Journal of Computational Physics 511 (2024) 113118

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Sampling free iterative PCE filter for state and parameter estimation of nonlinear dynamical systems

W. van Dijk, W.B.J. Hakvoort, B. Rosić *

Department of Mechanics of Solids, Surfaces & Systems, University of Twente, PO Box 217, 7500 AE, Enschede, the Netherlands

ARTICLE INFO

Keywords:
Bayesian estimation
Kalman filter
Polynomial chaos expansion
Gauss Newton
Square root filter

ABSTRACT

We present a novel filter for state and parameter estimation in non-linear dynamical systems, based on a generalised Kalman filter formulation. To achieve a sampling-free implementation, polynomial chaos expansion (PCE) and a Galerkin projection method are utilized for the propagation of uncertainties through the system dynamics. The non-linear dynamics of the system are then linearised by a sequence of Gauss-Newton iterations in combination with linear Kalman updates. Additionally, we introduce a new square root implementation of the PCE-based filter. The proposed filter is evaluated on the Lorenz-63 and Lorenz-84 models for the task of simultaneous state and parameter estimation and is compared with two related approaches. Finally, the computational complexity of our square-root implementation is compared against two existing square root approaches.

1. Introduction

During the modelling of physical systems, one often has to deal with incomplete knowledge of the model states and parameters. Therefore, data assimilation techniques can be used to extract useful information from sensory data. These assimilation techniques have applications in many fields, such as robotics and geophysics, and have therefore been widely studied in the past [14,10].

In case of simultaneous Bayesian like state and parameter estimation under nonlinear system dynamics, the relation between the parameters and the measurements is often non-linear. This makes the estimation of the likelihood function difficult and computationally expensive. However, as often the interest lies not in estimating the full posterior but its mean, few attempts are made in the direct estimation of posterior statistics [14,21,34,33]. Such an approach is known as nonlinear Kalman-like filtering.

The classical Kalman filter (KF) [22] is an optimal-variance filter for linear systems, and therefore fails when non-linear dynamics are encountered. Instead, the extended Kalman filter (EKF) [21] is proposed with the aim of linearising the measurement operator by a first order Taylor series expansion. The EKF filter can be further improved by iterating over the linearisation points [43] or including higher-order terms in the Taylor series expansion. However, the previously mentioned filter is focusing on the linearisation aspect, and not on the proper approximation of the uncertainty. Therefore, the Ensemble Kalman filter (EnKF) [13] was introduced in which the uncertainties are represented by a sample of sufficient size. However, the EnKF is a Monte Carlo method and is therefore characterised by slow convergence. Related Monte-Carlo methods, such as the particle filter, suffer a similar drawback [36, Chapter 11].

* Corresponding author.
E-mail address: b.rosic@utwente.nl (B. Rosić).

https://doi.org/10.1016/j.jcp.2024.113118
Received 8 August 2023; Received in revised form 2 May 2024; Accepted 15 May 2024

Available online 22 May 2024

More recently, a popular functional approximation method, also known as polynomial chaos expansion (PCE), has been combined with data assimilation techniques [34,33,30,24,35]. This leads to efficient propagation of uncertainties through the systems dynamics, here referred to as the "forward problem". With PCE, all the uncertainties in the model equations are expanded using optimal basis polynomials. Subsequently, one seeks the coefficients of the PCE representations. After obtaining these coefficients, the uncertainties can be sampled fast and their statistical moments can be evaluated efficiently. Several methods for solving the forward problem using PCE are present from literature [12,40]. However, although important, the problem of selecting a suitable method for obtaining the expansion coefficients in the forward step is outside the scope of this paper. In this paper, the focus is on the assimilation step.

During the assimilation step, the expansion coefficients representing the unknowns can be updated directly, i.e. without sampling of the PCE approximations as shown in [33]. In contrast to this, PCE can be used for the forward propagation problem only, whereas the assimilation step is approximated by sampling, as demonstrated in [24,3]. [35] introduced a minimum mean square error estimator of the polynomial coefficients (PCKF), where the forward problem is solved using a stochastic Galerkin method and the update is derived using the Kalman filter theory. Similarly, [42] used the stochastic collocation method (SCKF) for both steps. [6] proposed a PCE-based EKF for state and parameter estimation of a mechanical system. They use a linearised observation operator that is available analytically and PCE for the forward problem allowing for non-linear forward propagation of the uncertainties. The filter generalization in which both the forward problem and the update step are solved by PCE is presented in [34,33,30]. Here, the authors proposed a filter based on the approximation of the conditional expectation.

However, as noted by several authors, the PCE based filter has a computational drawback. Each measurement is modelled by a new random variable, therefore increasing the number of basis polynomials at each assimilation step [30]. Eventually, the PCE based filters can become intractable in applications with large parameter/state spaces. The problem of the expanding basis can be simply ignored by assuming that the observation does not depend on a random variable associated with the measurement noise, as was done by [42] and [6]. This is in contrast with [9], where in context of the EnKF, it was shown that the observations must be treated as random variables to prevent underestimating the variance. [34,33,30] re-used the basis polynomials of the forward problem for the measurement to mitigate the growth in the PCE basis. On the other hand, [37] projected the error covariance on a PCE with reduced order allowing additional random variables without increasing the size of the basis functions. Another way to circumvent an expanding basis, is to avoid the creation of a measurement PCE in the first place by using a square root approach. This was proposed by [31] for the linear Bayesian update and later by [41] for the Kalman filter. In this paper, another square root approach is introduced to circumvent the growth in the PCE basis.

This paper presents a new sampling-free filter that combines a generalized Kalman filter formulation with PCE. We combine this sampling-free filter with Gauss-Newton iterations that linearise the non-linearities present in the problem. This filter is implemented using a square root approach based on the work of [5] to circumvent the growth in the basis. Finally, the presented filter is applied on joint state and parameter estimation of the Lorenz-63 system [26] and the square root approach is compare in terms of numerical complexity to two existing square root approaches.

The remaining part of the paper proceeds as follows: in Section 2 the state and parameter estimation problem is introduced. In Section 3 the proposed filter is derived starting from a generalized Kalman filter [33]. A numerical experiment and the comparison between two closely related filters are given in Section 4. Lastly, in Section 5 concluding remarks are made.

2. Problem setting

Consider the following non-linear initial value problem

$$\frac{\mathrm{d}\boldsymbol{x}(t,\boldsymbol{p})}{\mathrm{d}t} = f(\boldsymbol{x}(t),\boldsymbol{p}), \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0, \tag{1}$$

in which $\boldsymbol{x}(t,\boldsymbol{p})$ denotes the state-vector of the dynamical system, and $\boldsymbol{p}$ is a vector containing parameters describing system properties. This system is observed at discrete time moments t_k by:

$$\boldsymbol{y}_k = h(\boldsymbol{x}_k,\boldsymbol{p}), \tag{2}$$

where $h(\cdot,\cdot)$ denotes a possibly non-linear observation operator. Typically, the observations are corrupted by sensory noise such that

$$\boldsymbol{z}_k = h(\boldsymbol{x}_k^{true},\boldsymbol{p}) + \boldsymbol{\epsilon}_k \tag{3}$$

holds. Here, $\boldsymbol{x}_k^{true}$ denotes the true state and $\boldsymbol{\epsilon}_k$ is a realization of the sensory noise. For notational simplicity the time index is dropped in the notation. Thus, at the time instance t_k, the observation is $\boldsymbol{y} = h(\boldsymbol{x},\boldsymbol{p})$ and the measurement is $\boldsymbol{z} = h(\boldsymbol{x}^{true},\boldsymbol{p}) + \boldsymbol{\epsilon}$.

Typically, the states and/or parameters in Eq. (1) are unknown and are to be estimated given the measurement data in Eq. (3). As such an estimation is known to be ill-posed in a Hadamard sense, we restrain ourselves to the Bayesian approach [18] in which unknown parameters/states are described by prior expert's knowledge. The overarching goal is therefore to combine the prior available information on the states and parameters with the information obtained by measurements, and hence, to improve our knowledge about the uncertain states and/or parameters.

3. Generalized Kalman filter

In a Bayesian setting the model parameters and states are modelled a priori as random variables:

$$\boldsymbol{q}_f(\omega_f) := [\boldsymbol{x}_f(t,\omega_f), \boldsymbol{p}_f(\omega_f)],$$

belonging to a probability space $(\Omega_f, \mathcal{F}_f, \mathbb{P}_f)$ in which the index "f" denotes "the forecast". Here, Ω_f denotes the sample space, $\mathcal{F}_f$ denotes the σ-algebra, and $\mathbb{P}_f$ is a probability measure. Incorporating prior information in Eq. (1) results in a stochastic ordinary differential equation of the following form:

$$\begin{aligned}\frac{\mathrm{d}\boldsymbol{x}_f(t,\omega_f)}{\mathrm{d}t} &= f(\boldsymbol{x}_f(t,\omega_f), \boldsymbol{p}_f(\omega_f)),\\ &= f(\boldsymbol{q}_f(\omega_f)),\end{aligned} \tag{4}$$

with the initial condition being a random variable $\boldsymbol{x}_0(\omega_f)$. As a consequence of Eq. (4), the observation forecast becomes a random variable too,

$$\begin{aligned}\boldsymbol{y}_f(\omega_f) &:= h(\boldsymbol{x}_f(t,\omega_f), \boldsymbol{p}_f(\omega_f)))\\ &= h(\boldsymbol{q}_f(\omega_f)).\end{aligned} \tag{5}$$

Similarly, the sensory noise is modelled as a random variable, such that the measurement forecast is given by:

$$\boldsymbol{z}_f(\omega_f, \omega_e) = \boldsymbol{y}_f(\omega_f) + \boldsymbol{e}_f(\omega_e), \tag{6}$$

in which $\boldsymbol{e}_f(\omega_e)$ is the predicted measurement noise belonging to $(\Omega_e, \mathcal{F}_e, \mathbb{P}_e)$, and is typically described by a zero-mean Gaussian random variable with a covariance matrix $\boldsymbol{C}_{e_f}$, i.e. $\boldsymbol{e}_f(\omega_e) \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{C}_{e_f})$. Assuming independence between the measurement noise $\boldsymbol{e}_f$ and $\boldsymbol{q}_f$, the overall probability space is described by the triplet $(\Omega := \Omega_f \times \Omega_e, \mathcal{F} := \sigma(\mathcal{F}_f \times \mathcal{F}_e), \mathbb{P} := \mathbb{P}_f \mathbb{P}_e)$, and an elementary event ω. Thus, $\boldsymbol{q}_f(\omega_f)$ and $\boldsymbol{y}_f(\omega_f)$ are further denoted as $\boldsymbol{q}_f(\omega)$ and $\boldsymbol{y}_f(\omega)$ respectively, and $\boldsymbol{z}_f(\omega_f, \omega_e)$ as $\boldsymbol{z}_f(\omega)$.

To assimilate the prior knowledge with a measurement, a general Kalman filter derived in [34,33] is used:

$$\boldsymbol{q}_a(\omega) = \boldsymbol{q}_f(\omega) + \mathbb{E}[\boldsymbol{q}_f(\omega)|\boldsymbol{z}] - \mathbb{E}[\boldsymbol{q}_f(\omega)|\boldsymbol{z}_f(\omega))]. \tag{7}$$

Here, the posterior mean $\boldsymbol{q}_a(\omega)$ ("a" stands for "assimilated") linearly depends both on the prior knowledge $\boldsymbol{q}_f(\omega)$ and the innovation term, $\mathbb{E}[\boldsymbol{q}_f(\omega)|\boldsymbol{z}] - \mathbb{E}[\boldsymbol{q}_f(\omega)|\boldsymbol{z}_f(\omega))]$. $\mathbb{E}[\boldsymbol{q}_f(\omega)|\boldsymbol{z}]$ and $\mathbb{E}[\boldsymbol{q}_f(\omega)|\boldsymbol{z}_f(\omega)]$ denote the conditional expectations of $\boldsymbol{q}_f(\omega)$ given the measurement $\boldsymbol{z}$ or the predicted measurement $\boldsymbol{z}_f(\omega)$, respectively.

