Data Assimilation with surrogate measurement equations

Marco Arzeo[2], Stefano Chessa[2], Fabio Marcuzzi[1], Pietro Paglierani[2], and Joao Alves[2]

[1] Department of Mathematics "Tullio Levi Civita", University of Padova, Via Trieste 63, 35121 Padova, Italy
marcuzzi@math.unipd.it
[2] NATO STO Centre for Maritime Research and Experimentation, La Spezia, Italy
pietro.paglierani@cmre.nato.int

Abstract. Classical data-assimilation methods, like the Kalman Filter, and even most recent formulations, are model-based methods where it is assumed that the relation between the state-vector and the measurements can be expressed analytically up to a random error component. Here we refer to situations where the available analytical expression for the measurement equation is a surrogate of the true one, in the sense that the former approximates the latter with not negligible deterministic discrepancies. Moreover, the analytic expression of these discrepancies, if it exists, is presumed to be very unlikely to be found at a reasonable cost. We also assume that an accurate measurement equation exists for measurements that can be done only in laboratory experiments. The aim of this paper is to show that a Deep Kalman Filter can use a surrogate measurement equation to form the innovations and optimally estimate the state-vector, when the supervised learning of the corrector-gain matrices has been done using an accurate measurement equation (and corresponding data). The resulting method is trajectory-dependent, in principle, but can generalize to multiple trajectories. As a general test, we show some results with multiple abstract dynamical systems and measurement equations with additive piecewise-polynomial and/or trigonometric biases.

Keywords: Kalman filtering · Deep unfolding · surrogate measurements.

1 Introduction

A current trend is the incorporation of machine learning (ML) in data assimilation, in various forms, see e.g. the recent review [5].

There is a common assumption in data-assimilation methods, i.e. that the measurement equation is accurate, and the ML effort is more concentrated, e.g. on using machine learning to map the joint predicted state and observation to the updated state estimate [3], to learn the dynamical error in the predictor [12] [6] and/or to learn compression mechanisms for high-dimensional measurements [4].

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_14

Here, instead, we leverage machine learning for adapting the data-assimilation algorithm to work with surrogate measurement equations that, otherwise, would deviate the state-trajectory estimate far from the real one. We will study this adaptation in the context of the Deep Kalman Filter (DKF) [6], which implements a predictor-corrector scheme similar to the Kalman Filter (KF), but the corrector gains are learned from data. Related approaches that combine Kalman-type filters with learned or surrogate observation operators have recently appeared in several application domains. For instance, in satellite data assimilation, machine-learning-based observation operators have been trained to emulate complex radiative-transfer models and associated bias-correction schemes, and then used within an ensemble Kalman filter without explicit access to the underlying physical measurement equation [13]. In robotics, the Koopman-Inspired Learned Observations Extended Kalman Filter (KILO-EKF) learns a measurement model for complex or poorly calibrated sensors by lifting raw measurements into a feature space where they are linearly related to the state, thereby enabling an EKF correction step driven by a data-driven observation operator rather than an analytic sensor model [10]. In environmental monitoring, deep neural surrogates of high-fidelity forward models have been embedded in sequential Monte Carlo schemes to perform real-time Bayesian inversion of gas emissions, effectively replacing an expensive but accurate measurement equation with a learned surrogate while preserving the recursive Bayesian estimation structure [16]. In a similar spirit, Latent Assimilation combines a convolutional autoencoder, a recurrent neural network surrogate of the dynamics, and an Optimal Interpolated Kalman Filter in the latent space, so that the effective observation mapping is mediated by a learned encoder–decoder rather than a simple analytic operator on the full physical state [1]. In structural health monitoring, Neural Extended Kalman Filters parameterize both the process dynamics and the sensory observation mapping with neural networks and train them end-to-end under a variational-inference framework, yielding an EKF-like architecture with a learned measurement model that can predict structural responses on simulated and real datasets [14]. In oceanography, LSTM-based Kalman filters have been proposed for data assimilation of spatio-temporal ocean currents, combining LSTM-derived state dynamics with glider and radar observations to produce short-term forecasts for path planning [18]. In parallel, several "deep Kalman" formulations have been proposed in which the predictor–corrector structure of the Kalman filter is retained, but the correction step is parametrized by neural networks trained from data, as in Neural Assimilation [2] and Recursive KalmanNet [15]. These works collectively demonstrate that it is possible either to learn surrogate measurement equations or to learn data-driven correctors that compensate for nonideal measurement models. However, they typically assume that the observation model used in training and in operation coincide, or do not explicitly address the case where only a biased surrogate measurement equation is available in the field while a more accurate measurement equation exists only in laboratory conditions. In contrast, the present work considers precisely this scenario and shows that a Deep Kalman Filter can be trained, using laboratory data

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_14

and an accurate measurement equation, to compute trajectory-dependent gains that optimally exploit innovations formed with a surrogate measurement equation, thereby mitigating its deterministic discrepancies and recovering accurate state estimates even when the operational observation operator is structurally imperfect.

