Nonlinear Filtering with a Polynomial Series of Gaussian Random Variables

Simone Servadio *The University of Texas at Austin*, Austin, Texas, simo_serva@utexas.edu
Renato Zanetti *The University of Texas at Austin*, Austin, Texas, renato@utexas.edu
Brandon A. Jones *The University of Texas at Austin*, Austin, Texas, brandon.jones@utexas.edu

***Abstract*—Filters relying on the Gaussian approximation typically incorporate the measurement linearly, i.e., the value of the measurement is pre-multiplied by a matrix-valued gain in the state update. Nonlinear filters that relax the Gaussian assumption, on the other hand, typically approximate the distribution of the state with a finite sum of point masses or Gaussian distributions. In this work, the distribution of the state is approximated by a polynomial transformation of a Gaussian distribution, allowing for all moments, central and raw, to be rapidly computed in closed form. Knowledge of the higher-order moments is then employed to perform a polynomial measurement update, i.e., the value of the measurement enters the update function as a polynomial of arbitrary order. A filter employing a Gaussian approximation with linear update is, therefore, a special case of the proposed algorithm when the order of the update is set to one. At the cost of more computations, the new methodology guarantees performance better than the linear/Gaussian approach for nonlinear systems. This work employs monomial basis functions and Taylor series, but it is readily extendable to an orthogonal polynomial basis.**

***Index Terms*—Differential Algebra, Nonlinear filtering, Lorenz96, Polynomial update**

I. Introduction

The conditional mean is the optimal estimate to the sequential stochastic Minimum Mean Square Error (MMSE) estimation problem. It is derived recursively thought Bayes' rule. When the system is linear and Gaussian, the Kalman filter [1] [2] exactly represents the conditional distribution. However, when the dynamics of the system are nonlinear, the conditional probability density function (PDF) is generally a non-Gaussian distribution, and the optimal MMSE estimate is not typically obtainable in closed form. Consequentially, many algorithms have been developed that approximate the optimal MMSE solution in the presence of nonlinearities. A simple approach is to perform a linearization of the measurement and dynamics functions around the current estimate, and apply the Kalman filter equations as if the system were linear; the so-called Extended Kalman Filter (EKF) [3]. However, simple linearization fails to achieve convergence for highly nonlinear systems [4].

The Unscented Kalman Filter (UKF) [5], [6] usually achieves better results and is more robust to nonlinearities than the EKF. Using deterministically placed sigma points, the distribution of the state is approximated by applying the actual nonlinear dynamics and nonlinear measurement function to estimate the mean and covariance matrix via a weighted average.

Several methods to better incorporate the nonlinear dynamics during the time propagation phase of filtering algorithms exist. Park and Sheeres [7] employ State Transition Tensors (STT) to propagate mean and high order central moments. Majji et al. [8] expand Park and Sheeres work to include the high-order moments in the update. Valli et al. [9] reproduced these results in the differential algebra (DA) framework.

All above-mentioned filters, including the EKF and UKF, are linear estimators, i.e. the estimate is a linear function of the measurements. The conditional mean, which is the optimal MMSE solution, is generally some unknown nonlinear function of the measurement whose exact form usually cannot be calculated. A linear estimator, even when accounting for the nonlinearities of the measurement function, is typically outperformed by nonlinear estimators such as the Gaussian Sum Filter (GSF) [10] [11] or particle filters [12].

In the GSF, the optimal nonlinear update is approximated by approximating all distributions as Gaussian Mixture Models (GMM). As long as each component of the GMM has a covariance "small" enough, linearization of the measurement and dynamic functions can accurately represent the uncertainty in the component's support, and linear update equations (EKF or UKF) can be used for each component. The total update is a weighted combination of the linear updates, where each weight is obtained from the likelihood of the corresponding Gaussian component.

De Santis et al. [13] developed a quadratic update by augmenting the state of the system with its square. The estimator in [13] also augments the measurement with its square, by doing so they are able to rewrite the quadratic update with a linear update structure. A polynomial approximation of the optimal nonlinear update also exists [14].

Servadio and Zanetti [15] implemented a quadratic update (extendable to a polynomial update of any order) based on Taylor series expansions. By carrying central moments up to a desired order, the polynomial high-order coefficients are evaluated to minimize the mean square estimation error. The computational demand of calculating higher order central moments grows quickly with the order of the series, the size of the state vector, and the order of the polynomial update.

In this work all non-Gaussian distributions are approximated as a polynomial transformation of Gaussian random variables. In doing so, all high order central moments are easily and efficiently calculated in close form with Isserlis formula [16]. As a consequence, polynomial updates can be performed more efficiently than in prior works.

The paper is structured as follows: Section II presents the mathematics and fundamentals for the newly proposed polynomial estimator; Section III describes the filtering algorithm and its implementation in the DA framework; then, Section IV applies the new filtering techniques to a challenging simulation of a Lorenz96 application [17], [18]; lastly, Section V draws conclusions.

II. Polynomial Estimator

Our previous work presented a nonlinear update using polynomial residuals [15]. While the polynomial approach can be extended to arbitrary order, the quadratic update is shown here. Let $\mathbf{x}$ be the state of the system we wish to estimate, modeled as a random vector, and $\mathbf{y}$ another, related, random vector that is sampleable. We will use the known outcome of $\mathbf{y}$ to estimate the unknown outcome of $\mathbf{x}$. Let $\mathbf{g}(\mathbf{y})$ be a family of quadratic estimators of $\mathbf{x}$ from $\mathbf{y}$, defined by constants $\mathbf{a}$, which is a vector, and $\mathbf{K}_1, \mathbf{K}_2$ which are matrices of appropriate dimensions.

$$\mathbf{g}(\mathbf{y}) = \mathbf{a} + \mathbf{K}_1\mathbf{y} + \mathbf{K}_2\mathbf{y}^{[2]} \tag{1}$$

where $\mathbf{y}^{[2]} = \mathbf{y} \otimes \mathbf{y}$ and the symbol $\otimes$ indicates the Kronecker product between two vectors, but with the redundant components of $\mathbf{y} \otimes \mathbf{y}$ eliminated, e.g., only one y_1y_2 or y_2y_1 is present in $\mathbf{y}^{[2]}$. It is often convenient to work with deviation vectors of the variables from their mean instead of the vectors themselves. Therefore let us define the quantity $d\mathbf{y} = \mathbf{y} - \mathbb{E}\{\mathbf{y}\}$, which indicates the measurement residual, and similarly, the state deviation as $d\mathbf{x} = \mathbf{x} - \mathbb{E}\{\mathbf{x}\}$. Ref. [15] defines the quadratic estimator in terms of $d\mathbf{y}^{[2]} = d\mathbf{y} \otimes d\mathbf{y}$, but, in this work, a different quantity is used:

$$d\mathbf{y}^{\{2\}} = \mathbf{y} \otimes \mathbf{y} - \mathbb{E}\{\mathbf{y} \otimes \mathbf{y}\} = \mathbf{y}^{[2]} - \mathbb{E}\left\{\mathbf{y}^{[2]}\right\}$$