By the Doob-Dynkin lemma [8] the conditional expectation $\mathbb{E}[\boldsymbol{q}_f(\omega)|\cdot]$ can be approximated by a map $\varphi_{\boldsymbol{q}}(\cdot)$ parametrized by $\boldsymbol{\beta}$, which can be estimated given the following optimality condition:

$$\boldsymbol{\beta}^* = \arg\min_{\boldsymbol{\beta}} \mathbb{E}\left[\left\|\boldsymbol{q}_f(\omega) - \varphi_{\boldsymbol{q}}(\boldsymbol{z}_f(\omega), \boldsymbol{\beta})\right\|^2\right], \tag{8}$$

as shown in [33]. This further leads to

$$\boldsymbol{q}_a(\omega) = \boldsymbol{q}_f(\omega) + \varphi_{\boldsymbol{q}}(\boldsymbol{z}, \boldsymbol{\beta}^*) - \varphi_{\boldsymbol{q}}(\boldsymbol{z}_f(\omega), \boldsymbol{\beta}^*), \tag{9}$$

in which, based on the problem at hand one can pick a suitable model structure for the map $\varphi_{\boldsymbol{q}}$.

3.1. Gauss-Newton iterations

The previously suggested filter represents the update of the prior random variable via innovation term into the posterior random variable. In the previous derivations we did not assume the type of measure used to describe the prior knowledge. We have only assumed the type of map used to approximate the observation-quantity of interest relation. In this aspect there are no constraints on the definition of the prior distribution. In the subsequent chapter we will show that by discretizing the random variable in a functional approximation manner one would then obtain a possibility to compute more than the first two moments of the posterior in contrast to the classical Kalman filter.

In this work, under an appropriate differentiability assumption we use Gauss-Newton iterations [4] to approximate the non-linearities present in Eq. (9). The measurement forecast is linearised around the point $\hat{\boldsymbol{q}}$ such that,

$$\boldsymbol{y}_f(\omega) \approx \boldsymbol{s}_l(\omega) := \hat{\boldsymbol{H}}\left(\boldsymbol{q}_f(\omega) - \hat{\boldsymbol{q}}\right) + \hat{\boldsymbol{h}} \tag{10}$$

holds. Once the measurement forecast is linearized, one may use the linear Kalman formula as shown in [34,33]:

$$\boldsymbol{q}_a(\omega) = \boldsymbol{q}_f(\omega) + \boldsymbol{K}(\boldsymbol{z} - \boldsymbol{s}_l(\omega) - \boldsymbol{e}_f(\omega)) \tag{11}$$

where

$$\boldsymbol{K} = \boldsymbol{C}_{\boldsymbol{q}_f \boldsymbol{s}_l}(\boldsymbol{C}_{\boldsymbol{s}_l} + \boldsymbol{C}_{\boldsymbol{e}_f})^{-1}, \tag{12}$$

denotes the generalized Kalman gain, and $\boldsymbol{C}_{\boldsymbol{q}_f \boldsymbol{s}_l}$, $\boldsymbol{C}_{\boldsymbol{s}_l}$ and $\boldsymbol{C}_{\boldsymbol{e}_f}$ denote the (cross-)covariance matrices of $\boldsymbol{q}_f(\omega)$, $\boldsymbol{s}_l(\omega)$ and $\boldsymbol{e}_f(\omega)$ respectively. Equation (11) reduces to the well-known Kalman equation when assuming all random variables to be Gaussian. However, the latter assumption is not present in this work, meaning that Eq. (11) can also be used in a non-Gaussian case.

The linearisation in Eq. (10) is however greatly dependent on the linearisation point $\hat{\boldsymbol{q}}$ and its position determines to a large extent the performance of the linear update. In comparable linearisation approaches, the initial linearisation point is sub-optimally selected as $\mathbb{E}\left[\boldsymbol{q}_f(\omega)\right]$, leading to poor and biased approximations of the observation operator. Therefore, in this paper the approximation is improved by using a sequence of Gauss-Newton iterations, where the linearisation point is improved between iterations. The converged solution is then used as the assimilated posterior mean. In other words, in the ith iteration, one linearises the measurement forecast around the point $\hat{\boldsymbol{q}}^{(i)}$ as:

$$\boldsymbol{s}_l^{(i)}(\omega) = \hat{\boldsymbol{H}}^{(i)}(\boldsymbol{q}_f(\omega) - \hat{\boldsymbol{q}}^{(i)}) + \hat{\boldsymbol{h}}^{(i)}, \tag{13}$$

and thus Eq. (11) is substituted by the sequence of updates:

$$\begin{aligned} \boldsymbol{q}_a^{(i+1)}(\omega) &= \boldsymbol{q}_f(\omega) + \boldsymbol{K}^{(i)}\left(\boldsymbol{z} - \boldsymbol{s}_l^{(i)}(\omega) - \boldsymbol{e}_f(\omega)\right), \\ \boldsymbol{q}_a^{(0)}(\omega) &= \boldsymbol{q}_f(\omega). \end{aligned} \tag{14}$$

Here, $\boldsymbol{K}^{(i)}$ is calculated using

$$\boldsymbol{K}^{(i)} = \boldsymbol{C}_{\boldsymbol{q}_f \boldsymbol{s}_l^{(i)}}(\boldsymbol{C}_{\boldsymbol{s}_l^{(i)}} + \boldsymbol{C}_{\boldsymbol{e}})^{-1}, \tag{15}$$

and the linearisation point is updated as

$$\hat{\boldsymbol{q}}^{(i+1)} = \mathbb{E}\left[\boldsymbol{q}_a^{(i+1)}(\omega)\right]. \tag{16}$$

The suboptimal linearisation point is still used as an initial linearisation point, i.e. $\hat{\boldsymbol{q}}^{(i+1)} = \mathbb{E}\left[\boldsymbol{q}_a^{(i+1)}(\omega)\right]$.

The linear map in Eq. (13) can be estimated by estimating the Jacobian, or can be found by minimizing the following optimality criteria [32]:

$$\underset{\hat{\boldsymbol{H}}^{(i)}, \hat{\boldsymbol{h}}^{(i)}}{\arg\min} \left\| \boldsymbol{y}_l^{(i)}(\omega) - \left(\hat{\boldsymbol{H}}^{(i)}(\boldsymbol{q}_a^{(i)}(\omega) - \hat{\boldsymbol{q}}^{(i)}) + \hat{\boldsymbol{h}}^{(i)}\right)\right\|^2, \tag{17}$$

with an optimum defined as

$$\hat{\boldsymbol{H}}^{(i)} = \boldsymbol{C}_{\boldsymbol{y}_l^{(i)} \boldsymbol{q}_a^{(i)}} \boldsymbol{C}^{-1}_{\boldsymbol{q}_a^{(i)}}, \tag{18}$$

$$\hat{\boldsymbol{h}}^{(i)} = \mathbb{E}\left[\boldsymbol{y}_l^{(i)}(\omega)\right] - \hat{\boldsymbol{H}}^{(i)}(\mathbb{E}\left[\boldsymbol{q}_a^{(i)}(\omega)\right] - \hat{\boldsymbol{q}}^{(i)}), \tag{19}$$

given $\boldsymbol{y}_l^{(i)}(\omega) = h(\boldsymbol{q}_a^{(i)}(\omega))$.

The update in Eq. (14) is repeated until a maximum number of iterations i_{max} is reached, or until the shift in linearisation point between iterations is small, i.e. until $\gamma < \gamma_{tol}$, where

$$\gamma = \frac{\left\| \hat{\boldsymbol{q}}_l^{(i+1)} - \hat{\boldsymbol{q}}_l^{(i)} \right\|_2}{\left\| \hat{\boldsymbol{q}}_l^{(i)} \right\|_2}, \tag{20}$$

and γ_{tol} is chosen accordingly. The convergence properties of the algorithm can be studied via fixed point theorem [19], according to which the algorithm has local convergence characterised by a spectral radius of $\rho(\boldsymbol{K}^{(i)}\hat{\boldsymbol{H}}^{(i)})$. Up to now, the random variables present in Eq. (11) are not discretized. In the next section, these random variables are discretized using a functional expansion method.

3.2. Discretization by polynomial chaos expansion

In this paper, the random variables are discretized using Wiener-Askey PCE [39]. The PCE of the random variable $\boldsymbol{q}_f(\omega)$ reads:

$$\begin{aligned} \boldsymbol{q}_f(\omega) &= \sum_{\boldsymbol{\alpha} \in \mathcal{J}} \boldsymbol{q}_f^{(\boldsymbol{\alpha})} \boldsymbol{\Phi}_{\boldsymbol{\alpha}}(\xi_1(\omega), \dots, \xi_N(\omega)) \\ &= \sum_{\boldsymbol{\alpha} \in \mathcal{J}} \boldsymbol{q}_f^{(\boldsymbol{\alpha})} \boldsymbol{\Phi}_{\boldsymbol{\alpha}}(\boldsymbol{\xi}), \end{aligned} \tag{21}$$

where $\mathcal{J}$ denotes a multi-index set (see Appendix A), $\boldsymbol{\Phi}_{\boldsymbol{\alpha}}(\boldsymbol{\varepsilon})$ denotes the multivariate basis polynomials in the random variables $\boldsymbol{\xi}$, and $\boldsymbol{q}_f^{(\boldsymbol{\alpha})} := [q_{f,1}^{(\boldsymbol{\alpha})}, \dots, q_{f,N}^{(\boldsymbol{\alpha})}]^\intercal$ denotes the expansion coefficients of the random variables, where N denotes the number of random variables. For practical implementation, it is necessary to limit the number of basis polynomials. The expansion then only approximates the random variables as

$$\boldsymbol{q}_f(\omega) \approx \sum_{\boldsymbol{\alpha}\in\mathcal{J}_t} \boldsymbol{q}_f^{(\boldsymbol{\alpha})} \boldsymbol{\Phi}_{\boldsymbol{\alpha}}(\boldsymbol{\xi}), \tag{22}$$

where $\mathcal{J}_t \subset \mathcal{J}$ denotes the truncated multi-index set defined by the polynomial order d. This maximum order is imposed on all basis polynomials, although other adaptive schemes exist [7]. For this scheme, the number of basis polynomials used in the expansion, M, depends on the number of random variables N and the maximum polynomial order d. This reveals the problem of modelling of the measurement noise $\boldsymbol{e}_f$ (see Eq. (11)) by independent random variables, since each additional random variable adds terms to the stochastic basis.