2 Model problem

The *reference model*, used by the Deep Kalman Filter as (deterministic) predictor, is a general, discrete-time, nonlinear state-space model (SSM) of the form:

$$\begin{aligned} x(k+1) &= f(\ x(k),\ p,\ u(k)+w_u(k)\ ) + w_m(k) \\ y(k) &= h(\ x(k)\ ) + w_y(k) \end{aligned} \tag{1}$$

where $f = f(x, p, u)$ is a p-parametric transition map acting on both current input $u(k) \in R^{n_u}$ and state $x(k) \in R^{n_x}$ vectors; $w_u(k)$ is input measurement noise and $w_m(k) \in R^{n_x}$ is model noise; $y(k) \in R^{n_y}$ is current output and $w_y(k) \in R^{n_y}$ is output measurement noise. Note that in classic Kalman filtering the input is assumed to be known exactly and input noise is eventually incorporated in the model noise, while here we assume both that we can simulate our reference model only with noisy inputs, and that the true model is driven by the exact, noiseless inputs, i.e. we have an input-measurement noise.A common choice is $w_m(k) \sim \mathcal{N}(0, Q)$, $w_y(k) \sim \mathcal{N}(0, R)$ and $w_u(k) \sim \mathcal{N}(0, Z)$, but not the only possible. The measurement equation $h(x(k))$ is a general nonlinear function of the states, with an important special case $h(x(k)) = C\ x(k)$, i.e. a linear relation between measurements and state variables. For an assumed optimal choice of the map f, the parameters and the noise models, this reference model approximates a true, at least partially unknown, state-transition model, which is supposed of the form:

$$\begin{aligned} x(k+1) &= f^*(x(k), u(k)+w_u^*(k)) + w_m^*(k) \\ y(k) &= h^*(x(k)) + w_y^*(k) \end{aligned} \tag{2}$$

where a classic case is $w_m^*(k) \sim \mathcal{N}(0, Q^*)$, $w_y^*(k) \sim \mathcal{N}(0, R^*)$ and $w_u^*(k) \sim \mathcal{N}(0, Z^*)$, but not the only possible here. This true model is intended to generate the measurement data. For simplicity, without loss of generality here we concentrate on a model problem without a deterministic model error in the predictor (see e.g. [12] for a clarification of this concept). Instead, we consider the presence of a deterministic model error in the measurement equation. Precisely, if the sensor data are defined as $\tilde{y}(k) = \tilde{h}(x(k)) + \tilde{\omega}(k)$, where $\tilde{h}(x(k))$ is analytically unknown, we assume that in laboratory tests it holds that:

$$\tilde{h}(x(k)) = h(x(k)) = h^*(x(k)) \quad , \tag{3}$$

while in general operating conditions there is available only a *surrogate* measurement equation $h_s(x(k))$, i.e.

$$h_s(x(k)) = h^*(x(k)) + m_h(x(k)) + \omega_h(x(k)) \tag{4}$$

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_14

with $m_h(x(k))$ a deterministic error component and $\omega_h(x(k))$ a stochastic error component. The notation adopted for the different introduced measurement equations is listed in Table 1.

h^*	the measurement equation of the true model
$\tilde{h}$	the measurement equation of the sensor
h	the measurement equation used by the Deep Kalman Filter
h_s	the surrogate measurement equation available in field operations

Table 1. Notation used in the text for the measurement equations.

In the numerical experiments (see Sec. 4), we will refer to the following three case studies:

1. $\tilde{h}(x(k)) = h^*(x(k))$ and $h(x(k)) = h_s(x(k))$, where we are able to do very good measurements but we don't have an accurate analytical expression of the measurements as a function of the state vector;
2. $\tilde{h}(x(k)) = h_s(x(k))$ and $h(x(k)) = h^*(x(k))$, where we know an accurate analytical description of the measurements as a function of the state vector, but we have actual measurements that are surrogates of the real ones;
3. $\tilde{h}(x(k)) = h_s^{'}(x(k))$ and $h(x(k)) = h_s^{''}(x(k))$, where both the measurements and the measurement equation are surrogate, but not the same.

3 The Deep Kalman Filter for surrogate measurement equations

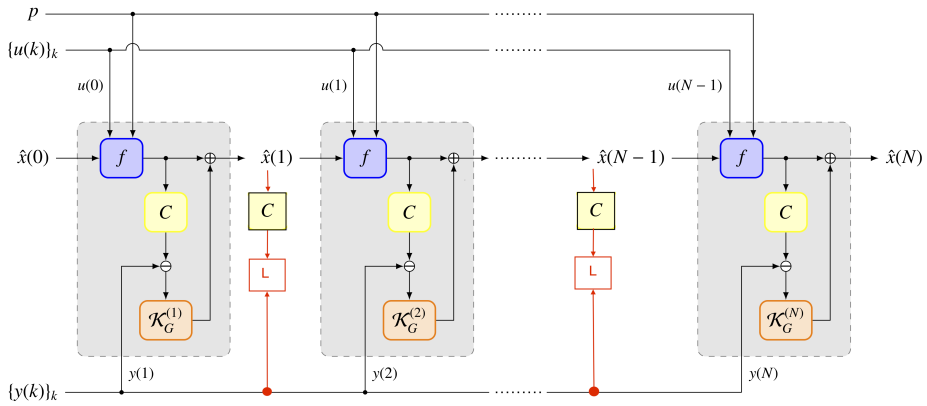
In this section we briefly present the Deep Kalman Filter equations in a schematic form. Given a measurable final state $x(N)$ we consider the following loss function:

$$\begin{aligned}\mathcal{L} = &\frac{\lambda_N}{2}\|\hat{x}(N) - x(N)\|_2^2 \; + \; \frac{1}{2}\sum_{l=1}^{N}\|y(l) - C\hat{x}(l)\|^2_{R_l^{-1}} \; + \\ &+ \frac{1}{2}\sum_{l=1}^{N}\|\mathcal{K}_G^{(l)}\left(y(l) - C\hat{x}(l\,|\,l-1)\right)\|^2_{P_l^{-1}} + \frac{\lambda_D}{2}\sum_{i,j=1}^{m,n}\|Dv^{ij}\|_2^2 = \qquad (5)\\ &= \mathcal{E}_{\lambda_N} + \sum_{l=1}^{N}\mathcal{F}^l + \sum_{l=1}^{N}\mathcal{G}^l + \mathcal{H}_{\lambda_D}.\end{aligned}$$