The two formulations are equivalent, but the latter is convenient because it has zero mean. Thus, we work with the deviation of the square instead of the square of the deviation. Without any loss of generality, we can redefine $\mathbf{a}$, $\mathbf{K}_1$, and $\mathbf{K}_2$ by adding and subtracting constants in order to obtain a different, but equivalent, family of quadratic estimators

$$\mathbf{g}(\mathbf{y}) = \mathbf{a} + \mathbb{E}\{\mathbf{x}\} + \mathbf{K}_1(\mathbf{y} - \mathbb{E}\{\mathbf{y}\}) + \mathbf{K}_2\left(\mathbf{y}^{[2]} - \mathbb{E}\left\{\mathbf{y}^{[2]}\right\}\right) \tag{2}$$

It is known that the optimal values of the estimator's coefficients (denoted with an asterisk) satisfy the orthogonality principle

$$\mathbb{E}\left\{\left(d\mathbf{x} - \mathbf{a}^* - \mathbf{K}_1^* d\mathbf{y} - \mathbf{K}_2^* d\mathbf{y}^{\{2\}}\right)\cdot \cdot\left(\mathbf{a} + \mathbb{E}\{\mathbf{x}\} + \mathbf{K}_1 d\mathbf{y} + \mathbf{K}_2 d\mathbf{y}^{\{2\}}\right)^{\mathrm{T}}\right\} = \mathbf{O} \tag{3}$$

which is valid $\forall \mathbf{a}, \mathbf{K}_1, \mathbf{K}_2$. Therefore the optimal coefficients can be found solving the linear system

$$\mathbf{a}^* = \mathbf{0} \tag{4}$$

$$\mathbf{K}_1^*\mathbf{P}_{d\mathbf{y}d\mathbf{y}} + \mathbf{K}_2^*\mathbf{P}_{d\mathbf{y}^{\{2\}}d\mathbf{y}} = \mathbf{P}_{d\mathbf{x}d\mathbf{y}} \tag{5}$$

$$\mathbf{K}_1^*\mathbf{P}_{d\mathbf{y}d\mathbf{y}^{\{2\}}} + \mathbf{K}_2^*\mathbf{P}_{d\mathbf{y}^{\{2\}}d\mathbf{y}^{\{2\}}} = \mathbf{P}_{d\mathbf{x}d\mathbf{y}^{\{2\}}} \tag{6}$$

where the following notation is used:

$$\mathbf{P}_{\mathbf{xy}} = \mathbb{E}\left\{(\mathbf{x} - \mathbb{E}\{\mathbf{x}\})\ (\mathbf{y} - \mathbb{E}\{\mathbf{y}\})^{\mathrm{T}}\right\} \tag{7}$$

Since $\mathbf{y}^{[2]}$ and $d\mathbf{y}^{\{2\}}$ only differ by a constant, they share the same covariance matrix.

The optimal solution is given by

$$\mathbf{a}^* = \mathbf{0} \tag{8}$$

$$\begin{bmatrix}\mathbf{K}_1^* & \mathbf{K}_2^*\end{bmatrix} = \begin{bmatrix}\mathbf{P}_{d\mathbf{x}d\mathbf{y}}^{\mathrm{T}} \\ \mathbf{P}_{d\mathbf{x}d\mathbf{y}^{\{2\}}}^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}} \begin{bmatrix}\mathbf{P}_{d\mathbf{y}d\mathbf{y}} & \mathbf{P}_{d\mathbf{y}d\mathbf{y}^{\{2\}}} \\ \mathbf{P}_{d\mathbf{y}^{\{2\}}d\mathbf{y}} & \mathbf{P}_{d\mathbf{y}^{\{2\}}d\mathbf{y}^{\{2\}}}\end{bmatrix}^{-1} \tag{9}$$

The optimal estimator is

$$\hat{\mathbf{x}} = \mathbb{E}\{\mathbf{x}\} + \mathbf{K}_1^* d\mathbf{y} + \mathbf{K}_2^*\ d\mathbf{y}^{\{2\}} \tag{10}$$

and the posterior estimation error is

$$\mathbf{e} = \mathbf{x} - \hat{\mathbf{x}} = \mathbf{x} - \mathbf{g}(\mathbf{y}) = \tag{11}$$

$$= \mathbf{x} - \left(\mathbb{E}\{\mathbf{x}\} + \mathbf{K}_1^* d\mathbf{y} + \mathbf{K}_2^* d\mathbf{y}^{\{2\}}\right) \tag{12}$$

$$= d\mathbf{x} - \mathbf{K}_1^* d\mathbf{y} - \mathbf{K}_2^* d\mathbf{y}^{\{2\}} \tag{13}$$

A. Generalization to Higher Orders

It is possible to generalize the results of the previous section to higher orders of the polynomial update. Let us expand Equation (1) to an arbitrary order

$$\mathbf{g}(\mathbf{y}) = \mathbf{a} + \mathbf{K}_1\mathbf{y} + \mathbf{K}_2\mathbf{y}^{[2]} + \mathbf{K}_3\mathbf{y}^{[3]} + \mathbf{K}_4\mathbf{y}^{[4]} + \dots \tag{14}$$

where each $\mathbf{K}_i$ is a matrix of appropriate dimensions and each $\mathbf{y}^{[i]}$ is calculated using the Kronecker product with the repeated elements deleted. The measurements and its powers can be stacked, defining the augmented measurement vector

$$\mathcal{Y} = \begin{bmatrix}\mathbf{y}^{\mathrm{T}} & \mathbf{y}^{[2]T} & \mathbf{y}^{[3]T} & \dots\end{bmatrix}^{\mathrm{T}} \tag{15}$$