It is useful to split the expansion coefficients into parts representing the mean and the fluctuating component of the expansion. The approximated mean is equal to the first expansion coefficient, $\bar{\boldsymbol{q}} = \boldsymbol{q}^{(\boldsymbol{\alpha}_0)}$, and the fluctuating part is represented by all other coefficients. The multi-index set for the fluctuating part of the approximation is given by $\mathcal{J}_{\neq 0}$, which denotes the truncated multi-index set excluding the first index. Collecting the coefficients of the fluctuating part in a matrix:

$$\tilde{\boldsymbol{Q}}_f = [\boldsymbol{q}_f^{(\alpha_1)}, \boldsymbol{q}_f^{(\alpha_2)}, \dots, \boldsymbol{q}_f^{(\alpha_{M-1})}]. \tag{23}$$

Following this, one may write:

$$\boldsymbol{Q}_f := [\bar{\boldsymbol{q}}_f, \tilde{\boldsymbol{Q}}_f], \tag{24}$$

such that $\boldsymbol{Q}$ contains all the expansion coefficients. Using the coefficients of the fluctuating part, the covariance matrices required to evaluate Eq. (14) can now be calculated efficiently by:

$$\begin{aligned} \boldsymbol{C}_{\boldsymbol{q}_f \boldsymbol{s}_l^{(i)}} &= \tilde{\boldsymbol{Q}}_f \boldsymbol{\Delta} (\tilde{\boldsymbol{S}}_l^{(i)})^\intercal, \\ \boldsymbol{C}_{\boldsymbol{s}_l^{(i)}} &= \tilde{\boldsymbol{S}}_l^{(i)} \boldsymbol{\Delta} (\tilde{\boldsymbol{S}}_l^{(i)})^\intercal, \end{aligned} \tag{25}$$

where $\boldsymbol{\Delta} = \mathrm{diag}(\langle \boldsymbol{\Phi}_\alpha(\boldsymbol{\xi})^2 \rangle)$ for $\boldsymbol{\alpha} \in \mathcal{J}_{\neq 0}$. The computational cost of the repeated evaluation of the covariance matrices required for the Kalman gain in Eq. (15) scales with the number of expansion coefficients M. For sampling based implementations, this scales with the size of the ensemble. Hence, if the number of expansion coefficients is smaller than the number of samples required for accurate sampling based implementations, an improvement in terms of the computational cost is made. For high-dimensional problems one might use more advanced versions of PCE that incorporate low-rank and sparse approximations [28], such that the number of expansion coefficients M is reduced. The iterative linearisation, found by Eqs. (18) and (19) in terms of PCE coefficients are (see Appendix B):

$$\begin{aligned} \hat{\boldsymbol{H}}^{(i)} &= \tilde{\boldsymbol{Y}}_l^{(i)} (\tilde{\boldsymbol{Q}}_a^{(i)})^{-1} \\ \hat{\boldsymbol{h}}^{(i)} &= \bar{\boldsymbol{y}}_l^{(i)} - \hat{\boldsymbol{H}}^{(i)} (\bar{\boldsymbol{q}}_a^{(i)} - \hat{\boldsymbol{q}}^{(i)}). \end{aligned} \tag{26}$$

By expansion of all random variables in Eq. (14) and Galerkin projection onto the space spanned by the stochastic basis [34], one obtains:

$$(\boldsymbol{q}_l^{(\boldsymbol{\alpha})})^{(i+1)} = \boldsymbol{q}_f^{(\boldsymbol{\alpha})} + \boldsymbol{K}^{(i)} \left(\boldsymbol{z}^{(\boldsymbol{\alpha})} - (\boldsymbol{s}_l^{(\boldsymbol{\alpha})})^{(i)} - \boldsymbol{e}_f^{(\boldsymbol{\alpha})} \right), \qquad \boldsymbol{\alpha} \in \mathcal{J}_t. \tag{27}$$

The previous equation can be split into a mean and fluctuating part in matrix form:

$$\bar{\boldsymbol{q}}_l^{(i+1)} = \bar{\boldsymbol{q}}_f + \boldsymbol{K}^{(i)} \left(\boldsymbol{Z} - \boldsymbol{S}_l^{(i)} - \boldsymbol{E}_f \right), \tag{28}$$

$$\boldsymbol{Q}_l^{(i+1)} = \boldsymbol{Q}_f + \boldsymbol{K}^{(i)} \left(\boldsymbol{Z} - \boldsymbol{S}_l^{(i)} - \boldsymbol{E}_f \right), \tag{29}$$

respectively. Here, $\boldsymbol{Q}_l^{(i+1)}$, $\boldsymbol{S}_l^{(i)}$, and $\boldsymbol{E}_f$ denote the collection of expansion coefficients of $\boldsymbol{q}_l^{(i)}$, $\boldsymbol{s}_l^{(i)}$, and $\boldsymbol{e}_f$, respectively, constructed in the same way as Eq. (23) and Eq. (24). Note that since the measurement $\boldsymbol{z}$ is deterministic, only the first expansion coefficient is non-zero, i.e. $\boldsymbol{Z} = [\boldsymbol{z}, \boldsymbol{0}]$. Furthermore, a PCE of the random variable $\boldsymbol{e}_f(\omega)$ is present in Eq. (27). This random variable is problematic as it increases the size of the basis, since for each measurement, a new random independent variable must be introduced.

3.2.1. Forward propagation

In this work, a stochastic Galerkin method is used to propagate the uncertainties forward in time. Note that other methods for forward propagation are valid options as well, however, in this work a sampling free approach is preferred. In the stochastic Galerkin framework, the random variables in Eq. (4) are substituted with their PCE approximations and the resulting residual equation is projected onto the space spanned by the stochastic basis. The result is an augmented ODE, in which the random variables are replaced by the expansion coefficients:

$$\frac{\mathrm{d}\boldsymbol{x}^{(\boldsymbol{\alpha})}(t)}{\mathrm{d}t} = \breve{f}(\boldsymbol{x}^{(\boldsymbol{\alpha})}(t), \boldsymbol{p}^{(\alpha)}), \qquad \boldsymbol{x}^{(\boldsymbol{\alpha})}(0) = \boldsymbol{x}_0^{(\boldsymbol{\alpha})}, \tag{30}$$

where $\breve{f}(\cdot)$ denotes the for PCE augmented version of the function $f(\cdot)$ in Eq. (4). Similarly, an augmented observation function $\breve{h}(\cdot)$ can be derived,

$$\boldsymbol{y}_{f,k}^{(\boldsymbol{\alpha})} = \breve{h}(\boldsymbol{q}_{f,k}^{(\boldsymbol{\alpha})}). \tag{31}$$

The derivation of such an augmented function for a case-study used in this paper is shown in Appendix C. Other examples can be found in [39,11]. The augmented ODE can be integrated in time using existing time-integration schemes.

3.3. Square root implementation

As noted before, every measurement introduces an additional independent random variable therefore increasing the size of the polynomial basis. To prevent the creation of a PCE for the predicted measurement noise, [30] used square root implementations of the linear Bayesian update (SRPCU) based on earlier work of [15]. Similarly, [41] introduced the polynomial chaos based square root Kalman filter (PCSKF) based on the earlier work of [38]. In this work a similar approach is used, but the square root implementation is based on the Ensemble Transform Kalman filter (ETKF) introduced by [5]. Our implementation has a benefit in terms of computational complexity when a diagonal covariance structure of the measurement noise is assumed and the number of sensors (N_y) is larger than M, i.e. when $M < N_y$. We refer to our newly proposed implementation as the Polynomial Chaos Transform Update (PCTU) and in combination with Gauss-Newton updates as GN-PCTU.

Consider a single iteration of Eq. (29), so that for ease of notation, one can drop the superscript i:

$$\boldsymbol{Q}_a = \boldsymbol{Q}_f + \boldsymbol{K}\left(\boldsymbol{Z} - \boldsymbol{S}_l - \boldsymbol{E}_f\right). \tag{32}$$

The previous equation then results in the update for the mean:

$$\bar{\boldsymbol{q}}_a = \bar{\boldsymbol{q}}_f + \boldsymbol{K}(\boldsymbol{z} - \bar{\boldsymbol{s}}_l). \tag{33}$$

To obtain the update for the covariance, one may note that the covariance can be decomposed into its square root:

$$\begin{aligned} \boldsymbol{C}_{q_f} &= \tilde{\boldsymbol{Q}}_f \boldsymbol{\Delta} \tilde{\boldsymbol{Q}}_f^\intercal \\ &= \left(\tilde{\boldsymbol{Q}}_f \sqrt{\boldsymbol{\Delta}}\right)\left(\tilde{\boldsymbol{Q}}_f \sqrt{\boldsymbol{\Delta}}\right)^\intercal \\ &= \hat{\boldsymbol{Q}}_f \hat{\boldsymbol{Q}}_f^\intercal \end{aligned} \tag{34}$$

where $\hat{\boldsymbol{Q}}_f$ denotes the scaled PCE coefficients, i.e. $\hat{\boldsymbol{Q}}_f = \tilde{\boldsymbol{Q}}_f \sqrt{\boldsymbol{\Delta}}$. Similarly, $\hat{\boldsymbol{S}}_l$ and $\hat{\boldsymbol{E}}_l$ denote $\hat{\boldsymbol{S}}_l = \tilde{\boldsymbol{S}}_l \sqrt{\boldsymbol{\Delta}}$ and $\hat{\boldsymbol{E}}_f = \tilde{\boldsymbol{E}}_f \sqrt{\boldsymbol{\Delta}}$, respectively. Note that since $\boldsymbol{\Delta}$ is diagonal, its square root, $\sqrt{\boldsymbol{\Delta}}$, is efficient to compute.

The square root representation of the covariance matrix in Eq. (34) is not unique as one may use any orthonormal matrix $\boldsymbol{\Gamma}$ such that

$$\boldsymbol{C}_{q_f} = \hat{\boldsymbol{Q}}_f \boldsymbol{\Gamma}\boldsymbol{\Gamma}^\intercal \hat{\boldsymbol{Q}}_f^\intercal = \hat{\boldsymbol{Q}}_f \hat{\boldsymbol{Q}}_f^\intercal \tag{35}$$

holds. Thus, to find the posterior square root one seeks the transformation matrix $\boldsymbol{T}$ that transforms the square root representation of the prior covariance matrix into the posterior one, i.e.

$$\hat{\boldsymbol{Q}}_a = \hat{\boldsymbol{Q}}_f \boldsymbol{T}\boldsymbol{\Gamma}. \tag{36}$$

Following Eq. (32), the expansion coefficients of the fluctuating part are updated by:

$$\begin{aligned} \tilde{\boldsymbol{Q}}_a &= \tilde{\boldsymbol{Q}}_f + \boldsymbol{K}\left(\boldsymbol{0} - \tilde{\boldsymbol{S}}_l - \tilde{\boldsymbol{E}}_f\right) \\ &= \tilde{\boldsymbol{Q}}_f - \boldsymbol{K}\left(\tilde{\boldsymbol{S}}_l + \tilde{\boldsymbol{E}}_f\right), \end{aligned} \tag{37}$$

resulting in the following posterior covariance:

$$\begin{aligned} \boldsymbol{C}_{q_a} &= \tilde{\boldsymbol{Q}}_a \boldsymbol{\Delta} \tilde{\boldsymbol{Q}}_a^\intercal \\ &= \left[\tilde{\boldsymbol{Q}}_f - \boldsymbol{K}(\tilde{\boldsymbol{S}}_l + \tilde{\boldsymbol{E}}_f)\right] \boldsymbol{\Delta} \left[\tilde{\boldsymbol{Q}}_f - \boldsymbol{K}(\tilde{\boldsymbol{S}}_l + \tilde{\boldsymbol{E}}_f)\right]^\intercal \\ &= \tilde{\boldsymbol{Q}}_f \boldsymbol{\Delta} \tilde{\boldsymbol{Q}}_f^\intercal - \tilde{\boldsymbol{Q}}_f \boldsymbol{\Delta}(\tilde{\boldsymbol{S}}_l + \tilde{\boldsymbol{E}}_f)^\intercal \boldsymbol{K}^\intercal - \boldsymbol{K}(\tilde{\boldsymbol{S}}_l + \tilde{\boldsymbol{E}}_f)\boldsymbol{\Delta} \tilde{\boldsymbol{Q}}_f^\intercal + \\ &\quad \boldsymbol{K}(\tilde{\boldsymbol{S}}_l + \tilde{\boldsymbol{E}}_f)\boldsymbol{\Delta}(\tilde{\boldsymbol{S}}_l + \tilde{\boldsymbol{E}}_f)^\intercal \boldsymbol{K}^\intercal \\ &= \tilde{\boldsymbol{Q}}_f \sqrt{\boldsymbol{\Delta}}\sqrt{\boldsymbol{\Delta}} \tilde{\boldsymbol{Q}}_f^\intercal - \tilde{\boldsymbol{Q}}_f \sqrt{\boldsymbol{\Delta}}\sqrt{\boldsymbol{\Delta}}(\tilde{\boldsymbol{S}}_l + \tilde{\boldsymbol{E}}_f)^\intercal \boldsymbol{K}^\intercal - \boldsymbol{K}(\tilde{\boldsymbol{S}}_l + \tilde{\boldsymbol{E}}_f)\sqrt{\boldsymbol{\Delta}}\sqrt{\boldsymbol{\Delta}} \tilde{\boldsymbol{Q}}_f^\intercal + \\ &\quad \boldsymbol{K}(\tilde{\boldsymbol{S}}_l + \tilde{\boldsymbol{E}}_f)\sqrt{\boldsymbol{\Delta}}\sqrt{\boldsymbol{\Delta}}(\tilde{\boldsymbol{S}}_l + \tilde{\boldsymbol{E}}_f)^\intercal \boldsymbol{K}^\intercal. \end{aligned} \tag{38}$$