where R_l, P_l are symmetric weight matrices, D is the second order finite differences matrix and $v^{ij} = \left[(\mathcal{K}_G^{(1)})_{ij} \; \ldots \; (\mathcal{K}_G^{(N)})_{ij}\right]^\top$. For a in-depth description of the contribution given by each term and its computational involvement in the optimization process, see [6]. In this paper we present the modification from the standard form of the DKF, schematically depicted in Figure 1a, to the evolved form that can exploit surrogate measurement functions, depicted in Figure 1b.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_14

a) DKF standard formulation [6]:



b) DKF evolved formulation for surrogate measurement equations:

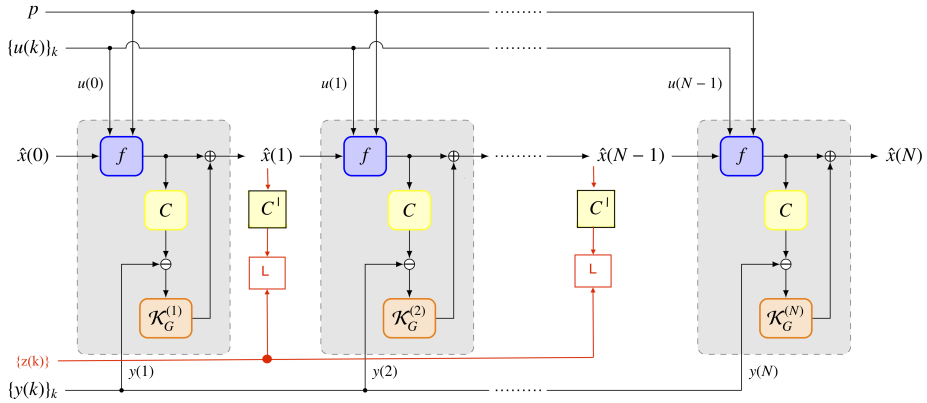


Fig. 1. At the top part ("a)") is shown the structure of the standard Deep Kalman Filter network. Each layer is contained in a gray box. The l-th layer receives a state estimate $\hat{x}(l-1)$, measurement $y(l)$, input $u(l-1)$, parameter p, and outputs the new estimated state $\hat{x}(l)$. The predictor component f (blue box) receives $u(l-1)$ and p to form the predicted state $\hat{x}(l\,|\,l-1)$. The corrector component, composed of the observation matrix C (yellow box), or of an arbitrary measurement equation h, and gain matrix $\mathcal{K}_G^{(l)}$ (orange box), receives $y(l)$ and computes the correction $\mathcal{K}_G^{(l)} e_{\hat{y}}(k)$. Predicted-state and correction are then summed to form the new state estimate. The learnable weights are the gains $\{\mathcal{K}_G^{(l)}\}_{l=1,\ldots,N}$ and parameters p. The loss-function "L" (in red) (5) of back-propagation learning uses the same measurements and the same measurement equation used by the filter. At the bottom part ("b)") is shown the evolved formulation for surrogate measurement equations: the difference is that here the loss-function "L" (in red) 5 uses different measurements and measurement-equation than those used by the filter.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_14

The standard form of the DKF is derived adopting a common assumption in data-assimilation methods, i.e. that the measurement equation is accurate. The DKF aims at estimating the state-vector of a dynamical system by using a deterministic predictor (the function f in (1) and shown in Figure 1) and a correction scheme based on the product between learned gain matrices and vectors containing the output prediction error (see Fig. 1). This error, called *innovation*, is the comparison between measurements and their prediction obtained through the state-prediction and a measurement equation. For situations where measurements corresponding to a precise measurement equation are available only during laboratory tests, we show in this section how the information produced in these tests can be used by the DKF also in an operational context where a less precise measurement equation is applicable, i.e. a surrogate measurement function.

In Figure 1a it is shown the standard situation depicted so far, where we can notice in black color the *forward path*, where the output prediction errors are multiplied by the gains matrix $\mathcal{K}_G^{(l)}$ to correct the state-prediction. In red color there is the *backward path*, existing only during the learning process, that uses also the output filter errors to compute the loss function L (5) and then update the matrices $\mathcal{K}_G^{(l)}$ by backpropagation learning. Note that here the measurement equations are the same, i.e. the matrix C (but it could be an arbitrary nonlinear function h, as said before). Then, when the learning process is finished, the red part is removed and the DKF behaves like has been described in the previous sections.

If in the operational context only a surrogate measurement function can be used, the evolved scheme here presented, shown in Figure 1b, has a modified backward path. Indeed, here we assume that $y(k)$ is the measurement data comparable with the output of the surrogate measurement equation (C in the Figure), that the filter can use, but for backpropagation learning we introduce into the loss function $\mathcal{L}$ the high-precision measurements $z(k)$ and the comparison with the corresponding measurement equation (C' in the Figure). In this way, the DKF learns how to accurately track the state-vector even when it uses a measurement function which is not accurate. When the learning process is finished, the red part is removed and the learned filter (black part) is able to repeat the same trajectory by using a surrogate measurement function. Note that the optimization given by this learning method is in principle trajectory-dependent, but has the ability to generalize, as we will see in the following sections.