Once again, the estimator family is redefined in order to work with deviations. Since deviations have zero mean by construction, the identities $\mathbf{P}_{\mathbf{y}^{[i]}\mathbf{y}^{[j]}} = \mathbf{P}_{d\mathbf{y}^{\{i\}}d\mathbf{y}^{\{j\}}}$ and $\mathbf{P}_{\mathbf{x}\mathbf{y}^{[j]}} = \mathbf{P}_{d\mathbf{x}d\mathbf{y}^{\{j\}}}$ are valid $\forall i, j \in \mathbb{N}_0$. Therefore, it is possible to evaluate the augmented measurement covariance matrix as

$$\mathbf{P}_{\mathcal{Y}\mathcal{Y}} = \begin{bmatrix}\mathbf{P}_{\mathbf{y}\mathbf{y}} & \mathbf{P}_{\mathbf{y}\mathbf{y}^{[2]}} & \mathbf{P}_{\mathbf{y}\mathbf{y}^{[3]}} & \dots \\ \mathbf{P}_{\mathbf{y}^{[2]}\mathbf{y}} & \mathbf{P}_{\mathbf{y}^{[2]}\mathbf{y}^{[2]}} & \mathbf{P}_{\mathbf{y}^{[2]}\mathbf{y}^{[2]}} & \dots \\ \mathbf{P}_{\mathbf{y}^{[3]}\mathbf{y}} & \mathbf{P}_{\mathbf{y}^{[3]}\mathbf{y}^{[2]}} & \mathbf{P}_{\mathbf{y}^{[3]}\mathbf{y}^{[3]}} & \dots \\ \vdots & \vdots & \vdots & \ddots\end{bmatrix} \tag{16}$$

and the augmented state-measurement cross-covariance matrix as

$$\mathbf{P}_{\mathbf{x}\mathcal{Y}} = \begin{bmatrix}\mathbf{P}_{\mathbf{x}\mathbf{y}} & \mathbf{P}_{\mathbf{x}\mathbf{y}^{[2]}} & \mathbf{P}_{\mathbf{x}\mathbf{y}^{[3]}} & \dots\end{bmatrix} \tag{17}$$

The deviations can be stacked as well, creating an augmented deviation vector.

$$d\mathcal{Y} = \begin{bmatrix}d\mathbf{y}^{\mathrm{T}} & d\mathbf{y}^{\{2\}\mathrm{T}} & d\mathbf{y}^{\{3\}\mathrm{T}} & \dots\end{bmatrix}^{\mathrm{T}} \tag{18}$$

The optimal polynomial update estimator becomes

$$\hat{\mathbf{x}} = \mathbb{E}\{\mathbf{x}\} + \mathbf{P}_{\mathbf{x}\mathcal{Y}}\mathbf{P}_{\mathcal{Y}\mathcal{Y}}^{-1} d\mathcal{Y} \tag{19}$$

where the product $\mathbf{P}_{\mathbf{x}\mathcal{Y}}\mathbf{P}_{\mathcal{Y}\mathcal{Y}}^{-1}$ is similar to the Kalman gain but realized with knowledge of the central moments of the distribution up to a selected order, improving accuracy.

III. The High-Order Polynomial Update Filter - HOPUF-ℓ-c

The new filtering technique has been developed in the Differential Algebra (DA) framework, using an algebra based on Taylor series expansion. An introduction to DA techniques and the Differential Algebra Core Engine (DACE2.0) software is omitted due to the large availability of references, such as [9], [19]–[21]. The High-Order Polynomial Update Filter (HOPUF-ℓ-c) algorithm is composed of three main parts: the prediction, the polynomial update, and the polynomial reduction. The two integers ℓ and c in HOPUF-ℓ-c refer to the order of the polynomial update and of the Taylor series expansion, respectively. Hence, HOPUF-1-1 is the extended Kalman filter and HOPUF-2-3 is a quadratic estimator with nonlinear functions approximated by their third order Taylor series expansion.

A. Prediction

Consider the following equations of motion and measurement equations describing a generic dynamic system:

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k) + \mathbf{v}_k \tag{20}$$

$$\mathbf{y}_{k+1} = \mathbf{h}(\mathbf{x}_{k+1}) + \mathbf{w}_{k+1} \tag{21}$$

where $\mathbf{f}(\cdot)$ is the dynamics function, $\mathbf{x}_k$ is the n-dimensional state of the system at time-step k, $\mathbf{y}_{k+1}$ is the m-dimensional measurement vector at time-step $k+1$, and $\mathbf{h}(\cdot)$ is the measurement function. The process noise and the measurement noise are assumed to be zero mean and uncorrelated, i.e., $\mathbb{E}\left\{\mathbf{v}_i \mathbf{w}_j^T\right\} = 0 \ \forall i,j$, with the autocovariance functions $\mathbb{E}\left\{\mathbf{v}_i \mathbf{v}_j^T\right\} = \mathbf{Q}_i \delta_{ij}$, and $\mathbb{E}\left\{\mathbf{w}_i \mathbf{w}_j^T\right\} = \mathbf{R}_i \delta_{ij}$ for all discrete time indexes i and j. It is assumed that the initial condition $\mathbf{x}_0$ and the noises are Gaussian, however, the nonlinear functions will make the distribution of $\mathbf{x}_k$ non-Gaussian for all $k > 0$. series expansion up to an arbitrary order. The state is initialized in the DA framework with a first order polynomial.

$$\mathbf{x}_0 = \mathbf{x}_0(\delta\mathbf{x}_0) = \hat{\mathbf{x}}_\mathbf{0} + \mathbf{S}_0 \delta\mathbf{x}_0 \tag{22}$$

where $\mathbf{x}_0 \sim \mathcal{N}(\hat{\mathbf{x}}_\mathbf{0}, \mathbf{P}_0)$, $\mathbf{S}_0\mathbf{S}_0^T = \mathbf{P}_0$, and the DA variable $\delta\mathbf{x}_0 = \mathbf{x}_0 - \hat{\mathbf{x}}_\mathbf{0}$ is therefore interpreted as Gaussian with zero mean and identity covariance matrix.

The propagation function is applied directly to the state polynomials $\mathbf{x}_k$, thus, the ith component of the predicted state vector is

$$\begin{aligned} x^-_{i,k+1} &= x^-_{i,k+1}(\delta\mathbf{x}_k) = f_i(\mathbf{x}_k(\delta\mathbf{x}_k)) = \\ &= f_i(\hat{\mathbf{x}}_k) + \sum_{r=1}^{c} \frac{1}{r!} \sum \frac{\partial^r f_i(\mathbf{x}_k)}{\partial x_1^{\gamma_1} \dots \partial x_n^{\gamma_r}} \delta x_{1,k}^{\gamma_1} \dots \delta x_{n,k}^{\gamma_r} \end{aligned} \tag{23}$$

where the integer c indicates the user-defined order of the Taylor expansion (the same numerical value specified in HOPUF-ℓ-c); the second summation is over all permutations of $\gamma_i \in \{1, \dots, n\}$ with $i \in \{1, \dots, r\}$. Therefore $\mathbf{x}^-_{k+1}(\delta\mathbf{x}_k)$ is a vector of polynomial functions that map the deviations $(\delta\mathbf{x}_k)$ into the distribution of the state at time $k+1$. The final step of the propagation is to add the process noise contribution.

TABLE I
Isserlis' moments of Gaussian $\mathcal{N}(0,1)$

exponent	0	1	2	3	4	5	6	7	8	…
coefficient	1	0	1	0	3	0	15	0	105	…

The process noise is introduced as a DA variable $\delta\mathbf{v}_k$, which once again is interpreted as a standard normal random vector

$$\mathbf{x}^-_{k+1}(\delta\mathbf{x}_k, \delta\mathbf{v}_k) \coloneqq \mathbf{x}^-_{k+1}(\delta\mathbf{x}_k) + \mathbf{T}_k \delta\mathbf{v}_k \tag{24}$$

where $\mathbf{v}_k = \mathbf{T}_k \delta\mathbf{v}_k$ and $\mathbf{T}_k\mathbf{T}_k^{\mathrm{T}} = \mathbf{Q}_k$.

