasing CKN's ability for precise UQ.

B. Synthetic nonlinear state and nonlinear observation models

We explore a nonlinear system with sinusoidal state evolution and second-order polynomial observation model [13]:

TABLE I
PARAMETERS OF INACCURATE SYSTEM MODELS FOR IV-B

State-space Model	α	β	ϕ	δ	a	b	c
Accurate	0.9	1.1	0.1π	0.01	1	1	0
Tier 1	1.0	1.0	0.0	0	1	1	0
Tier 2	1.1	0.9	-0.1π	-0.01	1	1	0
Tier 3	1.0	1.0	0.0	0	1	1	0.1

TABLE II
MSE FOR THREE TIERS OF PARAMETERS FOR IV-B

Methods	EKF	KN	SKN	CKN
Tier 1 [dB]	3.64	-23.39	-22.11	**-23.91**
Tier 2 [dB]	8.19	-20.79	-20.76	**-23.73**
Tier 3 [dB]	8.80	-15.47	-13.91	**-19.12**

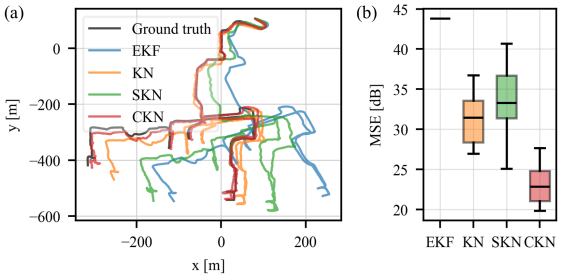

Fig. 4. (a) Estimated trajectory comparison and (b) MSE variation for 10 different data splits for IV-C

$$\mathbf{f}(\mathbf{x}) = \alpha \cdot \sin(\beta \cdot \mathbf{x} + \phi) + \delta \qquad \mathbf{x} \in \mathbb{R}^2 \tag{9a}$$

$$\mathbf{h}(\mathbf{x}) = a \cdot (b \cdot \mathbf{x} + c)^2 \qquad \mathbf{y} \in \mathbb{R}^2 \tag{9b}$$

We consider three cases, i.e., Tier 1, Tier 2, and Tier 3, with increasing levels of inaccuracy in system parameters as tabulated in Table I. In the following, training and testing data were generated with σ_w^2=10^{-3} and ν=10 [dB].

Table II summarizes the MSE values, where the results underscore CKN's ability to maintain high accuracy and reliable performance even under significant model mismatches, making it the most robust method among those evaluated.

C. Real World Dynamics: Michigan NCLT Data Set

We evaluate CKN using the Michigan NCLT dataset [29]. This dataset includes various labeled trajectories, documenting noisy sensor data such as GPS and odometer readings, along with the ground truth positions of a Segway robot. A nearly constant acceleration motion model is adopted as the state evolution model [30]. Our goal is to track the underlying state vector using only noisy velocity observations.

We selected the session dated 2012-01-22 with 5,850 time steps. After preprocessing, we retained 5,150 time steps (103 sequences of T=50) and split the trajectory into two sections: 80 sequences for training and 10 sequences for validation. For testing, we used another trajectory dated 2012-04-29.

An ideal stable model should exhibit minimal performance variation across different data splits. To evaluate the stability of DNN-aided KFs under varying training and testing conditions, we trained networks using three different algorithms (KN, SKN, and CKN) 10 times. Each training instance involved the

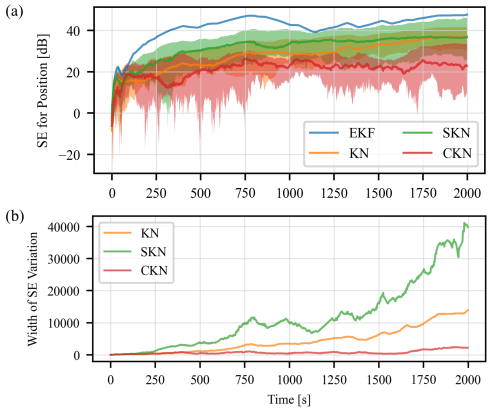

Fig. 5. Performance variance for 10 runs of DNN-aided KFs for IV-C: (a) Median of SE for position with a band of SE variation and (b) Width of SE variation for testing data

random selection of 80 training sequences and 10 validation sequences from the 103 shuffled sequences of $T = 50$.