In terms of the scaled expansion coefficients one may further write:

$$\begin{aligned} \hat{\boldsymbol{Q}}_a \hat{\boldsymbol{Q}}_a^\intercal &= \left[\hat{\boldsymbol{Q}}_f - \boldsymbol{K}(\hat{\boldsymbol{S}}_l + \hat{\boldsymbol{E}}_f)\right] \left[\hat{\boldsymbol{Q}}_f - \boldsymbol{K}(\hat{\boldsymbol{S}}_l + \hat{\boldsymbol{E}}_f)\right]^\intercal \\ &= \hat{\boldsymbol{Q}}_f \hat{\boldsymbol{Q}}_f^\intercal - \hat{\boldsymbol{Q}}_f(\hat{\boldsymbol{S}}_l + \hat{\boldsymbol{E}}_f)^\intercal \boldsymbol{K}^\intercal - \boldsymbol{K}(\hat{\boldsymbol{S}}_l + \hat{\boldsymbol{E}}_f)\hat{\boldsymbol{Q}}_f^\intercal + \boldsymbol{K}(\hat{\boldsymbol{S}}_l + \hat{\boldsymbol{E}}_f)(\hat{\boldsymbol{S}}_l + \hat{\boldsymbol{E}}_f)^\intercal \boldsymbol{K}^\intercal, \end{aligned} \tag{39}$$

where the generalized Kalman gain $\boldsymbol{K}$ in terms of the scaled expansion coefficients is given by

$$\begin{aligned} \boldsymbol{K} &= \boldsymbol{C}_{\boldsymbol{q}_f \boldsymbol{s}_l}(\boldsymbol{C}_{\boldsymbol{s}_l} + \boldsymbol{C}_{\boldsymbol{e}_f})^{-1}, \\ &= \hat{\boldsymbol{Q}}_f \hat{\boldsymbol{S}}_l^{\intercal}(\boldsymbol{C}_{\boldsymbol{s}_l} + \boldsymbol{C}_{\boldsymbol{e}_f})^{-1}. \end{aligned} \tag{40}$$

By assuming that the sensory noise is independent from the prior random variables and the observation forecast, i.e. by assuming

$$\boldsymbol{C}_{\boldsymbol{q}_f \boldsymbol{e}_f} = \boldsymbol{C}_{\boldsymbol{e}_f \boldsymbol{q}_f}^{\intercal} = \hat{\boldsymbol{Q}}_f \hat{\boldsymbol{E}}_f^{\intercal} = \boldsymbol{0}, \qquad \boldsymbol{C}_{\boldsymbol{s}_l \boldsymbol{e}_f} = \boldsymbol{C}_{\boldsymbol{e}_f \boldsymbol{s}_l}^{\intercal} = \hat{\boldsymbol{S}}_l \hat{\boldsymbol{E}}_f^{\intercal} = \boldsymbol{0}, \tag{41}$$

Eq. (39) simplifies to:

$$\hat{\boldsymbol{Q}}_a \hat{\boldsymbol{Q}}_a^{\intercal} = \hat{\boldsymbol{Q}}_f \hat{\boldsymbol{Q}}_f^{\intercal} - \hat{\boldsymbol{Q}}_f \hat{\boldsymbol{S}}_l^{\intercal} \boldsymbol{K}^{\intercal} - \boldsymbol{K} \hat{\boldsymbol{S}}_l \hat{\boldsymbol{Q}}_f^{\intercal} + \boldsymbol{K}(\boldsymbol{C}_{\boldsymbol{s}_l} + \boldsymbol{C}_{\boldsymbol{e}_f})\boldsymbol{K}^{\intercal}. \tag{42}$$

Substitution of Eq. (40) in Eq. (42) then yields:

$$\begin{aligned} \hat{\boldsymbol{Q}}_a \hat{\boldsymbol{Q}}_a^{\intercal} = \hat{\boldsymbol{Q}}_f \hat{\boldsymbol{Q}}_f^{\intercal} &- \hat{\boldsymbol{Q}}_f \hat{\boldsymbol{S}}_l^{\intercal}(\boldsymbol{C}_{\boldsymbol{s}_l} + \boldsymbol{C}_{\boldsymbol{e}_f})^{-\intercal} \hat{\boldsymbol{S}}_l \hat{\boldsymbol{Q}}_f^{\intercal} - \hat{\boldsymbol{Q}}_f \hat{\boldsymbol{S}}_l^{\intercal}(\boldsymbol{C}_{\boldsymbol{s}_l} + \boldsymbol{C}_{\boldsymbol{e}_f})^{-1} \hat{\boldsymbol{S}}_l \hat{\boldsymbol{Q}}_f^{\intercal} \\ &+ \hat{\boldsymbol{Q}}_f \hat{\boldsymbol{S}}_l^{\intercal}(\boldsymbol{C}_{\boldsymbol{s}_l} + \boldsymbol{C}_{\boldsymbol{e}_f})^{-1}(\boldsymbol{C}_{\boldsymbol{s}_l} + \boldsymbol{C}_{\boldsymbol{e}_f})(\boldsymbol{C}_{\boldsymbol{s}_l} + \boldsymbol{C}_{\boldsymbol{e}_f})^{-\intercal} \hat{\boldsymbol{S}}_l \hat{\boldsymbol{Q}}_f^{\intercal}, \end{aligned} \tag{43}$$

where after cancellation of the second and last term in the right-hand side one obtains:

$$\begin{aligned} \hat{\boldsymbol{Q}}_a \hat{\boldsymbol{Q}}_a^{\intercal} &= \hat{\boldsymbol{Q}}_f \hat{\boldsymbol{Q}}_f^{\intercal} - \hat{\boldsymbol{Q}}_f \hat{\boldsymbol{S}}_l^{\intercal}(\boldsymbol{C}_{\boldsymbol{s}_l} + \boldsymbol{C}_{\boldsymbol{e}_f})^{-1} \hat{\boldsymbol{S}}_l \hat{\boldsymbol{Q}}_f^{\intercal} \\ &= \hat{\boldsymbol{Q}}_f \left[\boldsymbol{I} - \hat{\boldsymbol{S}}_l^{\intercal}(\boldsymbol{C}_{\boldsymbol{s}_l} + \boldsymbol{C}_{\boldsymbol{e}_f})^{-1} \hat{\boldsymbol{S}}_l \right] \hat{\boldsymbol{Q}}_f^{\intercal}. \end{aligned} \tag{44}$$

Using Eq. (36) one may write:

$$\begin{aligned} \hat{\boldsymbol{Q}}_a \hat{\boldsymbol{Q}}_a^{\intercal} &= \hat{\boldsymbol{Q}}_f \boldsymbol{T} \boldsymbol{\Gamma} (\hat{\boldsymbol{Q}}_f \boldsymbol{T} \boldsymbol{\Gamma})^{\intercal} \\ &= \hat{\boldsymbol{Q}}_f \boldsymbol{T} \boldsymbol{T}^{\intercal} \hat{\boldsymbol{Q}}_f^{\intercal}, \end{aligned} \tag{45}$$

such that its clear that a transformation matrix $\boldsymbol{T}$ that satisfies

$$\boldsymbol{T} \boldsymbol{T}^{\intercal} = \left[\boldsymbol{I} - \hat{\boldsymbol{S}}_l^{\intercal}(\boldsymbol{C}_{\boldsymbol{s}_l} + \boldsymbol{C}_{\boldsymbol{e}_f})^{-1} \hat{\boldsymbol{S}}_l \right], \tag{46}$$

would suffice. Using the Sherman-Morrison-Woodbury matrix identity [20], one may note:

$$\begin{aligned} (\boldsymbol{I} + \hat{\boldsymbol{S}}_l^{\intercal} \boldsymbol{C}_{\boldsymbol{e}_f}^{-1} \hat{\boldsymbol{S}}_l)^{-1} &= \boldsymbol{I} - \hat{\boldsymbol{S}}_l^{\intercal}(\hat{\boldsymbol{S}}_l \hat{\boldsymbol{S}}_l^{\intercal} + \boldsymbol{C}_{\boldsymbol{e}_f})^{-1} \hat{\boldsymbol{S}}_l \\ &= \boldsymbol{I} - \hat{\boldsymbol{S}}_l^{\intercal}(\boldsymbol{C}_{\boldsymbol{s}_l} + \boldsymbol{C}_{\boldsymbol{e}_f})^{-1} \hat{\boldsymbol{S}}_l, \end{aligned} \tag{47}$$

such that Eq. (46) becomes

$$\boldsymbol{T} \boldsymbol{T}^{\intercal} = \left[\boldsymbol{I} + \hat{\boldsymbol{S}}_l^{\intercal} \boldsymbol{C}_{\boldsymbol{e}_f}^{-1} \hat{\boldsymbol{S}}_l \right]^{-1}. \tag{48}$$

Introducing the eigenvalue decomposition $\boldsymbol{U} \boldsymbol{\Lambda} \boldsymbol{U}^{\intercal} = \hat{\boldsymbol{S}}_l^{\intercal} \boldsymbol{C}_{\boldsymbol{e}_f}^{-1} \hat{\boldsymbol{S}}_l$, results in

$$\begin{aligned} \boldsymbol{T} \boldsymbol{T}^{\intercal} &= (\boldsymbol{I} + \hat{\boldsymbol{S}}_l^{\intercal} \boldsymbol{C}_{\boldsymbol{e}_f}^{-1} \hat{\boldsymbol{S}}_l)^{-1} \\ &= (\boldsymbol{I} + \boldsymbol{U} \boldsymbol{\Lambda} \boldsymbol{U}^{\intercal})^{-1} \\ &= \boldsymbol{U}(\boldsymbol{I} + \boldsymbol{\Lambda})^{-1} \boldsymbol{U}^{\intercal}, \end{aligned} \tag{49}$$

where $\boldsymbol{U}$ and $\boldsymbol{\Lambda}$ denote the matrices of eigenvectors and eigenvalues, respectively. Following Eq. (49), the transformation matrix is obtained as follows:

$$\boldsymbol{T} = \boldsymbol{U}(\boldsymbol{I} + \boldsymbol{\Lambda})^{-1/2}. \tag{50}$$

The unitary matrix $\boldsymbol{U}$ rotates the prior scaled PCE coefficients to the vector space of the measurement forecast scaled with the measurement noise. In the aligned space, the covariance reduces with $(\boldsymbol{I} + \boldsymbol{\Lambda})^{-1/2}$ where $\boldsymbol{\Lambda}$ scales the eigenvalues of the forecast covariance with the eigenvalues of the noise covariance. Subsequently, the result is rotated back into the space of the scaled PCE coefficients by setting $\boldsymbol{\Gamma} = \boldsymbol{U}^{\intercal}$. The post-multiplication with the unitary matrix $\boldsymbol{U}^{\intercal}$ ensures $\boldsymbol{T}\boldsymbol{\Gamma} = \boldsymbol{U}(\boldsymbol{I} + \boldsymbol{\Lambda})^{-1/2}\boldsymbol{U}^{\intercal}$ is symmetric and therefore results in an unbiased filter [25]. The complete square root update of the PCE coefficients (PCTU) is summarized in Algorithm 1.