The predicted measurements are evaluated similarly and the jth component is given by:

$$\begin{aligned} &y_{j,k+1} = y_{j,k+1}(\delta\mathbf{x}_k, \delta\mathbf{v}_k) = h_j(\mathbf{f}(\mathbf{x}^-_{k+1})) = \\ &= h_j(\mathbf{f}(\hat{\mathbf{x}}_{k+1})) + \sum_{r=1}^{c} \frac{1}{r!} \sum \frac{\partial^r h_j(\mathbf{x}^-_{k+1})}{\partial x_1^{\gamma_1} \dots \partial x_n^{\gamma_r} \partial v_1^{\gamma_1} \dots \partial v_n^{\gamma_r}} \cdot \\ &\cdot \delta x_{1,k}^{\gamma_1} \dots \delta x_{n,k}^{\gamma_r} \delta v_{1,k}^{\gamma_1} \dots \delta v_{n,k}^{\gamma_r} \end{aligned} \tag{25}$$

where, again, the second summation is over all permutations of $\gamma_i \in \{1, \dots, n\}$ with $i \in \{1, \dots, r\}$. Note that the Taylor series expansion is now in terms of both $\delta\mathbf{x}_k$ and $\delta\mathbf{v}_k$. The measurement noise is also introduced as a DA variable $\delta\mathbf{w}_{k+1}$, which is interpreted as a standard normal random vector

$$\mathbf{y}_{k+1}(\delta\mathbf{x}_k, \delta\mathbf{v}_k, \delta\mathbf{w}_{k+1}) \coloneqq \mathbf{y}_{k+1}(\delta\mathbf{x}_k, \delta\mathbf{v}_k) + \mathbf{U}_k \delta\mathbf{w}_{k+1} \tag{26}$$

where $\mathbf{w}_k = \mathbf{U}_k \delta\mathbf{w}_k$ and $\mathbf{U}_k\mathbf{U}_k^T = \mathbf{R}_k$.

The total number of DA variables is $2n+m$: n are used to map the state behavior $(\delta\mathbf{x}_k)$, n map the process noise $(\delta\mathbf{v}_k)$, and the remaining m map the measurement noise $(\delta\mathbf{w}_{k+1})$. All quantities of interest are represented by polynomial functions of standard normal random vectors. Therefore all expectations can be easily computed using Isserlis' formula to calculate central moments of standard normals [16].

B. Quadratic Update

Once the predicted state and measurement polynomials have been calculated, the next step in performing the quadratic update is evaluating the augmented Kalman gain with Equation (9). We start by calculating

$$\mathbf{y}^{[2]}_{k+1} = \mathbf{y}_{k+1} \otimes \mathbf{y}_{k+1} \tag{27}$$

where, once again, all the redundant components are eliminated.

The mean of each component of the predicted state is

$$\hat{x}^-_{i,k+1} = \mathbb{E}\left\{x^-_{i,k+1}\right\} \tag{28}$$

$$= f_i(\hat{\mathbf{x}}_k) + \sum_{r=1}^{c} \frac{1}{r!} \sum \frac{\partial^r f_i(\mathbf{x}_k)}{\partial x_1^{\gamma_1} \dots \partial x_n^{\gamma_r}} \mathbb{E}\left\{\delta x_{1,k}^{\gamma_1} \dots \delta x_{n,k}^{\gamma_r}\right\} \tag{29}$$

where the expectations are applied directly on the monomials of the series. Since the deviations have a Gaussian distribution with zero mean and identity covariance, the expected value operator just substitutes to each monomial the relative Isserlis' moment, according to Table I. As an illustrative example:

$\mathbb{E}\left\{a_i \delta x_1^2 \delta x_2^6 \delta x_4^4\right\} = 45 a_i$. Similar to Equation (28), the predicted means of the measurements are evaluated as:

$$\hat{\mathbf{y}}_{k+1} = \mathbb{E}\left\{\mathbf{y}_{k+1}\right\} \tag{30}$$

$$\hat{\mathbf{y}}^{[2]}_{k+1} = \mathbb{E}\left\{\mathbf{y}^{[2]}_{k+1}\right\} \tag{31}$$

The augmented measurement covariance matrix is calculated blockwise:

$$\mathbf{P}_{\mathcal{Y}\mathcal{Y}} = \begin{bmatrix} \mathbf{P}_{\mathbf{yy}} & \mathbf{P}_{\mathbf{y}\mathbf{y}^{[2]}} \\ \mathbf{P}_{\mathbf{y}^{[2]}\mathbf{y}} & \mathbf{P}_{\mathbf{y}^{[2]}\mathbf{y}^{[2]}} \end{bmatrix} \tag{32}$$

where, having removed the redundant components from the square of the measurements, the matrix is guaranteed to avoid any singularities. The matrix is symmetric and the three different blocks are calculated as:

$$\mathbf{P}_{\mathbf{yy}} = \mathbb{E}\left\{(\mathbf{y}_{k+1} - \hat{\mathbf{y}}_{k+1})(\mathbf{y}_{k+1} - \hat{\mathbf{y}}_{k+1})^{\mathrm{T}}\right\} \tag{33}$$

$$\mathbf{P}_{\mathbf{y}\mathbf{y}^{[2]}} = \mathbb{E}\left\{(\mathbf{y}_{k+1} - \hat{\mathbf{y}}_{k+1})(\mathbf{y}^{[2]}_{k+1} - \hat{\mathbf{y}}^{[2]}_{k+1})^{\mathrm{T}}\right\} \tag{34}$$

$$\mathbf{P}_{\mathbf{y}^{[2]}\mathbf{y}^{[2]}} = \mathbb{E}\left\{(\mathbf{y}^{[2]}_{k+1} - \hat{\mathbf{y}}^{[2]}_{k+1})(\mathbf{y}^{[2]}_{k+1} - \hat{\mathbf{y}}^{[2]}_{k+1})^{\mathrm{T}}\right\} \tag{35}$$