4 Numerical experiments

Here we assume that in lab experiments we have a trivial linear measurement equation, i.e. the identity $h(x(k)) = C\ x = I\ x = x$, where I is the identity matrix, whereas in the field operation we will consider the three cases defined in Sec. 2. The surrogate nonlinear map $h_s(x)$ is described in the next subsection. The experiment consists in estimating a trajectory in either a bi-dimensional or a tri-dimensional state-space.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_14

4.1 The surrogate measurement equation $h_s(x)$

Let $x \in \mathbb{R}^d$ be the state vector. The surrogate measurement equation is defined as:

$$h_s(x) = x + \alpha_0 h_0 + \alpha_1 H_1 x + \alpha_2 \underbrace{x \cdot e^{x^T \Sigma x}}_{\text{Exp. Scaling}} + \alpha_3 \sin(kx) \; . \tag{6}$$

The function relies on the following parameters:

- **Bias Vector (h_0):** A constant offset vector in $\mathbb{R}^d$.
- **Linear Matrix (H_1):** A transformation matrix in $\mathbb{R}^{d \times d}$.
- **Quadratic Matrix (Σ):** A symmetric matrix in $\mathbb{R}^{d \times d}$ (often positive semi-definite) representing the covariance or metric of the quadratic form.
- **Wave vector (k):** A vector determining the frequency of the oscillatory cosine term in the state space.
- **Noise attenuation (α_i):** Scalar parameters that allow us to "tune" the noise contribution of each component in the function.

The function is composed of four distinct structural components. At this stage, we did not enforce any specific meaning to the shape (and incidentally to the components) of the function. We added terms that could be interpreted as "representatives" of the typical basis employed for function approximation, e.g. polynomial, harmonic, (exponential) kernel functions in order to have a superficial understanding of the flexibility of our approach in this sense. Nevertheless, the considered measurement equation is also thought to capture possible noise contributions when sensing inertial and/or geophysical quantities. First, sensors often introduce a persistent offset (here represented as $\alpha_0 h_0$) that can rapidly result in large errors when performing state estimations. In real-scenarios this bias is rarely constant, but rather a stochastic process often modeled as a first-order Gauss-Markov process or a random walk (bias drift) [8]. In conventional data-assimilation methods, such as the Kalman filter, the detrimental effect of a bias term is typically mitigated by accounting it as an additional state variable (state augmentation) [7]. Additionally, measurement errors can also evolve through complex polynomial (here limited to a linear component $\alpha_1 H_1 x$) and periodic dynamics (here expressed as $\alpha_3 \sin(kx)$). While polynomial terms typically reflect sensor non-linearities, aging, and scale-factor instabilities, periodic errors represent platform resonances and environmental coupling such as mechanical vibrations, thermal cycling [17], or cyclic satellite signal multipath [9] in integrated systems. Finally, the exponential magnitude scaling is included to capture possible correlations between state variables. Other transient effects characterized by exponential decay could also appear due to abrupt variations of the external environment conditions (e.g. temperature, powering, etc.). Functionally speaking, the components are:

Linear Component

$$L(x) = x + h_0 + H_1 x$$

This represents the affine transformation of the input, preserving the baseline geometry and orientation.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_14

Exponential Magnitude Scaling

$$E(x) = x \cdot e^{x^T \Sigma x}$$

The term $x^T \Sigma x$ computes a scalar quadratic form (representing the "energy" or Mahalanobis distance of the vector). Note that:

- Since the exponent is a scalar, it acts as a non-linear gain.
- The original vector direction x is preserved, but its magnitude is scaled exponentially based on its alignment with Σ.

Due to the rapid growth of the exponential function, the term $e^{x^T \Sigma x}$ can dominate the output or cause numerical overflow. To mitigate this, a temperature scaling parameter τ is often introduced. The stabilized exponential term is written as:

$$E_{\text{stable}}(x) = x \cdot e^{\frac{1}{\tau} x^T \Sigma x}, \tag{7}$$

where $\tau > 0$ acts as a dampening factor. This is mathematically equivalent to scaling the matrix Σ by $1/\tau$.

Periodic Component

$$P(x) = \cos(kx)$$

This term introduces (possibly) periodic non-linearities. The cosine function is applied element-wise to the vector kx.

In Figure 2 we see the effect of each term in the function $h_s(k)$, i.e. only one α_i different from zero.

4.2 ML contribution to the data assimilation

Let us consider an example, where the model problem is simplified to avoid unnecessary phenomena (in accordance with the Occam's razor): $f^*(x, p, u) = A\ x + B\ u$, where $A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ and $B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, i.e. there is no model error ($Q^* \approx 0$).

A more complex dynamical system would add observability issues that are per-se well studied in data assimilation methods. Therefore, here, we primarily focus on the effects of a misleading measurement equation. Nevertheless, in Sec. 4.4 we will see that the obtained results hold also with a nonlinear dynamical system. In all cases we will assume that $f(x, p, u) = f^*(x, p, u)$.