The PCTU has a computational benefit over the square root approaches by [31,41] when $n_y > M$ and when $\boldsymbol{C}_{\boldsymbol{e}_f}$ is assumed to be diagonal. The PCTU requires an eigenvalue decomposition on $\hat{\boldsymbol{S}}_l^{\intercal} \boldsymbol{C}_{\boldsymbol{e}_f}^{-1} \hat{\boldsymbol{S}}_l$ of size $M \times M$. This is in contrast with the square root update derived in [31], where one has to apply an eigenvalue decomposition on the matrix $(\boldsymbol{C}_{\boldsymbol{s}_l} + \boldsymbol{C}_{\boldsymbol{e}_f})$ of size $n_y \times n_y$ or with the square root update derived in [41], where one has to take the matrix square root of $(\boldsymbol{C}_{\boldsymbol{s}_l} + \boldsymbol{C}_{\boldsymbol{e}_f})$.

Algorithm 1 Square root update of PCE coefficients.

procedure PCTU($\boldsymbol{z}, \boldsymbol{Q}_f, \boldsymbol{S}_l, \boldsymbol{C}_{e_f}, \Delta$)
  $\bar{\boldsymbol{q}}_f, \tilde{\boldsymbol{Q}}_f \leftarrow \boldsymbol{Q}_f$ ▷ Split coefficients
  $\bar{\boldsymbol{s}}_l, \tilde{\boldsymbol{S}}_l \leftarrow \boldsymbol{S}_l$
  $\hat{\boldsymbol{Q}}_f \leftarrow \tilde{\boldsymbol{Q}}_f \sqrt{\boldsymbol{\Delta}}$ ▷ Scale
  $\hat{\boldsymbol{S}}_l \leftarrow \tilde{\boldsymbol{S}}_l \sqrt{\boldsymbol{\Delta}}$
  $\boldsymbol{K} \leftarrow \hat{\boldsymbol{Q}}_f \hat{\boldsymbol{S}}_l^\intercal (\hat{\boldsymbol{S}}_l \hat{\boldsymbol{S}}_l^\intercal + \boldsymbol{C}_{e_f})^\dagger$ ▷ Kalman gain
  $\bar{\boldsymbol{q}}_a = \bar{\boldsymbol{q}}_f + \boldsymbol{K}(\boldsymbol{z} - \bar{\boldsymbol{s}}_l)$ ▷ Update mean
  $\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^\intercal \leftarrow \text{EIGEN}(\hat{\boldsymbol{S}}_l^\intercal \boldsymbol{C}_{e_f}^{-1} \hat{\boldsymbol{S}}_l)$
  $\boldsymbol{T} \leftarrow \boldsymbol{U}(\boldsymbol{I} + \boldsymbol{\Lambda})^{-1/2} \boldsymbol{U}^\intercal$
  $\hat{\boldsymbol{Q}}_a \leftarrow \hat{\boldsymbol{Q}}_f \boldsymbol{T}$ ▷ Update fluctuating part
  $\tilde{\boldsymbol{Q}}_a = \hat{\boldsymbol{Q}}_a (\sqrt{\boldsymbol{\Delta}})^{-1}$ ▷ De-scale coefficients
  $\boldsymbol{Q}_a \leftarrow [\bar{\boldsymbol{q}}_a, \tilde{\boldsymbol{Q}}_a]$ ▷ Collect coefficients
  return $\boldsymbol{Q}_a$
end procedure

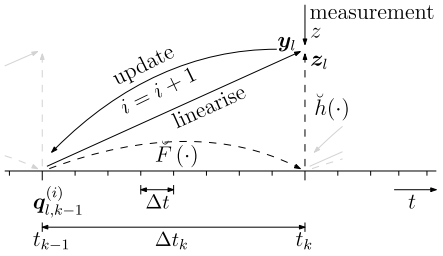

Fig. 1. Schematics of iterative linearisation in a smoother setting. $\breve{F}(\cdot)$ and $\breve{h}(\cdot)$ denote the forward problem and observation function respectively. The solid lines represent the actual update, while the dashed lines represent the function composition used to find linearised observation forecast. Δt_k denotes the discrete time moments at which measurements arrive and Δt denotes the step size of the time integrations scheme used to evaluate $\breve{F}(\cdot)$.

3.4. Smoother

In this work, the iterative PCE filter is used in a smoother setting, where the forecast at the previous time-step, $\boldsymbol{q}_{f,k-1}(\omega)$, is updated using a measurement at the current time-step $\boldsymbol{z}_k$. In this setting the Gauss-Newton iterations linearise the composition of the forward problem and the observation function, instead of the observation function only. The schematics of the iterative linearisation in a smoother setting is shown in Fig. 1.

Solving the forward problem with the assimilated random variable at time t_{k-1} results in an improved forecast at the current time instance, $\boldsymbol{q}_{f,k}(\omega)$. With this forecast, another update can be performed such that one obtains the assimilated random variable at time t_k, i.e. $\boldsymbol{q}_{a,k}(\omega)$.

The performance of the iterative PCE filter in a smoother setting is limited by the quality of the uncertainty propagation step using PCE. The performance of the PCE approximations are known to deteriorate over time. The rate at which this happens is highly problem dependent. The deteriorating performance of the PCE approximations, however, can cause convergence issue in the iterative linearisation step. To resolve this, one has to use time adaptive PCE [17] where the deteriorating performance over time is reduced.

4. Numerical examples

4.1. Experiment A-I

In this section the proposed filter is applied on the Lorenz-63 model, a coupled nonlinear ordinary differential equation introduced in [26]. This problem is well-studied and used frequently in the data assimilation community, see for example [29,13,16,24] and particularly for state and parameter estimation [23,2,1,31]. The governing equations of the model are

$$\begin{aligned} \frac{\mathrm{d}x}{\mathrm{d}t} &= -\sigma x + \sigma y, \\ \frac{\mathrm{d}y}{\mathrm{d}t} &= \rho x - xz - y, \\ \frac{\mathrm{d}z}{\mathrm{d}t} &= xy - \beta z. \end{aligned} \tag{51}$$

Here, x, y, and z are the states collected in $\boldsymbol{x} = [x, y, z]$, and σ, ρ, and β are the parameters collected in $\boldsymbol{p} = [\sigma, \rho, \beta]$ such that Eq. (51) resembles Eq. (1).

The parameters are typically chosen as $\boldsymbol{p} = [10, 28, 8/3]$ and the true initial state is chosen as $\boldsymbol{x}_0 = [1.508870, -1.531271, 25.46091]$ taken from [29]. The reference solution is obtained by integrating Eq. (51) forward in time using a fourth order Runge-Kutta scheme with fixed time intervals of $\Delta t = 0.0005$ to limit the time integration error. The system is observed at time intervals of Δt_k and

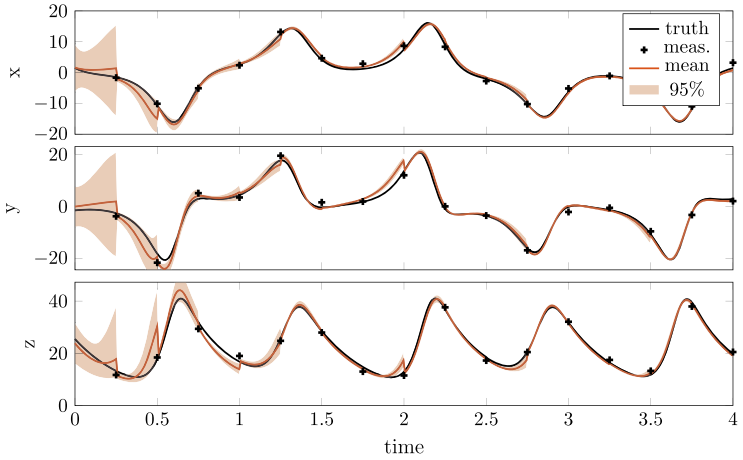

Fig. 2. State estimates from the GN-PCTU smoother ($d = 3$) for the first 4 time units of a single experiment. (For interpretation of the colours in the figure(s), the reader is referred to the web version of this article.)

the noisy measurements are obtained by adding Gaussian noise with a variance of σ_e^2 to the reference solution at the observation instances.

In this experiment, the GN-PCTU smoother is used for simultaneous state and parameter estimation of the Lorenz-63 problem. The initial states and the parameters are uncertain, therefore the total number of random variables in this problem is $N = 6$. The states are observed every 0.25 time units, $\Delta t_k = 0.25$, equivalent to 500 integration steps over an interval of 10 time units resulting in 40 assimilation steps. The magnitude of this observation interval ensures that non-linear dynamics are present between updates (see Fig. 2). Direct measurements are performed, i.e. $h(\boldsymbol{x}) = \boldsymbol{x}$, and the measurement noise variance is set to $\sigma_e^2 = 2$. Note that the parameters are not measured directly, i.e. they are non-linearly present in the measurements. Furthermore, since we are directly measuring the states, we are only approximating the forward problem with Gauss-Newton iterations in this experiment.

The initial state $\boldsymbol{x}_0$ and the parameters $\boldsymbol{p}$ are uncertain and are therefore modelled as independent Gaussian distributed random variables

$$\boldsymbol{x}_0(\omega) \sim \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{x}_0}, \sigma_{\boldsymbol{x}_0}^2 I), \qquad \boldsymbol{p}(\omega) \sim \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{p}}, \sigma_{\boldsymbol{p}}^2 I), \tag{52}$$

respectively. Here $\boldsymbol{\mu}_{\boldsymbol{x}_0}$ and $\boldsymbol{\sigma}_{\boldsymbol{x}_0}^2$ are the prior mean and variance of the initial states, whereas $\boldsymbol{\mu}_{\boldsymbol{x}_0}$ and $\boldsymbol{\sigma}_{\boldsymbol{p}}^2$ denotes the prior mean and variance of the parameters.

The prior mean of the states and parameters itself are drawn from uniform distributions,

$$\mu_{\boldsymbol{x}_0} \sim \mathcal{U}(\boldsymbol{a}_{\boldsymbol{x}_0}, \boldsymbol{b}_{\boldsymbol{x}_0}), \qquad \mu_{\boldsymbol{p}} \sim \mathcal{U}(\boldsymbol{a}_{\boldsymbol{p}}, \boldsymbol{b}_{\boldsymbol{p}}), \tag{53}$$

where $[\boldsymbol{a}_{\boldsymbol{x}_0}, \boldsymbol{b}_{\boldsymbol{x}_0}]$ and $[\boldsymbol{a}_{\boldsymbol{p}}, \boldsymbol{b}_{\boldsymbol{p}}]$ denote the support of the uniform distributions respectively. They are chosen as:

$$\boldsymbol{a}_{\boldsymbol{x}_0} = [-5, -5, 20], \qquad \boldsymbol{b}_{\boldsymbol{x}_0} = [5, 5, 30],$$

and

$$\boldsymbol{a}_{\boldsymbol{p}} = [5, 24, 2/3], \qquad \boldsymbol{b}_{\boldsymbol{p}} = [15, 32, 14/3].$$

The prior distributions are furthermore defined by the variance of the initial states and parameters:

$$\boldsymbol{\sigma}_{\boldsymbol{x}_0}^2 = [4, 4, 4]^2,$$

and

$$\boldsymbol{\sigma}_{\boldsymbol{p}}^2 = [3, 4, 0.75]^2,$$

respectively.