The augmented cross-covariance matrix between the state and measurement is evaluated blockwise as well:

$$\mathbf{P}_{\mathbf{x}\mathcal{Y}} = \begin{bmatrix} \mathbf{P}_{\mathbf{xy}} & \mathbf{P}_{\mathbf{x}\mathbf{y}^{[2]}} \end{bmatrix} \tag{36}$$

where each block is calculated in a similar way:

$$\mathbf{P}_{\mathbf{xy}} = \mathbb{E}\left\{(\mathbf{x}^{-}_{k+1} - \hat{\mathbf{x}}^{-}_{k+1})(\mathbf{y}_{k+1} - \hat{\mathbf{y}}_{k+1})^{\mathrm{T}}\right\} \tag{37}$$

$$\mathbf{P}_{\mathbf{x}\mathbf{y}^{[2]}} = \mathbb{E}\left\{(\mathbf{x}^{-}_{k+1} - \hat{\mathbf{x}}^{-}_{k+1})(\mathbf{y}^{[2]}_{k+1} - \hat{\mathbf{y}}^{[2]}_{k+1})^{\mathrm{T}}\right\} \tag{38}$$

From these quantities it is now possible to use Equation (9) to calculate the Kalman gain associated with the augmented system:

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_1 & \mathbf{K}_2 \end{bmatrix} = \mathbf{P}_{\mathbf{x}\mathcal{Y}} \mathbf{P}^{-1}_{\mathcal{Y}\mathcal{Y}} \tag{39}$$

Denote $\tilde{\mathbf{y}}_k$ as the outcome of random vector $\mathbf{y}_k$, i.e., the numerical value of the measurement from the sensors, and its square as

$$\tilde{\mathbf{y}}^{[2]}_k = \tilde{\mathbf{y}}_k \otimes \tilde{\mathbf{y}}_k \tag{40}$$

The updated distribution of the state and the posterior estimate are given by

$$\mathbf{x}^{+}_{k+1}(\delta\mathbf{x}_k, \delta\mathbf{v}_k, \delta\mathbf{w}_{k+1}) = \tag{41}$$
$$= \mathbf{x}^{-}_{k+1}(\delta\mathbf{x}_k, \delta\mathbf{v}_k) + \mathbf{K}\begin{bmatrix} \tilde{\mathbf{y}} - \mathbf{y}^{-}_{k+1}(\delta\mathbf{x}_k, \delta\mathbf{v}_k, \delta\mathbf{w}_{k+1}) \\ \tilde{\mathbf{y}}^{[2]} - \mathbf{y}^{[2]-}_{k+1}(\delta\mathbf{x}_k, \delta\mathbf{v}_k, \delta\mathbf{w}_{k+1}) \end{bmatrix}$$

$$\hat{\mathbf{x}}^{+}_{k+1} = \mathbb{E}\left\{\mathbf{x}^{+}_{k+1}\right\} = \hat{\mathbf{x}}^{-}_{k+1} + \mathbf{K}\begin{bmatrix} \tilde{\mathbf{y}}_{k+1} - \hat{\mathbf{y}}_{k+1} \\ \tilde{\mathbf{y}}^{[2]}_{k+1} - \hat{\mathbf{y}}^{[2]}_{k+1} \end{bmatrix} \tag{42}$$

Note that the updated polynomials are a function of the state deviations, the process noise and the measurement noise. Moreover, since the order of $\mathbf{x}^{-}_{k+1}(\delta\mathbf{x}_k, \delta\mathbf{v}_k)$ is c, then, because of the quadratic update, $\mathbf{x}^{+}_{k+1}(\delta\mathbf{x}_k, \delta\mathbf{v}_k, \delta\mathbf{w}_{k+1})$ is a polynomial of order $2c$. The higher the polynomial order, the higher the number of moments to be calculated with Table I. The covariance matrix is obtained as

$$\mathbf{P}_{\mathbf{xx},k+1} = \mathbb{E}\left\{(\mathbf{x}^{+}_{k+1} - \hat{\mathbf{x}}^{+}_{k+1})(\mathbf{x}^{+}_{k+1} - \hat{\mathbf{x}}^{+}_{k+1})^{\mathrm{T}}\right\} \tag{43}$$

Thus, starting form a polynomial of order c at the beginning of each filter's iteration, equation (43) employs a polynomial of order $4c$ in order to calculate the state covariance matrix. Equations (42) and (43) are outputs calculated for downstream users; the recursive algorithm does not employ the actual value of the state estimate and its uncertainties to start the next iteration; the coefficients of the Taylor series representation updated polynomial $\mathbf{x}^{+}_{k+1}$ are stored and contain all the information needed for the recursion.

C. Polynomial Least Squares Reduction

The updated state $\mathbf{x}^{+}_{k+1}(\delta\mathbf{x}_k, \delta\mathbf{v}_k, \delta\mathbf{w}_{k+1})$ is a $2c$th order polynomial in $2n+m$ variables and it describes the (typically non-Gaussian) posterior probability density function of the state. Without taking any further action and starting the next filter iteration from $\mathbf{x}^{+}_{k+1}$ to calculate $\mathbf{x}^{+}_{k+2}$, the order of the series will double to $4c$ and the number of independent variables will increase by two: the process noise associated with the subsequent propagation and the measurement noise associated with the next measurement.

To keep the recursive algorithm tractable, it is therefore desirable to: *i.* reduce the polynomial order back to c, and *ii.* reduce the number of variables back to n. In this section we introduce a new methodology to achieve these two objectives.

The proposed approach preserves the non Gaussian nature of the posterior distribution. The idea is to exactly match the zeroth and first order components of the series representation of $\mathbf{x}^{+}_{k+1}$ and to approximate the higher order elements with least squares.

The posterior is a nonlinear (polynomial) function of three Gaussian random vectors ($\delta\mathbf{x}_k$, $\delta\mathbf{v}_k$, $\delta\mathbf{w}_{k+1}$), and the goal is to accurately approximate it as a polynomial function of a single Gaussian random vector of size n. The linear part of the Taylor series (zeroth and first order) is a linear combination of Gaussian random vectors and it is represented exactly as a single Gaussian; while higher than first order contributions are approximated by minimizing a least-squares approximation error.

We start by dividing the polynomials into their linear part (constant plus first order) and the remaining higher order terms. Thus at time t_{k+1}

$$\mathbf{x}^{+}_{k+1} = \mathbf{x}_{L,k+1} + \mathbf{x}_{H,k+1} \tag{44}$$

where $\mathbf{x}_{L,k+1}$ indicates the linear part of the Taylor series and $\mathbf{x}_{H,k+1}$ represents the nonlinearities. For linear and Gaussian systems, $\mathbf{x}_{H,k+1}$ is identically zero and $\mathbf{x}_{L,k+1}$ is the posterior distribution found with the Kalman filter equations. For nonlinear systems, $\mathbf{x}_{H,k+1}$ is non-zero and $\mathbf{x}_{L,k+1}$ is not the Kalman filter posterior, however $\mathbf{x}_{L,k+1}$ is still Gaussian because it is the linear combination of three Gaussian random vectors. The mean and covariance matrix of the linear polynomials are readily found in the DA framework

$$\hat{\mathbf{x}}_{L,k+1} = \mathbb{E}\left\{\mathbf{x}_{L,k+1}\right\} \tag{45}$$

$$\mathbf{P}_{L,k+1} = \mathbb{E}\left\{(\mathbf{x}_{L,k+1} - \hat{\mathbf{x}}_{L,k+1})(\mathbf{x}_{L,k+1} - \hat{\mathbf{x}}_{L,k+1})^{\mathrm{T}}\right\} \tag{46}$$

$$\mathbf{S}_{k+1}\mathbf{S}^{\mathrm{T}}_{k+1} = \mathbf{P}_{L,k+1} \tag{47}$$

Defining $\delta\mathbf{x}_{k+1} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ we can represent $\mathbf{x}_{L,k+1}$ exactly:

$$\mathbf{x}_{L,k+1} = \mathbf{S}_{k+1}\delta\mathbf{x}_{k+1} + \hat{\mathbf{x}}_{L,k+1} \tag{48}$$