The true measurement function is simply $h^*(x(k)) = C\ x = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\ x = x$, i.e. there is no measurement error ($R^* \approx 0$). In the lab tests, we assume that $\tilde{h}(x(k)) = h^*(x(k))$, while for the data assimilation we consider the three case studies listed in Sec. 2. The relative results are presented in the following subsections.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_14

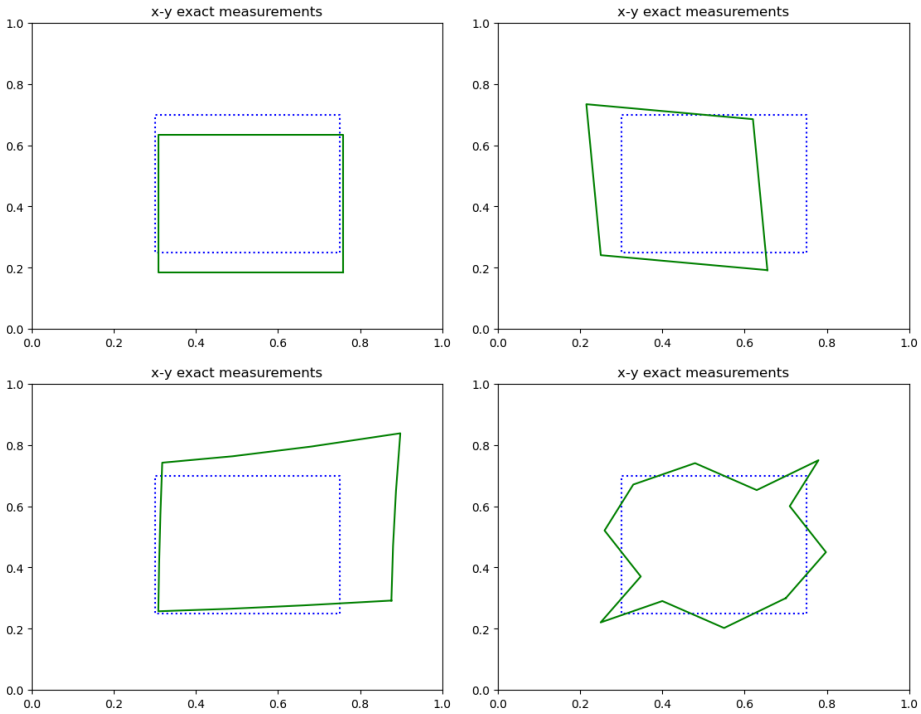


Fig. 2. The effect of each single term in the function $h_s(k)$, i.e. only one α_i different from zero, is shown. Top-left: the constant term (α_0); top-right: the linear term (α_1); bottom-left: the exponential magnitude scaling term (α_2); bottom-right: the periodic component (α_3).

Case study 1 Here $\tilde{h}(x(k)) = h^*(x(k))$ and $h(x(k)) = h_s(x(k))$ defined in Sec. 4.1. In particular, for the experiment presented in Figure 4, $\alpha_0 = \alpha_1 = 0$ and $\alpha_2 = \alpha_3 \neq 0$ for $h_s(x(k))$. The trajectory estimates obtained with the KF and the DKF are respectively shown in Figure 3 (a) and (b).

Case study 2 Here $\tilde{h}(x(k)) = h_s(x(k))$ defined in Sec. 4.1 and $h(x(k)) = h^*(x(k))$. In particular, for the experiment presented in Figure 4, $\alpha_0 = \alpha_1 = 0$ and $\alpha_2 = \alpha_3 \neq 0$ for $h_s(x(k))$. The trajectory estimates obtained with the KF and the DKF are respectively shown in Figure 4 (a) and (b).

Case study 3 Here $\tilde{h}(x(k)) = h_s^{'}(x(k))$ and $h(x(k)) = h_s^{''}(x(k))$, both defined in Sec. 4.1 but with different values of the coefficients α_0, α_1, α_2 and α_3. More specifically, for the experiment presented in Figure 5, $\alpha_0 = \alpha_1 = 0$ and $\alpha_2 = \alpha_3 \neq 0$ for $h_s^{'}(x(k))$, whereas $\alpha_0 \neq 0$ and $\alpha_1 = \alpha_2 = \alpha_3 = 0$ for $h_s^{''}(x(k))$. The trajectory estimates obtained with the KF and the DKF are respectively shown in Figure 5 (a) and (b).

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_14

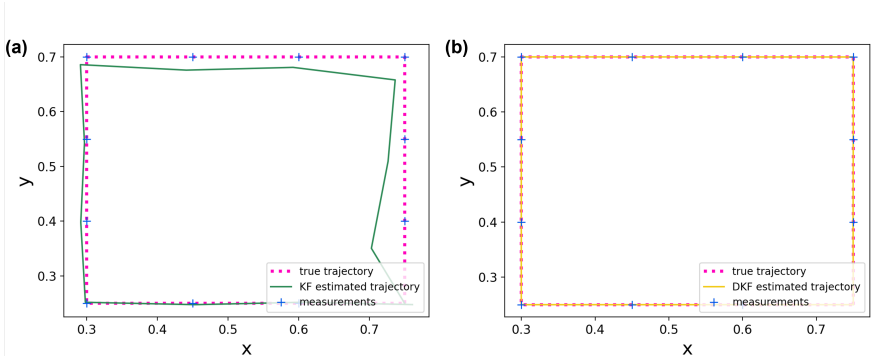


Fig. 3. True trajectory (magenta dotted line) and measurement points (blue cross) for case 1. The estimated trajectories via KF (a) and DKF (b) are respectively plotted as green and orange solid lines.