The multivariate basis used to expand the random variables contains Hermite polynomials with a maximum degree of $p = 3$ resulting in $M = 84$ coefficients for each random variable. The maximum number of iterations of GN-PCTU is set to $i_{max} = 100$ and the tolerance is set to $\gamma_{tol} = 1 \cdot 10^{-5}$.

The evolution of the states and their 95% confidence interval for a single experiment using GN-PCTU is shown in Fig. 2. The parameter estimates for the same experiment are shown in Fig. 3. The proposed filter successfully estimates the states and parameters of the system, although residuals for the states and parameters are present due to the observation noise. This residual is especially present for the parameter σ.

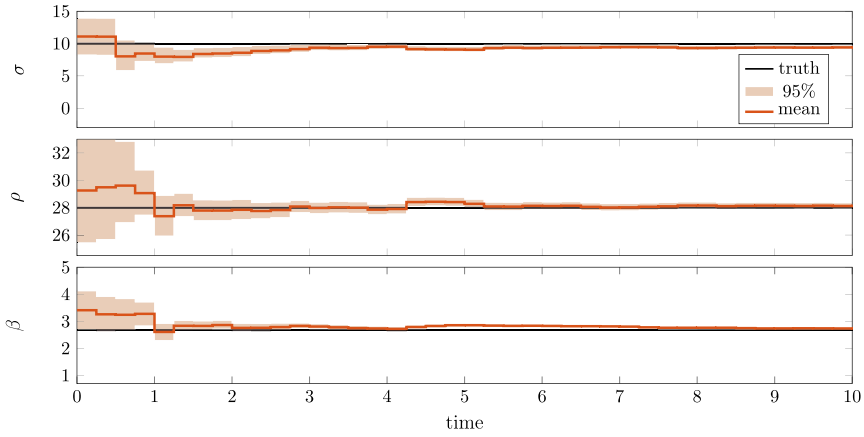

Fig. 3. Parameter estimates from the GN-PCTU smoother ($d = 3$) for a single experiment.

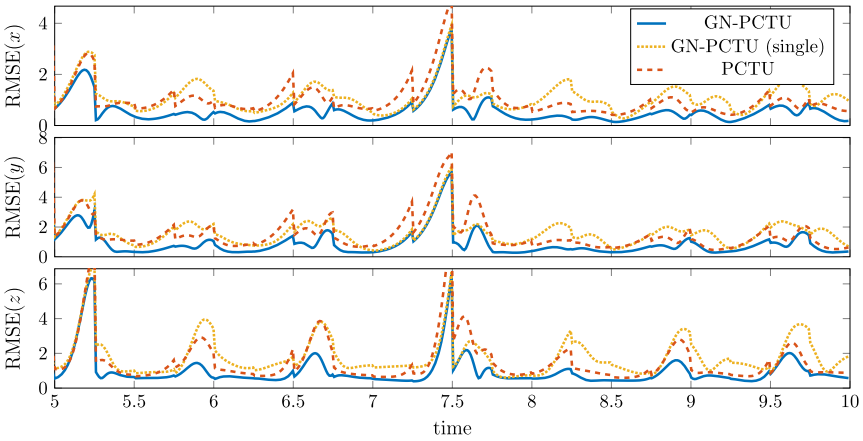

Fig. 4. RMSE error of the estimated states of the Lorenz-63 model obtained from 100 simulation on the interval $t = [5, 10]$. The GN-PCTU smoother is shown twice, once with $i_{max} = 100$ in blue and with a single iteration $i_{max} = 1$ with a dotted yellow line. The PCTU filter is indicated with a dashed red line.

4.2. Experiment A-II

Furthermore, the GN-PCTU smoother is compared to the GN-PCTU smoother with a single Gauss-Newton step, i.e. $i_{max} = 1$, and to the PCTU smoother without linearisation. The latter smoother is essentially the square root PCE implementation of Eq. (11) in a smoother setting. The different smoothers are applied on exactly the same experiment, i.e. the prior and observations fed to the smoother are identical for each smoother. This experiment is repeated 100 times ($N_{mc} = 100$), where the observation noise for each simulations is drawn using a different seed. Subsequently, the root mean squared error (RMSE) is used to evaluate the absolute performance of the filters. This RMSE is calculated over the N_{mc} simulations and is given by:

$$\mathrm{RMSE}_k(\bar{\boldsymbol{q}}) = \sqrt{\frac{1}{N_{mc}} \sum_{n}^{N_{mc}} (\bar{\boldsymbol{q}}_{n,k} - \boldsymbol{q}_k^{true})^2}, \tag{54}$$

where the subscript in $\bar{\boldsymbol{q}}_{n,k}$ denotes the estimated mean at the kth timestep of the nth simulation and $\boldsymbol{q}_k^{true}$ denotes is the kth timestep of the reference solution.

The GN-PCTU smoother shows the best performance in terms of the RMSE over the 100 simulations for the state estimates (Fig. 4) and the parameter estimates (Fig. 5). The GN-PCTU smoother with only a single iteration, i.e. a linearisation around $\mathbb{E}\left[\boldsymbol{q}_f(\omega)\right]$, shows the worst performance. So indeed, using $\mathbb{E}\left[\boldsymbol{q}_f(\omega)\right]$ as a linearisation point leads to suboptimal filters.

4.3. Experiment B

Another problem that can be used to evaluate the performance of the filter is the Lorenz-84 model introduced in [27]. The governing equations of the Lorenz-84 model are

$$\frac{\mathrm{d}x}{\mathrm{d}t} = -y^2 - z^2 - ax + F,$$

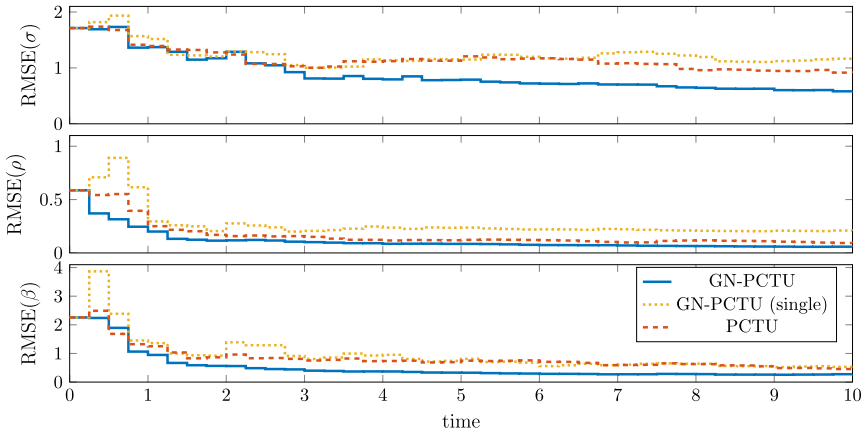

Fig. 5. RMSE error of the estimated parameters of the Lorenz-63 model obtained from 100 simulation. The GN-PCTU smoother is shown twice, once with $i_{max} = 100$ in blue and with a single iteration $i_{max} = 1$ with a dotted yellow line. The PCTU filter is indicated with a dashed red line.

$$\frac{\mathrm{d}y}{\mathrm{d}t} = xy - bxz - y + G, \tag{55}$$
$$\frac{\mathrm{d}z}{\mathrm{d}t} = bxy + xz - z.$$

Here, again, the states are again collected in $\boldsymbol{x} = [x, y, z]$ and the parameters a, b, F, and G are collected in $\boldsymbol{p} = [a, b, F, G]$. The true initial states are chosen as $\boldsymbol{x}_0 = [1, 0, -0.75]$ and the parameters are chosen as $\boldsymbol{p} = [0.25, 4, 2, 1.23]$. The estimation problem is similar to the one defined in experiment A-I in Section 4.1. However, the observation interval is set to $\Delta t_k = 0.5$, resulting in 20 assimilations steps in a time interval of 10 time units. The initial state $\boldsymbol{x}_0$ and the parameters $\boldsymbol{p}$ are uncertain and are modelled in the same manner as in experiment A-I, and are therefore described by Eq. (52) and (53). The total number of random variable in this problem is therefore $N = 7$. The support of the uniform distributions are however chosen as:

$$\boldsymbol{a}_{\boldsymbol{x}_0} = [0, -1, -1.75], \qquad \boldsymbol{b}_{\boldsymbol{x}_0} = [2, 1, 0.25],$$

and

$$\boldsymbol{a}_{\boldsymbol{p}} = [0, 2, 1, 0.5], \qquad \boldsymbol{b}_{\boldsymbol{p}} = [0.5, 6, 3, 2],$$

and the variance of the initial states and parameters are chosen as:

$$\boldsymbol{\sigma}^2_{\boldsymbol{x}_0} = [1, 1, 1]^2, \qquad \boldsymbol{\sigma}^2_{\boldsymbol{p}} = [0.1, 1.5, 0.5, 0.7]^2,$$

respectively.

As was done in experiment A-II in Section 4.2, the GN-PCTU smoother is applied on the problem above and is compared with the GN-PCTU smoother without a single iteration and with the PCTU smoother. To that end, the experiment is repeated 100 times and the RMSE of the states and parameters are evaluated using Eq. (54). Again, the GN-PCTU smoother shows the best performance in terms of the RMSE for both the states (Fig. 6) and parameters (Fig. 7).

4.4. Experiment C

Finally, the computational complexity of the our proposed PCE square root update is compared to those derived in [31] (SRPCU) and [41] (PCSKF). We focus explicitly on the update of the fluctuating part, since this is where the three methods differ in terms of computational complexity. The square root updates are applied on a simplified problem. The prior expansion coefficients are sampled according to a standard Gaussian distribution and are of size $N_q \times M$:

$$\tilde{\boldsymbol{Q}}_f \sim \mathcal{N}(0, 1) \in \mathbb{R}^{N_q, M}.$$

Similarly the measurement is sampled from a standard Gaussian distribution, $\boldsymbol{y}_m \sim \mathcal{N}(0, \boldsymbol{C}_n)$. A diagonal covariance matrix of the measurement noise is assumed, e.g. $\boldsymbol{C}_n = \boldsymbol{I}_{N_y}$. We set $N_q = 4$ and the number of expansion coefficients in this comparison is set to $M = 35$. The three square root updates are applied on $\tilde{\boldsymbol{Q}}_f$ and $\boldsymbol{y}_m$ and its elapsed times are timed using the `timeit` function in Matlab (Fig. 8). Clearly, our proposed square root update of the fluctuating part is faster when the number of sensors N_y is larger than the number of expansion coefficients, M. In other scenario's, the square root approach by [41] is preferred.

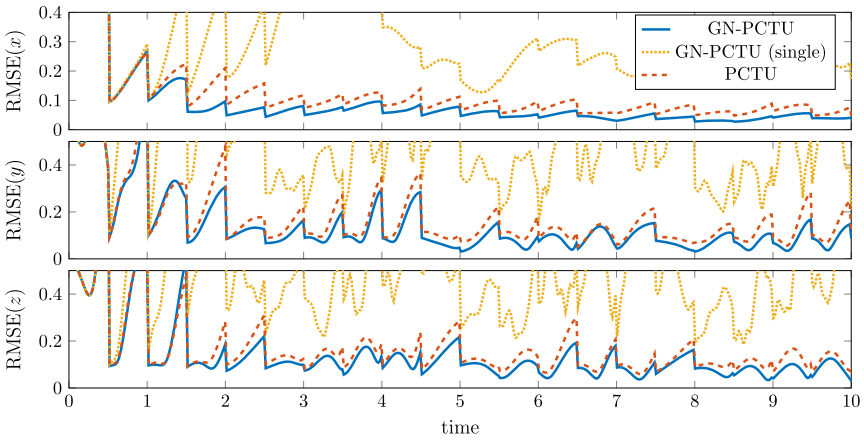

Fig. 6. RMSE error of the estimated states of the Lorenz-84 model obtained from 100 simulation on the interval $t = [0, 10]$. The GN-PCTU smoother is shown twice, once with $i_{max} = 100$ in blue and with a single iteration $i_{max} = 1$ with a dotted yellow line. The PCTU filter is indicated with a dashed red line.