The goal is to create a new series approximation of $\mathbf{x}^+_{k+1}$, which is only a function of $\delta\mathbf{x}_{k+1}$ rather than the three variables $(\delta\mathbf{x}_k, \delta\mathbf{v}_k, \delta\mathbf{w}_{k+1})$. To calculate the coefficients of the series, we solve a least-squares problem to best fit samples.

We draw N independent and identically distributed samples from a $2n + m$ Gaussian distribution with zero mean and identity covariance matrix in order to have N independent samples of $\delta\mathbf{x}_k, \delta\mathbf{v}_k, \delta\mathbf{w}_{k+1}$. Samples are denoted with a superscript (j), $j \in [1,\, N]$. The polynomials are then evaluated at the samples, resulting in N independent realizations of the posterior. We calculate N samples of $\mathbf{x}^+_{k+1}$, $\mathbf{x}_{L,k+1}$ and $\delta\mathbf{x}_{k+1}$:

$$\mathbf{x}^{+(j)}_{k+1} = \mathbf{x}^+_{k+1}(\delta\mathbf{x}^{(j)}_k, \delta\mathbf{v}^{(j)}_k, \delta\mathbf{w}^{(j)}_{k+1}) \tag{49}$$

$$\mathbf{x}^{(j)}_{L,k+1} = \mathbf{x}_{L,k+1}(\delta\mathbf{x}^{(j)}_k, \delta\mathbf{v}^{(j)}_k, \delta\mathbf{w}^{(j)}_{k+1}) \tag{50}$$

$$\delta\mathbf{x}^{(j)}_{k+1} = \mathbf{S}^{-1}_{k+1}\big(\mathbf{x}^{(j)}_{L,k+1} - \hat{\mathbf{x}}_{L,k+1}\big) \tag{51}$$

where Equation (51) scales and centers the realizations from the linear polynomials, such that their distribution is a standard Gaussian. The ith component of the new series is

$$\begin{aligned} x^+_{i,k+1} &= x^+_{i,k+1}(\delta\mathbf{x}_{k+1}) = \\ &= a_{i,0} + \sum_{r=1}^{c} \frac{1}{r!} \sum_{\ell} a_{i,r\ell}\ \delta x^{\gamma_1}_{1,k+1} \dots \delta x^{\gamma_r}_{n,k+1} \end{aligned} \tag{52}$$

where the second summation is over all permutations of $\gamma_i \in \{1, \dots, n\}$ with $i \in \{1, \dots, r\}$, describing each possible monomial through combinatorial mathematics.

The above equation is linear in the coefficients, therefore it can be rewritten as

$$x^+_{i,k+1} = \boldsymbol{\Delta}_{k+1}\mathbf{a}_i \tag{53}$$

where vector $\mathbf{a}_i$ contains the coefficients of the series we are solving for. We can now use the samples of $\mathbf{x}^+_{i,k+1}$ and $\delta\mathbf{x}_{k+1}$ to compute a least square estimate of $\mathbf{a}_i$.

The algorithm is now ready to start the next iteration with a new polynomial propagation.

IV. Numerical Example

The performance of the proposed filter is tested in a Lorenz96 example [17] [18]. The dynamic of the system is

$$\frac{dx_i(t)}{dt} = x_{i-1}(t)(x_{i+1}(t) - x_{i-2}(t)) - x_i(t) + F + \nu_i(t) \tag{54}$$

where $x_i(t)$, $i = 1...4$, are the four components of the state vector $\mathbf{x}(t)$ is used. The following conventions are used: $x_{-1}(t) = x_{n-1}(t)$, $x_0(t) = x_n(t)$, and $x_1(t) = x_{n+1}(t)$. The term F is a constant external force and it is chosen equal to eight in order to introduce a chaotic behavior in the system. The initial condition is assumed to be Gaussian, with mean $\hat{\mathbf{x}} = \begin{bmatrix} F & F & F + 0.01 & F \end{bmatrix}^T$ and diagonal covariance matrix, with the same standard deviation for each component of the state: $\sigma_{\mathbf{x}} = 10^{-3}$. The process noise is assumed to be Gaussian and uncorrelated among states, with known standard deviation $\sigma_{\boldsymbol{\nu}} = 10^{-3}$.

Measurements are acquired each time-step according to the following model:

$$\mathbf{y}_k = \mathbf{H}_{i,j}\mathbf{x}_k + \boldsymbol{\mu}_k, \quad \mathbf{H} = \begin{cases} 1 & j = 2i - 1 \\ 0 & \text{otherwise} \end{cases} \tag{55}$$

with $i = \{1, 2\}$ and $j = \{1, 2, 3, 4\}$. In other words, the measurements are linear and the components of the state vector with odd indexes measured. The measurement noises, uncorrelated from the process noise and between each other, are assumed to be Gaussian as well with standard deviation $\sigma_{\boldsymbol{\mu}} = 10^{-2}$. The dynamics are propagated at 2 Hz with a Runge-Kutta 7-8 integrator.

Figure 1 shows the Monte Carlo analysis performed with HOPUF-2-2 on the presented problem. Once again the notation HOPUF-ℓ-c indicates the selected truncation order of the Taylor series expansion c and the order of the update ℓ. The value $N = 1000$ samples is chosen for the least squares problem and the number of Monte Carlo runs is also 1000. Figure 1 shows the 1000 realizations of the estimation error (gray lines), evaluated as distance between the estimate and the true state, for each one of the four state components. The algorithms' predicted estimation error standard deviation of each state is reported in green lines (3σ values), calculated as the square roots of the diagonal terms of $\mathbf{P}_{\mathbf{xx}}$ in Equation (43). The actual performance of the system is assessed by the sample standard deviation of the Monte Carlo estimation errors, represented in the figure as blue lines (3σ values). The consistency of HOPUF-2-2 is established by the close values of the estimated and effective standard deviations, as shown by the overlapping of the green and blue lines. Finally, the black lines are the sample means of the estimation errors at each time-step and they demonstrate the unbiased nature of the filter, as expected from the theory of MMSE estimation.