Fig. 4(a) presents one of the 10 different estimated trajectories for each algorithm together with the ground truth trajectory. The CKN result exhibits the closest alignment to the ground truth. Fig. 4(b) shows the box plot of MSE variations of position, illustrating that CKN achieves the lowest MSE values overall as well as the smallest MSE variation.

Fig. 5(a) illustrates the Squared Error (SE) of position with respect to time for each algorithm together with the width of SE variation for the 10 different runs. The solid line represents the median SE for the 10 runs. Results indicate that CKN consistently achieves the lowest median SE at most time steps. The shaded area represents the width of SE variations for the 10 runs. Fig. 5(b) depicts the width of SE variations (maximum SE - minimum SE) for the 10 runs for each DNN-aided KF due to variance in training. The narrowest width of SE variations for CKN indicates its superior stability in performance.

V. CONCLUSION

Our proposed approach, Cholesky-KalmanNet (CKN), enhances the accuracy, robustness, and stability of state and error covariance estimation. By integrating a new layer to embed mathematical constraints through the Cholesky decomposition, CKN ensures that the resulting posterior error covariance inherently satisfies the condition of positive definiteness, which is crucial for the statistical accuracy of state estimation. Furthermore, we propose a novel hybrid loss function that minimizes discrepancies between the estimated and empirical error covariance, accounting for interdependencies among state variables. This comprehensive loss function maintains the positive definiteness throughout the training phase and ensures a thorough minimization of estimation errors, even effectively capturing transient fluctuations in error estimates. Extensive evaluations on both synthetic and real-world datasets demonstrate CKN's superior performance, offering more accurate state estimations and greater robustness against system inaccuracies and observation noise. CKN also shows enhanced stability across varying training data partitions. These attributes position CKN as an accurate and stable method for state estimation in complex and noisy environments.

REFERENCES

[1] O. El-Khoury, C. Kim, A. Shafieezadeh, J. E. Hur, and G. H. Heo, "Mitigation of the seismic response of multi-span bridges using MR dampers: Experimental study of a new SMC-based controller," *Journal of vibration and control*, vol. 24, no. 1, pp. 83–99, 2018.

[2] M. Sun, M. E. Davies, I. K. Proudler, and J. R. Hopgood, "Adaptive kernel Kalman filter based belief propagation algorithm for maneuvering multi-target tracking," *IEEE Signal Processing Letters*, vol. 29, pp. 1452–1456, 2022.

[3] Z. Wei, G. Dong, X. Zhang, J. Pou, Z. Quan, and H. He, "Noise-immune model identification and state-of-charge estimation for lithium-ion battery using bilinear parameterization," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 1, pp. 312–323, 2020.

[4] S. Battilotti, "Performance optimization via sequential processing for nonlinear state estimation of noisy systems," *IEEE Transactions on Automatic Control*, vol. 67, no. 6, pp. 2957–2972, 2021.

[5] Y. Wang, Y. Sun, and V. Dinavahi, "Robust forecasting-aided state estimation for power system against uncertainties," *IEEE Transactions on Power Systems*, vol. 35, no. 1, pp. 691–702, 2019.

[6] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 03 1960.

[7] A. H. Jazwinski, *Stochastic processes and filtering theory*. Courier Corporation, 2007.

[8] R. Zhan and J. Wan, "Neural network-aided adaptive unscented Kalman filter for nonlinear state estimation," *IEEE Signal Processing Letters*, vol. 13, no. 7, pp. 445–448, 2006.

[9] S. Julier, J. Uhlmann, and H. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *IEEE Transactions on Automatic Control*, vol. 45, no. 3, pp. 477–482, 2000.

[10] J. Qi, K. Sun, J. Wang, and H. Liu, "Dynamic state estimation for multi-machine power system by unscented Kalman filter with enhanced numerical stability," *IEEE Transactions on Smart Grid*, vol. 9, no. 2, pp. 1184–1196, 2016.

[11] I. Grooms, Y. Lee, and A. J. Majda, "Ensemble Kalman filters for dynamical systems with unresolved turbulence," *Journal of Computational Physics*, vol. 273, pp. 435–452, 2014.

[12] R. Dehghannasiri, M. S. Esfahani, and E. R. Dougherty, "Intrinsically Bayesian robust Kalman filter: An innovation process approach," *IEEE Transactions on Signal Processing*, vol. 65, no. 10, pp. 2531–2546, 2017.