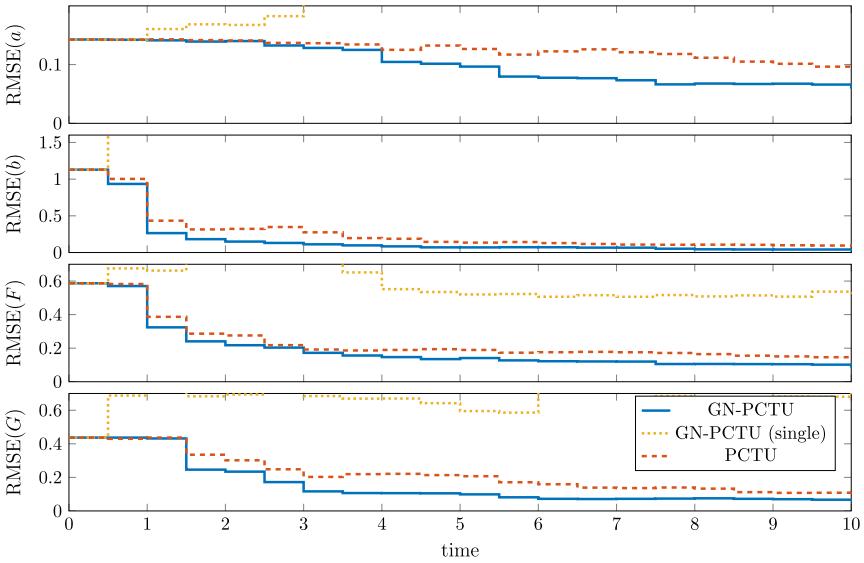

Fig. 7. RMSE error of the estimated parameters of the Lorenz-84 model obtained from 100 simulation. The GN-PCTU smoother is shown twice, once with $i_{max} = 100$ in blue and with a single iteration $i_{max} = 1$ with a dotted yellow line. The PCTU filter is indicated with a dashed red line.

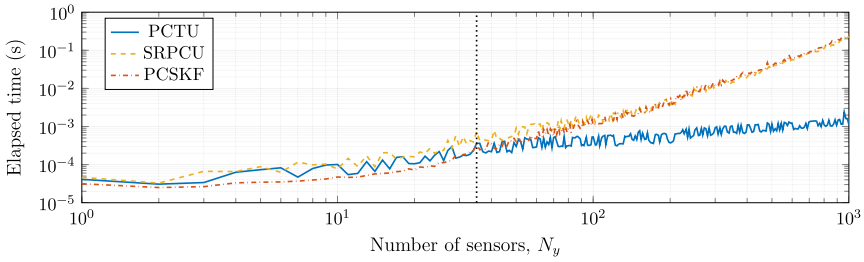

Fig. 8. Elapsed time of square root updates, evaluated with the `timeit` function in Matlab. Number of expansion coefficients M used in this comparison is indicated with a vertical line.

5. Conclusion

This paper presents a sampling free square root filter with Gauss-Newton iterations derived from a general Kalman filter formulation. In this filter, all random variables are discretized using PCE. The expansion coefficients of these PCEs were found without sampling by using a Galerkin projection method. The non-linearities present in the system are linearised by Gauss-Newton iterations

such that the non-Gaussian random variables are propagated forward linearly. Subsequently, the expansion coefficients are updated directly using a square root implementation such that there is no growth in the polynomial basis. This square root implementation was compared to two existing square root approaches in terms of numerical complexity.

For illustration, the filter was used for state and parameter estimation of the modestly sized Lorenz-63 and Lorenz-84 systems. On these problems, the sampling-free iterative filter was compared to the same filter with a single iteration and to the same filter without linearisation. Out of the three filters, the sampling-free iterative filter showed the best performance in terms of the RMSE error of the states and parameters. For reasonably sized systems, the Galerkin projection method is faster when compared to sampling based methods. However, in different scenario's, non-intrusive methods might be favourable and result in faster implementations.

In future work we will apply the iterative sampling free filter on an experimental setup for state and parameter estimation. Furthermore, we will combine this filter with stochastic and adaptive model predictive control.

CRediT authorship contribution statement

W. van Dijk: Methodology, Software, Visualization, Writing – original draft. **W.B.J. Hakvoort:** Supervision, Writing – review & editing. **B. Rosić:** Methodology, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Appendix A. Multi-index set

The multi-index $\mathcal{J}$ is defined by, $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_n) \in \mathcal{J} := \mathbb{N}_0^d$. The truncated multi-index set limits the number of basis polynomials by removing polynomial terms higher than d, i.e. $\mathcal{J}_t = \{\alpha \in \mathcal{J} | \sum \alpha \leq d\}$. The fluctuating part can be found by removing the first index, representing the mean, from the truncated multi-index set. The multi-index is then described by $\mathcal{J}_{\neq 0} = \{\alpha \in \mathcal{J}_t | \sum \alpha > 0\}$.

Appendix B. Affine map

Recall the optimum of the optimality condition (Eq. (17)):

$$\hat{\boldsymbol{H}}^{(i)} = \boldsymbol{C}_{\boldsymbol{y}_l^{(i)} \boldsymbol{q}_a^{(i)}} \boldsymbol{C}^{-1}_{\boldsymbol{q}_a^{(i)}}, \tag{B.1}$$

$$\hat{\boldsymbol{h}}^{(i)} = \mathbb{E}\left[\boldsymbol{y}_l^{(i)}(\omega)\right] - \hat{\boldsymbol{H}}^{(i)}(\mathbb{E}\left[\boldsymbol{q}_a^{(i)}(\omega)\right] - \hat{\boldsymbol{q}}^{(i)}). \tag{B.2}$$

For notational simplicity consider a single iteration, such that

$$\hat{\boldsymbol{H}} = \boldsymbol{C}_{\boldsymbol{y}_l \boldsymbol{q}_a} \boldsymbol{C}^{-1}_{\boldsymbol{q}_a}, \tag{B.3}$$

$$\hat{\boldsymbol{h}} = \mathbb{E}\left[\boldsymbol{y}_l(\omega)\right] - \hat{\boldsymbol{H}}(\mathbb{E}\left[\boldsymbol{q}_a(\omega)\right] - \hat{\boldsymbol{q}}), \tag{B.4}$$

holds. The covariance matrices in Eq. (B.3) can be expressed in terms of the expansion coefficients:

$$\boldsymbol{C}_{\boldsymbol{y}_l \boldsymbol{q}_a} = \tilde{\boldsymbol{Y}}_l \Delta \tilde{\boldsymbol{Q}}_a^\intercal, \qquad \boldsymbol{C}_{\boldsymbol{q}_a} = \tilde{\boldsymbol{Q}}_a \Delta \tilde{\boldsymbol{Q}}_a^\intercal. \tag{B.5}$$

Substitution of Eq. (B.3) in (B.5) yields:

$$\begin{aligned}
\hat{\boldsymbol{H}} &= \boldsymbol{C}_{\boldsymbol{y}_l \boldsymbol{q}_a} \boldsymbol{C}^{-1}_{\boldsymbol{q}_a} \\
&= \tilde{\boldsymbol{Y}}_l \Delta \tilde{\boldsymbol{Q}}_l^\intercal \left(\tilde{\boldsymbol{Q}}_a \Delta \tilde{\boldsymbol{Q}}_a^\intercal\right)^{-1} \\
&= \tilde{\boldsymbol{Y}}_l \sqrt{\Delta} \left(\tilde{\boldsymbol{Q}}_a \sqrt{\Delta}\right)^\intercal \left(\tilde{\boldsymbol{Q}}_a \sqrt{\Delta} \left(\tilde{\boldsymbol{Q}}_a \sqrt{\Delta}\right)^\intercal\right)^{-1} \\
&= \tilde{\boldsymbol{Y}}_l \sqrt{\Delta} \left(\tilde{\boldsymbol{Q}}_a \sqrt{\Delta}\right)^\intercal \left(\tilde{\boldsymbol{Q}}_a \sqrt{\Delta}\right)^{-\intercal} \left(\tilde{\boldsymbol{Q}}_a \sqrt{\Delta}\right)^{-1} \\
&= \tilde{\boldsymbol{Y}}_l \sqrt{\Delta} \sqrt{\Delta}^{-1} \tilde{\boldsymbol{Q}}_a^{-1} \\
&= \tilde{\boldsymbol{Y}}_l \tilde{\boldsymbol{Q}}_a^{-1}
\end{aligned} \tag{B.6}$$

The expectations in Eq. (B.4) can be substituted by the approximated mean values that follow from the PCE approximations, i.e.

$$\hat{\boldsymbol{h}} = \bar{\boldsymbol{y}}_l - \hat{\boldsymbol{H}}(\bar{\boldsymbol{q}}_a - \hat{\boldsymbol{q}}). \tag{B.7}$$

Appendix C. Augmented ODE

Consider the first equation of the Lorenz-63 system,

$$\frac{\mathrm{d}x(t)}{\mathrm{d}t} = \sigma(y - x). \tag{C.1}$$

By substitution of the states and parameters with random variables, one obtains:

$$\frac{\mathrm{d}x(t,\omega)}{\mathrm{d}t} = \sigma(\omega)(y(t,\omega) - x(t,\omega)). \tag{C.2}$$

Discretization of the time-dependent states, $x(t,\omega)$ and $y(t,\omega)$, and the parameter $\sigma(\omega)$ by PCEs yields

$$x(t,\omega) \approx \sum_{\alpha\in\mathcal{J}_t} x^{(\alpha)}(t)\mathbf{\Phi}_{\alpha}, \tag{C.3}$$

$$x(t,\omega) \approx \sum_{\alpha\in\mathcal{J}_t} y^{(\alpha)}(t)\mathbf{\Phi}_{\alpha}, \tag{C.4}$$

$$\sigma(\omega) \approx \sum_{\alpha\in\mathcal{J}_t} x^{(\alpha)}\mathbf{\Phi}_{\alpha}, \tag{C.5}$$

respectively. Substitution in the stochastic ODE and dropping the notation of time dependency of the states for notational simplicity,

$$\sum_{\alpha\in\mathcal{J}_t} \frac{\mathrm{d}x^{(\alpha)}}{\mathrm{d}t}\mathbf{\Phi}_{\alpha} = \sum_{\alpha\in\mathcal{J}_t}\sum_{\beta\in\mathcal{J}_t} \sigma^{(\alpha)}(y^{(\beta)} - x^{(\beta)})\mathbf{\Phi}_{\alpha}\mathbf{\Phi}_{\beta} \tag{C.6}$$

Projection on the space spanned by the stochastic basis $\{\mathbf{\Phi}_{\alpha}\}$ gives,

$$\sum_{\alpha\in\mathcal{J}_t} \frac{\mathrm{d}x^{(\alpha)}}{\mathrm{d}t}\left\langle\mathbf{\Phi}_{\alpha}\mathbf{\Phi}_{\gamma}\right\rangle = \sum_{\alpha\in\mathcal{J}_t}\sum_{\beta\in\mathcal{J}_t} \sigma^{(\alpha)}(y^{(\beta)} - x^{(\beta)})\left\langle\mathbf{\Phi}_{\alpha}\mathbf{\Phi}_{\beta}\mathbf{\Phi}_{\gamma}\right\rangle. \tag{C.7}$$

After utilizing the orthogonal property of the polynomial basis, one obtains the following,

$$\frac{\mathrm{d}x^{(\gamma)}}{\mathrm{d}t} = \frac{1}{\left\langle\mathbf{\Phi}_{\gamma}^2\right\rangle}\sum_{\alpha\in\mathcal{J}_t}\sum_{\beta\in\mathcal{J}_t} \sigma^{(\alpha)}(y^{(\beta)} - x^{(\beta)})C_{\alpha\beta\gamma} \tag{C.8}$$

where $C_{\alpha\beta\gamma} = \left\langle\mathbf{\Phi}_{\alpha}\mathbf{\Phi}_{\beta}\mathbf{\Phi}_{\gamma}\right\rangle$, denotes the triple inner product.