The benefits of the quadratic update can be appreciated when comparing HOPUF-2-2 with linear estimators, such as the EKF, the UKF [6], [22] and the DAHO-k [9]. DAHO-k is a linear estimator that approximates the nonlinear functions with their kth order Taylor series representation. The EKF performs a simple linearization of the equation of motion and, in the presence of high nonlinearities such as in the Lorenz96 problem, fails to estimate the state of the system (therefore it is not reported in the figures). The UKF and the DAHO-k, on the other hand, are linear estimators that better account for the system's nonlinearities and achieve better performance than the EKF [19]. Figure 2 compares HOPUF-2-2, with DAHO-2 and the UKF. DAHO-k is a well suited test bench for higher-order linear estimators [15], [20].

Figure 2 has six lines. The three continuous lines represent the filter's own estimate of accuracy in terms of the predicted estimation error standard deviation, calculated from the covariance matrix as the square root of its trace: $\bar{\sigma} = \sqrt{\text{tr}(\mathbf{P}_{\mathbf{xx}})}$. The three dashed lines represent the effective error standard deviation derived from the Monte Carlo analysis (1000 runs). The filter is consistent when it predicts its own uncertainty, i.e., when the continuous and dashed lines coincide. The figure shows how the quadratic update (blue lines) reduces the system uncertainties, improving accuracy especially in the

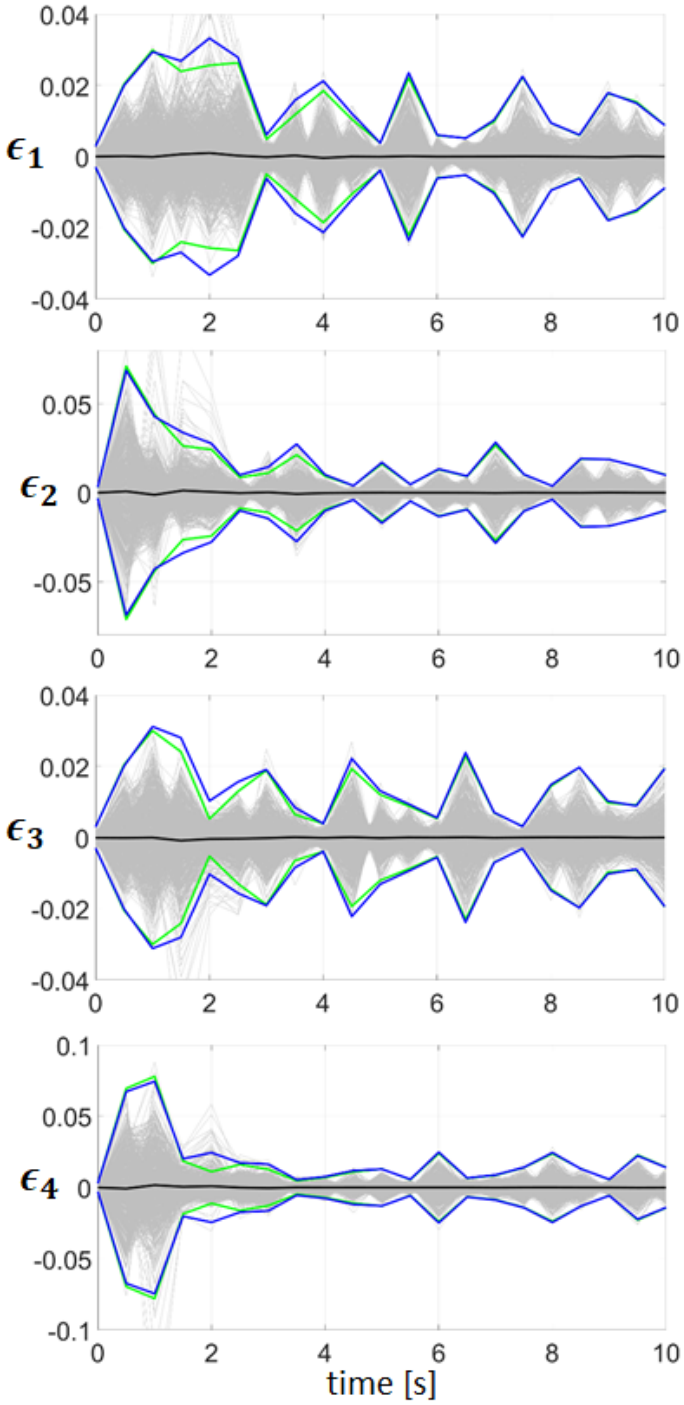

Fig. 1. 1000 runs Monte Carlo performance test for the HOPUF-2-2, $N = 1000$.

first steps of the simulation, during the transient behavior. HOPUF-2-2 reaches steady state quickly and the filter avoids overconfidence. The linear update of DAHO-2 (red lines), on the other hand, has a much slower and less accurate transient response but it is consistent and eventually converges. Once steady-state is reached, the nonlinearities cease to dominate and the quadratic update no longer significantly outperforms the linear update. The UKF (green lines) is inconsistent as the filter's prediction of its own estimation error does not match

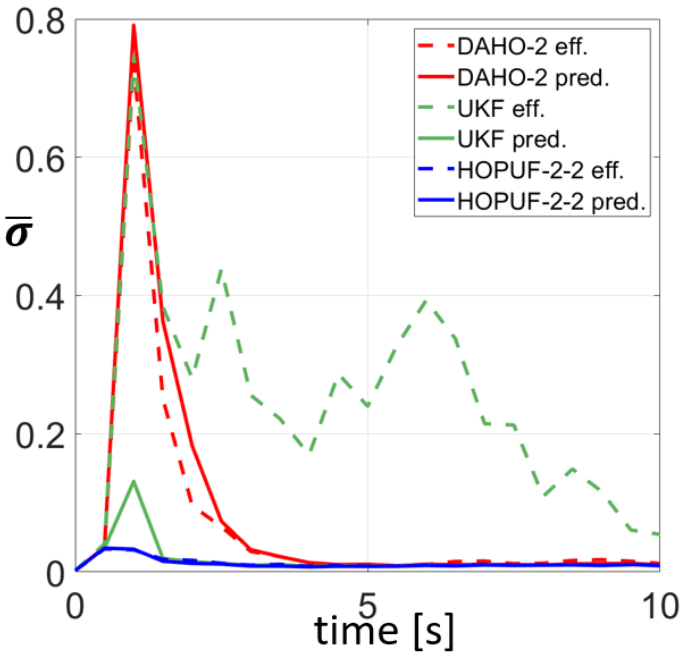

Fig. 2. Covariance comparison: quadratic vs. linear estimator. Solid lines are the filters own predictions, dashed lines represent the sample standard deviation from Monte Carlo.

the actual behavior, which, as shown by the dashed green line, does not settle to the same accuracy level as HOPUF-2-2 and DAHO-2.