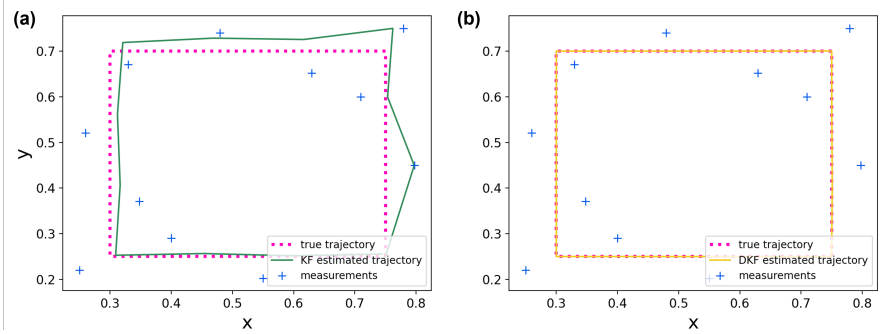


Fig. 4. True trajectory (magenta dotted line) and measurement points (blue cross) for case 2. The estimated trajectories via KF (a) and DKF (b) are respectively plotted as green and orange solid lines.

In all the three studied cases, despite the use of surrogate measurements and/or measurement equation, the DKF provides an estimate of the state-trajectory that accurately reproduce the considered true-trajectory. By comparison, the KF does not result in an accurate estimate.

4.3 Generalization

In Sec. 4.2 we have seen that the gain-learning process operated by the DKF is able to accurately estimate the state-trajectory in all the three case-studies. In those experiments the learning was performed on a single trajectory. Here we see that the DKF is able to generalize to multiple trajectories, and we give an explanation. In Figure 6 we see the simultaneous learning of multiple trajectories in the "Case 3" settings (the most difficult, since both the measurements

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_14

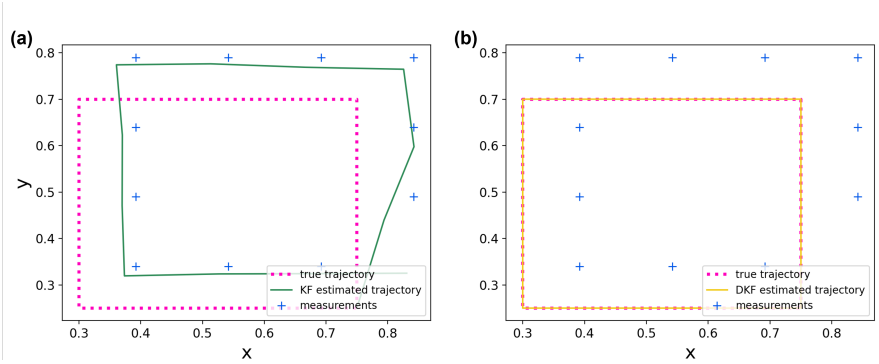


Fig. 5. True trajectory (magenta dotted line) and measurement points (blue cross) for case 3. The estimated trajectories via KF (a) and DKF (b) are respectively plotted as green and orange solid lines.

and the measurement equation are surrogate). The plots in the top row display the behavior of the KF (solid green lines), whereas in the bottom row the corresponding responses of the DKF are shown (solid orange line). The presented results indicate that a single DKF is able to learn multiple different trajectories.

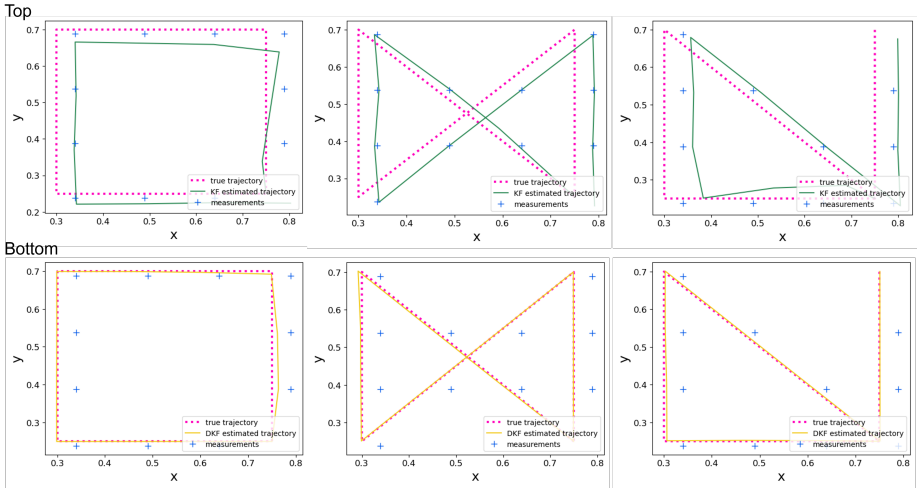


Fig. 6. True trajectory (magenta dotted line) and measurement points (blue cross) for Simultaneous learning of multiple trajectories in the "Case 3" settings. The estimated trajectories via KF (top row) and DKF (bottom row) are respectively plotted as green and orange solid lines.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_14

The example made in this section allows us to do a precise statement about the capacity limit. Since the predictor is exact, $f(x,p,u) = f^*(x,p,u)$, and there are no model nor measurement errors, the optimal correction made by the learned gain matrices $\mathcal{K}_G^{(l)}$ at each layer l should be zero, i.e.

$$\mathcal{K}_G^{(l)}\ e_{\hat{y}}(l) = 0 \tag{8}$$

and this can happen if the DKF learns gain matrices $\mathcal{K}_G^{(l)}$ whose kernel is aligned with the output prediction error $e_{\hat{y}}(l) = y(l) - h\left(\hat{x}(l\,|\,l-1)\right)$.
At each layer l we will have a number of output prediction errors (vectors) equal to the number of trajectories. If these vectors span a subspace of dimension not bigger than that of the biggest possible kernel of the matrices $\mathcal{K}_G^{(l)}$, than the DKF can learn gain matrices satisfing this property, otherwise not. This statement holds also in more general settings: the gain matrices $\mathcal{K}_G^{(l)}$ must be able to filter out the deterministic output error created by the surrogate measurement equations and the DKF make this by shaping their kernels during the learning process.