The result is a deterministic ODE, in which the states are the expansion coefficients $x^{(\alpha)}(t)$, $y^{(\alpha)}(t)$, and the parameters are $\sigma^{(\alpha)}$. Similarly, the augmented equations for the rest of the lorenz-63 system can be derived,

$$\begin{aligned}
\frac{\mathrm{d}y^{(\gamma)}}{\mathrm{d}t} &= \frac{1}{\left\langle\mathbf{\Phi}_{\gamma}^2\right\rangle}\sum_{\alpha\in\mathcal{J}_t}\sum_{\beta\in\mathcal{J}_t}\left(\rho^{(\alpha)}x^{(\beta)} - z^{(\alpha)}x^{(\beta)}\right)C_{\alpha\beta\gamma} - y^{(\gamma)},\\
\frac{\mathrm{d}z^{(\gamma)}}{\mathrm{d}t} &= \frac{1}{\left\langle\mathbf{\Phi}_{\gamma}^2\right\rangle}\sum_{\alpha\in\mathcal{J}_t}\sum_{\beta\in\mathcal{J}_t}\left(x^{(\alpha)}y^{(\beta)} - \beta^{(\alpha)}z^{(\beta)}\right)C_{\alpha\beta\gamma}.
\end{aligned} \tag{C.9}$$

References

[1] J.T. Ambadan, Y. Tang, Sigma-point Kalman filter data assimilation methods for strongly nonlinear systems, J. Atmos. Sci. 66 (2) (2009) 261–285, https://doi.org/10.1175/2008JAS2681.1.

[2] J. Annan, J. Hargreaves, Efficient parameter estimation for a highly chaotic system, Tellus, Ser. A Dyn. Meteorol. Oceanogr. 56 (5) (2004) 520–526, https://doi.org/10.3402/tellusa.v56i5.14438.

[3] V.A. Bavdekar, A. Mesbah, A polynomial chaos-based nonlinear Bayesian approach for estimating state and parameter probability distribution functions, in: 2016 American Control Conference (ACC), 2016, pp. 2047–2052.

[4] B.M. Bell, The iterated Kalman smoother as a Gauss–Newton method, SIAM J. Optim. 4 (3) (1994) 626–636, https://doi.org/10.1137/0804035.

[5] C.H. Bishop, B.J. Etherton, S.J. Majumdar, Adaptive sampling with the ensemble transform Kalman filter. Part I: theoretical aspects, Mon. Weather Rev. 129 (3) (2001) 420–436, https://doi.org/10.1175/1520-0493(2001)129<0420:ASWTET>2.0.CO;2.

[6] E.D. Blanchard, A. Sandu, C. Sandu, A polynomial chaos-based Kalman filter approach for parameter estimation of mechanical systems, J. Dyn. Syst. Meas. Control 132 (2010) 1–18, https://doi.org/10.1115/1.4002481.

[7] G. Blatman, B. Sudret, Adaptive sparse polynomial chaos expansion based on least angle regression, J. Comput. Phys. 230 (6) (2011) 2345–2367, https://doi.org/10.1016/j.jcp.2010.12.021.

[8] A. Bobrowski, Functional Analysis for Probability and Stochastic Processes: An Introduction, Cambridge University Press, 2005.

[9] G. Burgers, P.J. Van Leeuwen, G. Evensen, Analysis scheme in the ensemble Kalman filter, Mon. Weather Rev. 126 (6) (1998) 1719–1724, https://doi.org/10.1175/1520-0493(1998)126<1719:ASITEK>2.0.CO;2.

[10] S. Chen, Kalman filter for robot vision: a survey, IEEE Trans. Ind. Electron. 59 (11) (2011) 4409–4420, https://doi.org/10.1109/TIE.2011.2162714.

[11] B.J. Debusschere, H.N. Najm, P.P. Pébay, O.M. Knio, R.G. Ghanem, O.P. Le Maître, Numerical challenges in the use of polynomial chaos representations for stochastic processes, SIAM J. Sci. Comput. 26 (2) (2004) 698–719, https://doi.org/10.1137/S1064827503427741.

[12] M. Eldred, Recent advances in non-intrusive polynomial chaos and stochastic collocation methods for uncertainty analysis and design, in: 50th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference 17th AIAA/ASME/AHS Adaptive Structures Conference 11th AIAA No, 2009, p. 2274.
[13] G. Evensen, Advanced data assimilation for strongly nonlinear dynamics, Mon. Weather Rev. 125 (6) (1997) 1342–1354, https://doi.org/10.1175/1520-0493(1997)125<1342:ADAFSN>2.0.CO;2.
[14] G. Evensen, The ensemble Kalman filter: theoretical formulation and practical implementation, Ocean Dyn. 53 (2003) 343–347, https://doi.org/10.1007/s10236-003-0036-9.
[15] G. Evensen, Sampling strategies and square root analysis schemes for the EnKF, Ocean Dyn. 54 (6) (2004) 539–560, https://doi.org/10.1007/s10236-004-0099-2.
[16] G. Evensen, P.J. Van Leeuwen, An ensemble Kalman smoother for nonlinear dynamics, Mon. Weather Rev. 128 (6) (2000) 1852–1867, https://doi.org/10.1175/1520-0493(2000)128<1852:AEKSFN>2.0.CO;2.
[17] M. Gerritsma, J.B. Van der Steen, P. Vos, G. Karniadakis, Time-dependent generalized polynomial chaos, J. Comput. Phys. 229 (22) (2010) 8333–8363.
[18] M. Goldstein, D. Wooff, Bayes Linear Statistics: Theory and Methods, John Wiley & Sons, 2007.
[19] S. Gratton, A.S. Lawless, N.K. Nichols, Approximate Gauss–Newton methods for nonlinear least squares problems, SIAM J. Optim. 18 (1) (2007) 106–132.
[20] N.J. Higham, Accuracy and Stability of Numerical Algorithms, SIAM, 2002.
[21] A.H. Jazwinski, Stochastic Processes and Filtering Theory, Academic Press, San Diego, CA, 1970.
[22] R.E. Kalman, A new approach to linear filtering and prediction problems, J. Basic Eng. 82 (1) (1960) 35–45, https://doi.org/10.1115/1.3662552.
[23] G. Kivman, Sequential parameter estimation for stochastic systems, Nonlinear Process. Geophys. 10 (3) (2003) 253–259, https://doi.org/10.5194/npg-10-253-2003.
[24] J. Li, D. Xiu, A generalized polynomial chaos based ensemble Kalman filter with high accuracy, J. Comput. Phys. 228 (2009) 5454–5469, https://doi.org/10.1016/j.jcp.2009.04.029.
[25] D.M. Livings, S.L. Dance, N.K. Nichols, Unbiased ensemble square root filters, Phys. D: Nonlinear Phenom. 237 (8) (2008) 1021–1028, https://doi.org/10.1016/j.physd.2008.01.005.
[26] E.N. Lorenz, Deterministic nonperiodic flow, J. Atmos. Sci. 20 (2) (1963) 130–141, https://doi.org/10.1175/1520-0469(1963)020<0130:DNF>2.0.CO;2.
[27] E.N. Lorenz, Irregularity: a fundamental property of the atmosphere, Tellus A 36 (2) (1984) 98–110.
[28] N. Lüthen, S. Marelli, B. Sudret, Sparse polynomial chaos expansions: literature survey and benchmark, SIAM/ASA J. Uncertain. Quantificat. 9 (2) (2021) 593–649.
[29] R.N. Miller, M. Ghil, F. Gauthiez, Advanced data assimilation in strongly nonlinear dynamical systems, J. Atmos. Sci. 51 (8) (1994) 1037–1056, https://doi.org/10.1175/1520-0469(1994)051<1037:ADAISN>2.0.CO;2.
[30] O. Pajonk, B.V. Rosić, A. Litvinenko, H.G. Matthies, A deterministic filter for non-Gaussian Bayesian estimation—applications to dynamical system estimation with noisy measurements, Phys. D: Nonlinear Phenom. 241 (7) (2012) 775–788, https://doi.org/10.1016/j.physd.2012.01.001.
[31] O. Pajonk, B.V. Rosić, H.G. Matthies, Sampling-free linear Bayesian updating of model state and parameters using a square root approach, Comput. Geosci. 55 (2013) 70–83, https://doi.org/10.1016/j.cageo.2012.05.017.
[32] B. Rosic, Stochastic State Estimation via Incremental Iterative Sparse Polynomial Chaos Based Bayesian-Gauss-Newton-Markov-Kalman Filter, 2019.
[33] B.V. Rosić, A. Kučerová, J. Sỳkora, O. Pajonk, A. Litvinenko, H.G. Matthies, Parameter identification in a probabilistic setting, Eng. Struct. 50 (2013) 179–196, https://doi.org/10.1016/j.engstruct.2012.12.029.
[34] B.V. Rosić, A. Litvinenko, O. Pajonk, H.G. Matthies, Sampling-free linear Bayesian update of polynomial chaos representations, J. Comput. Phys. 231 (17) (2012) 5761–5787, https://doi.org/10.1016/j.jcp.2012.04.044.
[35] G. Saad, R. Ghanem, Characterization of reservoir simulation models using a polynomial chaos-based ensemble Kalman filter, Water Resour. Res. 45 (2009) 1–19, https://doi.org/10.1029/2008WR007148.
[36] S. Särkkä, L. Svensson, Bayesian Filtering and Smoothing, Institute of Mathematical Statistics Textbooks, 2 ed., Cambridge University Press, 2023.
[37] W. Slika, G. Saad, A practical polynomial chaos Kalman filter implementation using nonlinear error projection on a reduced polynomial chaos expansion, Int. J. Numer. Methods Eng. 112 (2017) 1869–1885, https://doi.org/10.1002/nme.5586.
[38] J.S. Whitaker, T.M. Hamill, Ensemble data assimilation without perturbed observations, Mon. Weather Rev. 130 (7) (2002) 1913–1924, https://doi.org/10.1175/1520-0493(2002)130<1913:EDAWPO>2.0.CO;2.
[39] D. Xiu, G.E. Karniadakis, The Wiener–Askey polynomial chaos for stochastic differential equations, SIAM J. Sci. Comput. 24 (2) (2002) 619–644, https://doi.org/10.1137/S1064827501387826.
[40] D. Xiu, J. Shen, Efficient stochastic Galerkin methods for random diffusion equations, J. Comput. Phys. 228 (2) (2009) 266–281, https://doi.org/10.1016/j.jcp.2008.09.008.
[41] Z. Yu, P. Cui, M. Ni, A polynomial chaos based square-root Kalman filter for Mars entry navigation, Aerosp. Sci. Technol. 51 (2016) 192–202, https://doi.org/10.1016/j.ast.2016.02.009.
[42] L. Zeng, D. Zhang, A stochastic collocation based Kalman filter for data assimilation, Comput. Geosci. 14 (4) (2010) 721–744, https://doi.org/10.1007/s10596-010-9183-5.
[43] Z. Zhang, Parameter estimation techniques: a tutorial with application to conic fitting, Image Vis. Comput. 15 (1) (1997) 59–76, https://doi.org/10.1016/S0262-8856(96)01112-2.