For the Lorenz96 example considered, the new quadratic estimator exhibits more robustness than the linear estimators tested. Increasing the measurement noise levels or decreasing the measurement frequency causes divergence of the UKF and DAHO-2 as tested, while the consistency of HOPUF-2-2 is not affected by those changes (the estimation accuracy on the other hand degrades, because of fewer or less precise measurements).

V. Conclusions

A new estimator with quadratic update has been presented. Unlike prior techniques, the proposed approach does not require storing higher order central moments of the state's distribution. The new technique accounts for the nonlinearities of the system both in the prediction and in the update step by approximating the distributions resulting from nonlinear transformations as polynomial functions of Gaussian random vectors. The new approach is easily expandable to any order of the polynomial update. For systems other than linear and Gaussian (where a linear update is globally optimal) the higher the order of the polynomial update, the more precise the resulting state estimate.

The proposed algorithm can be interpreted as an expansion of Gaussian filters. Gaussian filters approximate uncertainties as Gaussian. By representing the state uncertainty with an arbitrary polynomial function of a Gaussian random vector, it is possible to better approximate the shape of distributions undergoing nonlinear dynamics and measurements.

ACKNOWLEDGMENT

This work was sponsored in part by the Air Force Office of Scientific Research under grant number FA9550-18-1-0351

REFERENCES

[1] Rudolf E. Kalman, "A New Approach to Linear Filtering and Prediction Problems, Journal of Basic Engineering", March, doi:10.1115/1.3662552, No. Series D, pp. 35–45, Vol. 82, 1960

[2] Rudolf E. Kalman and Richard S. Bucy, "New Results in Linear Filtering and Prediction", Journal of Basic Engineering, March, doi:10.1115/1.3658902, No. Series D, pp. 95–108, Vol. 83, 1961

[3] Arthur Gelb, "Applied Optimal Estimation", Massachusetts Institute of Technology, ISBN:9780262200271, Cambridge, MA, The MIT press, 1974

[4] J. Junkins and P. Singla, "How Nonlinear is it? A tutorial on Nonlinearity of Orbit and Attitude Dynamics", Journal of the Astronautical Sciences, No. 1-2, pp. 7–60, Vol. 52, 2004

[5] Simon J. Julier and Jeffrey K. Uhlmann and Hugh F. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators", IEEE Transactions on Automatic Control, March, doi: 10.1109/9.847726, No. 3, pp. 477–482, Vol. 45, 2000

[6] Simon J. Julier and Jeffrey K. Uhlmann, "Unscented filtering and nonlinear estimation", Proceedings of the IEEE, March, doi: 10.1109/JPROC.2003.823141, No. 3, pp. 401–422, Vol. 92, 2004

[7] Ryan S. Park and Daniel J. Scheeres, "Nonlinear mapping of Gaussian statistics: theory and applications to spacecraft trajectory design", Journal of Guidance, Control, and Dynamics, doi: 10.2514/1.20177, No. 6, pp. 1367–1375, American Institute of Aeronautics and Astronautics, Vol. 29, 2006

[8] Manoranjan Majji and John Junkins and James Turner, "A High Order Method for Estimation of Dynamic Systems", The Journal of the Astronautical Sciences, doi: 10.1007/BF03256560, No. 3, pp. 401–440, Vol. 56, 2008

[9] Monica Valli and Roberto Armellin and Pierluigi Di Lizia and Michéle Lavagna, "Nonlinear mapping of uncertainties in celestial mechanics", Journal of Guidance, Control, and Dynamics, doi: 10.2514/1.58068, No. 1, pp. 48–63, Vol. 36,2012

[10] H. W. Sorenson and D. L. Alspach, "Recursive Bayesian Estimation Using Gaussian Sums", Automatica, July, doi:10.1016/0005-1098(71)90097-5, No. 4, pp. 465–479, Vol. 7, 1971

[11] D. Alspach and H. Sorenson, "Nonlinear Bayesian estimation using Gaussian sum approximations", IEEE Transactions on Automatic Control, August, doi: 10.1109/SAP.1970.270017, No. 4, pp. 439–448, Vol. 17, 1972

[12] Gordon, N. J., Salmond, D. J., and Smith, A. F. M., "Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation, Radar and Signal Processing, IEE Proceedings F, Vol. 140, No. 2, April 1993, pp. 107113. doi:10.1049/ip-f-2.1993.0015

[13] A. De Santis and A. Germani and M. Raimondi, "Optimal quadratic filtering of linear discrete-time non-Gaussian systems", doi:10.1109/9.400478, IEEE Transactions on Automatic Control, No. 7, pp. 1274-1278, Vol. 40, 1995, http://dx.doi.org/10.1109/9.400478.

[14] F. Carravetta and A. Germani and M. Raimondi, "Polynomial filtering of discrete-time stochastic linear systems with multiplicative state noise", IEEE Transactions on Automatic Control, doi:10.1109/9.618240, No. 8, pp. 1106-1126, Vol. 42, 1997.

[15] Simone Servadio and Renato Zanetti, "Recursive Polynomial Minimum Mean-Square Error Estimation with Applications to Orbit Determination", Journal of Guidance, Control, and Dynamics, pp. 1–16, 2020, American Institute of Aeronautics and Astronautics

[16] Isserlis, L., "On a formula for the product-moment coefficient of any order of a normal frequency distribution in any number of variables", Biometrika, Vol. 12, No. 1/2, pp. 134–139, 1918, JSTOR

[17] Dilshad Raihan and Suman Chakravorty, "Particle Gaussian mixture filters-I", Automatica, Vol. 98, pp. 331–340, 2018, Elsevier

[18] Ott, Edward and Hunt, Brian R and Szunyogh, Istvan and Zimin, Aleksey V and Kostelich, Eric J and Corazza, Matteo and Kalnay, Eugenia and Patil, DJ and Yorke, James A, "A local ensemble Kalman filter for atmospheric data assimilation", Tellus A: Dynamic Meteorology and Oceanography, Vol. 56, No. 5, pp. 415–428, 2004, Taylor & Francis

[19] Simone Servadio, "High Order Filters For Relative Pose Estimation Of An Uncooperative Target", Politecnico di Milano, Master Thesis, 2017, Milan, Italy, 20156.

[20] Francesco Cavenago and Pierluigi Di Lizia and Mauro Massari and Simone Servadio and Alexander Wittig, "DA-based nonlinear filters for spacecraft relative state estimation", 2018 Space Flight Mechanics Meeting, doi: 10.2514/6.2018-1964, p. 1964, 2018

[21] Di Lizia, Pierluigi and Massari, Mauro and Cavenago, Francesco, "Assessment of onboard DA state estimation for spacecraft relative navigation", Final report, ESA , 2017

[22] JK Uhlmann and J. Julier and HF. Durrant-Whyte , "A new approach for the nonlinear transformation of means and covariances in linear filters", IEEE Trans. Automatic Control, 1996