4.4 Nonlinear dynamics

To further validate the presented method, we tested it against four different examples of nonlinear dynamics: Duffing, Hindmarsh–Rose, Lorenz and van der Pol, two of which are in a tridimensional state. In Figure 7, the tridimensional true trajectory (magenta dotted line) and the measurements points (blue cross) from an Hindmarsh–Rose model [11] are plotted in the "Case 3" settings. In particular, Figure 7a shows the DKF estimates (green solid line) in the standard form (see Figure 1a), i.e. when the learning is carried out on the corrupted measurement equation, which is assumed accurate. In comparison, Figure 7b shows the DKF estimates (green solid line) in the evolved form (see Figure 1b), i.e. when instead a surrogate not accurate measurement equation is taken into account.

The results for all the four tested nonlinear dynamics are summarized in Fig. 8, where the euclidean norm is used as a metric to compare the two DKF approaches described in Sec. 3. The obtained results indicate that the evolved DKF can provide an accurate estimate of the true-trajectory when the standard form fails due to a not accurate measurement equation, even in more complex nonlinear dynamics.

4.5 Comparison with other methods

The key point of the method proposed here is that it uses two measurement equations during filter design: one is used to optimize the state trajectory tracking and the other to tune the corrector gains for this purpose. We have seen in Sec. 4.4 that if the DKF learns the gain matrices on accurate measurements but with a surrogate measurement equation, the estimated trajectory is intrinsically poor. The same occurs in principle to other methods, and an evolution

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_14

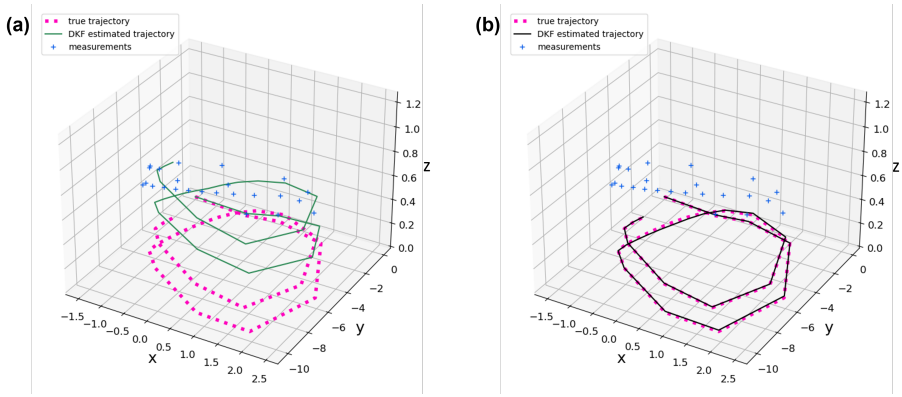


Fig. 7. Comparison between trajectory estimates obtained with the standard form of the DKF (a) and the evolved DKF (b) for an Hindmarsh–Rose nonlinear dynamics example. The magenta dotted line indicate the true trajectory, the blue cross the measurement points, the green and black solid lines are the the trajectory estimates respectively for the two different approaches. The evolved DKF provides a much more accurate estimate.

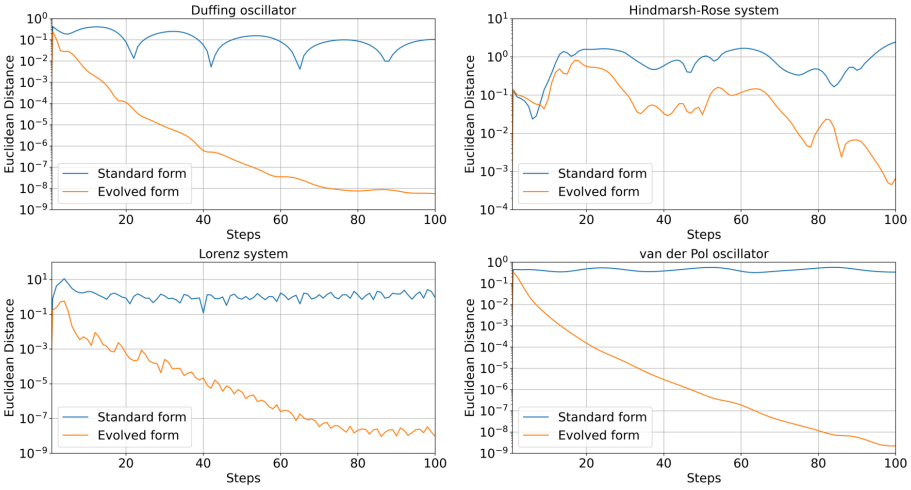


Fig. 8. Performance comparison between standard and evolved form of the DKF for four instances of nonlinear model dynamics.

analogous to that presented here does not appear to exist in the literature for other methods, as we know. What we can say is that the effort to modify the existing methods, be analytical like the (Extended) Kalman Filter, or based on the statistical ensemble like the EnKF or learning-enhanced like KalmanNet [4] would be not immediate and, probably, not trivial.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_14

Nonlinear model	Standard DKF 2-norm	Evolved DKF 2-norm
Duffing	10^{-1}	10^{-8}
Hindmarsh–Rose	10^{0}	10^{-3}
Lorenz	10^{0}	10^{-7}
van der Pol	10^{-1}	10^{-9}

Table 2. Performance comparison between standard and evolved form of the DKF for four instances of nonlinear model dynamics.