[13] G. Revach, N. Shlezinger, X. Ni, A. L. Escoriza, R. J. G. van Sloun, and Y. C. Eldar, "KalmanNet: Neural network aided Kalman filtering for partially known dynamics," *IEEE Transactions on Signal Processing*, vol. 70, pp. 1532–1547, 2022.

[14] G. Choi, J. Park, N. Shlezinger, Y. C. Eldar, and N. Lee, "Split-KalmanNet: A robust model-based deep learning approach for state estimation," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 9, pp. 12 326–12 331, 2023.

[15] I. Buchnik, G. Revach, D. Steger, R. J. Van Sloun, T. Routtenberg, and N. Shlezinger, "Latent-KalmanNet: Learned Kalman filtering for tracking from high-dimensional signals," *IEEE Transactions on Signal Processing*, 2023.

[16] I. Buchnik, G. Sagi, N. Leinwand, Y. Loya, N. Shlezinger, and T. Routtenberg, "GSP-KalmanNet: Tracking graph signals via neural-aided Kalman filtering," *arXiv preprint arXiv:2311.16602*, 2023.

[17] D. Shen, Y. Ma, G. Liu, J. Hu, Q. Weng, and X. Zhu, "Dynamical variational autoencoders and Kalmannet: New approaches to robust high-precision navigation," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 48, pp. 1141–1146, 2023.

[18] X. Ni, G. Revach, and N. Shlezinger, "Adaptive Kalmannet: Data-driven Kalman filter with fast adaptation," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 5970–5974.

[19] P. Hao, O. Karakus, and A. Achim, "RKFNet: A novel neural network aided robust Kalman filter," *arXiv preprint arXiv:2403.16756*, 2024.

[20] N. Piperigkos, A. Gkillas, C. Anagnostopoulos, and A. S. Lalos, "Cooperative Plug-and-Play-Kalmannet for 4d situational awareness in autonomous driving," in *2024 IEEE 7th International Conference on Industrial Cyber-Physical Systems (ICPS)*. IEEE, 2024, pp. 1–6.

[21] Z. Quan, S. Han, and W. H. Kwon, "A robust FIR filter for linear discrete-time state–space signal models with uncertainties," *IEEE Signal Processing Letters*, vol. 14, no. 8, pp. 553–556, 2007.

[22] Z. K. Nagy and R. D. Braatz, "Robust nonlinear model predictive control of batch processes," *AIChE Journal*, vol. 49, no. 7, pp. 1776–1786, 2003.

[23] S. Wang, Z. Wu, and A. Lim, "Robust state estimation for linear systems under distributional uncertainty," *IEEE Transactions on Signal Processing*, vol. 69, pp. 5963–5978, 2021.

[24] M.-Y. Zhao, W.-J. Yan, K.-V. Yuen, and M. Beer, "Non-probabilistic uncertainty quantification for dynamic characterization functions using complex ratio interval arithmetic operation of multidimensional parallelepiped model," *Mechanical Systems and Signal Processing*, vol. 156, p. 107559, 2021.

[25] M. Imholz, M. Faes, D. Vandepitte, and D. Moens, "Robust uncertainty quantification in structural dynamics under scarse experimental modal data: A Bayesian-interval approach," *Journal of Sound and Vibration*, vol. 467, p. 114983, 2020.

[26] I. Klein, G. Revach, N. Shlezinger, J. E. Mehr, R. J. van Sloun, and Y. C. Eldar, "Uncertainty in data-driven Kalman filtering for partially known state-space models," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 3194–3198.

[27] Y. Dahan, G. Revach, J. Dunik, and N. Shlezinger, "Uncertainty quantification in deep learning based Kalman filters," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 13 121–13 125.

[28] K. Xu, D. Z. Huang, and E. Darve, "Learning constitutive relations using symmetric positive definite neural networks," *Journal of Computational Physics*, vol. 428, p. 110072, 2021.

[29] N. Carlevaris-Bianco, A. K. Ushani, and R. M. Eustice, "University of Michigan North Campus long-term vision and lidar dataset," *International Journal of Robotics Research*, vol. 35, no. 9, pp. 1023–1035, 2015.

[30] M. A. Richards, J. A. Scheer, and W. A. Holm, Eds., *Principles of Modern Radar: Basic Principles*. Raleigh, N.C.: SciTech Publishing, 2010.