5 Conclusions

We are devising a strategy to tackle with surrogate measurements and/or measurement equations, in data assimilation procedures. The proposed method shows potentials in learning corrector gains, during over-instrumented laboratory experiments, that allow a field-operational surrogate measurement process to be accurate in state estimation.

Acknowledgments. This work was supported by the NATO Science and Technology Organization, NATO Office of the Chief Scientist, Bruxelles, Belgium.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Amendola, M., Arcucci, R., Mottet, L., Casas, C.Q., Fan, S., Pain, C., Linden, P., Guo, Y.K.: Data assimilation in the latent space of a neural network. In: arXiv preprint arXiv:2012.12056 (2020). https://doi.org/10.48550/arXiv.2012.12056
2. Arcucci, R., Moutiq, L., Guo, Y.K.: Neural assimilation. In: Krzhizhanovskaya, V.V., Závodszky, G., Lees, M.H., Dongarra, J.J., Sloot, e.M.A., Brissos, S., Teixeira, João", t.A. (eds.) Computational Science – ICCS 2020. pp. 155–168. Springer International Publishing, Cham (2020). https://doi.org/10.1007/978-3-030-50433-5_13
3. Bach, E., Baptista, R., Calvello, E., Chen, B., Stuart, A.: Learning enhanced ensemble filters. Journal of Computational Physics **547** (2026). https://doi.org/10.1016/j.jcp.2025.114550, cited by: 0
4. Buchnik, I., Revach, G., Steger, D., Van Sloun, R.J.G., Routtenberg, T., Shlezinger, N.: Latent-Kalmannet: Learned Kalman filtering for tracking from high-dimensional signals. IEEE Transactions on Signal Processing **72**, 352–367 (2024). https://doi.org/10.1109/TSP.2023.3344360
5. Cheng, S., et al.: Machine learning with data assimilation and uncertainty quantification for dynamical systems: A review. IEEE/CAA Journal of Automatica Sinica **10** (2023). https://doi.org/10.1109/JAS.2023.123537
6. Chinellato, E., Marcuzzi, F.: State, parameters and hidden dynamics estimation with the deep kalman filter: regularization strategies. Journal of Computational Science **87**, 102569 (2025). https://doi.org/https://doi.org/10.1007/978-3-031-63775-9

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_14

7. Farrell, J.: Aided Navigation: GPS with High Rate Sensors. McGraw-Hill professional engineering: Electronic engineering, McGraw Hill LLC (2008), https://books.google.it/books?id=yNujEvIMszYC
8. Groves, P.: Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems, Second Edition (03 2013)
9. Guo, H., Liu, X., Jin, X., Wang, G., Jiang, Y., Guo, J.: Gnss standard point positioning method based on spherical harmonic expansion of signal propagation path relating errors. Advances in Space Research **72**(4), 1153–1171 (2023). https://doi.org/https://doi.org/10.1016/j.asr.2023.04.003
10. Guo, Z.C., Forbes, J.R., Barfoot, T.D.: KILO-EKF: Koopman-inspired learned observations extended kalman filter. arXiv preprint arXiv:2601.12463 (2026). https://doi.org/10.48550/arXiv.2601.12463, submitted to IEEE Robotics and Automation Letters
11. Hindmarsh, J.L., Rose, R.M.: A model of neuronal bursting using three coupled first order differential equations. Proceedings of the Royal Society of London. B. Biological Sciences **221**(1222), 87–102 (03 1984). https://doi.org/10.1098/rspb.1984.0024
12. Levine, M.E., Stuart, A.M.: A framework for machine learning of model error in dynamical systems (2022)
13. LIANG, J., TERASAKI, K., MIYOSHI, T.: A machine learning approach to the observation operator for satellite radiance data assimilation. Journal of the Meteorological Society of Japan. Ser. II **101**(1), 79–95 (2023). https://doi.org/10.2151/jmsj.2023-005
14. Liu, W., Lai, Z., Bacsa, K., Chatzi, E.: Neural extended kalman filters for learning and predicting dynamics of structural systems. Structural Health Monitoring **23**(2), 1037–1052 (2024). https://doi.org/10.1177/14759217231179912
15. Mortada, H., Falcon, C., Kahil, Y., Clavaud, M., Michel, J.P.: Recursive kalmannet: Deep learning-augmented kalman filtering for state estimation with consistent uncertainty quantification. In: 2025 33rd European Signal Processing Conference (EUSIPCO). pp. 885–889 (2025). https://doi.org/10.23919/EUSIPCO63237.2025.11226444
16. Newman, T., Nemeth, C., Jones, M., Jonathan, P.: Deep learning surrogates for real-time gas emission inversion. arXiv preprint arXiv:2506.14597 (2025). https://doi.org/10.48550/arXiv.2506.14597
17. Titterton, D., Weston, J.: Strapdown Inertial Navigation Technology. The Institution of Engineering and Technology, 2nd edn. (2004). https://doi.org/10.1049/PBRA017E
18. Zhang, Z., Hou, M., Zhang, F., Edwards, C.R.: An lstm based kalman filter for spatio-temporal ocean currents assimilation. In: Proceedings of the 14th International Conference on Underwater Networks & Systems. WUWNet '19, Association for Computing Machinery, New York, NY, USA (2020). https://doi.org/10.1145/3366486.3366522

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_14