AAS 19-875

MAXIMUM A POSTERIORI ESTIMATION OF HAMILTONIAN SYSTEMS WITH HIGH ORDER SERIES EXPANSIONS

Simone Servadio*, Renato Zanetti† and Roberto Armellin‡

This paper presents a new approach to Maximum A Posteriori (MAP) estimation. Representing probability density functions through Taylor series expansions and using Differential Algebra techniques, this work proposes to derive the MAP estimate directly from high order polynomials. The new method is applied to the nonlinear Orbit Determination problem.

INTRODUCTION

Maximum A Posteriori (MAP) estimation is one where the optimal estimate of the state is given by the peak of the probability density function conditioned on the measurements; which can be calculated using Baye's rules. In the well-known linear and Gaussian case, the posterior distribution remains Gaussian at all times and it can be fully described by its mean and covariance matrix, which are calculated using the Kalman filter equations [1, 2]. For this type of problems, Linear Minimum Mean Square Error (LMMSE) and MAP estimation reduce to the same estimation problem. Most problems of practical interest to aerospace engineering applications, such as obit determination [3], contain nonlinearities in the dynamics which inevitably result in non-Gaussian posterior distribution; as such, LMMSE and MAP produce different estimators and for neither a closed form solution to this typically unavailable.

Many techniques have been developed to deal with the nonlinear estimation problem. The most exploited estimation algorithm is the Extended Kalman Filter (EKF) [4]. The EKF is based on linearization the dynamics and measurement equations around the most current estimate and applying the Kalman filter equations to the linearized system. In problems with high nonlinearities, such as orbit determination, the linearization assumption may fail to give a valid estimate [5]. Other approaches that account for system nonlinearities have been implemented, such as the Unscented Kalman filter (UKF) [6, 7]. Through the true nonlinear propagation of carefully chosen sample points, the UKF can achieve superior performance with respect to the EKF.

Park and Scheeres [8, 9] developed higher order filters that uses state transition tensors (STT) in the filter's prediction step that fully relies on the non linear mapping of the means and covariance matrices to more accurately represent the uncertainty with respect to the EKF. Valli et al.[10] recreated Park and Scheeres's work using Differential Algebra (DA), thus eliminating the need to evaluate high order tensors directly.

This works introduces a new nonlinear filter that approximates the probability distribution functions through their Taylor expansion series. The time propagation and measurements update phases of the filter operate directly on the probability density function (PDF). The MAP estimate is then found by polynomial maximization. The Taylor series are kept centered on the most recent estimate by polynomial evaluation of the PDF.

MAXIMUM A POSTERIORI ESTIMATION

Let $p(\mathbf{x})$ be the probability density function (PDF) of a random vector $\mathbf{x}$ that it is not directly sample-able and that we are interested in estimating; and let $\mathbf{y}$ be another random vector, called the measurement, that

*Ph.D. Student, Aerospace Engineering and Engineering Mechanics, The University of Texas at Austin
†Assistant Professor, Aerospace Engineering and Engineering Mechanics, The University of Texas at Austin
‡Senior Lecturer, Aerospace Engineering and Engineering Mechanics University of Surrey, Guildford, England

possesses a joint distribution with $\mathbf{x}$ and that it is available to be sampled. The conditional distribution of $\mathbf{x}$ given $\mathbf{y}$, denoted as $p_{\mathbf{x}|\mathbf{y}}(\mathbf{x}|\mathbf{y})$ is given by Baye's rule

$$p_{\mathbf{x}|\mathbf{y}}(\mathbf{x}|\mathbf{y}) = \frac{p_{\mathbf{y}|\mathbf{x}}(\mathbf{y}|\mathbf{x})\, p_{\mathbf{x}}(\mathbf{x})}{p_{\mathbf{y}}(\mathbf{y})} \tag{1}$$

where $p_{\mathbf{x}|\mathbf{y}}(\mathbf{x}|\mathbf{y})$ is called posterior distribution, $p_{\mathbf{x}}(\mathbf{x})$ prior distribution, and $p_{\mathbf{y}|\mathbf{x}}(\mathbf{y}|\mathbf{x}) = L(\mathbf{x}|\mathbf{y})$ likelihood function.

The Maximum A Posteriori (MAP) estimate $\hat{\mathbf{x}}$ is the, possibly not unique, value of $\mathbf{x}$ that maximizes the posterior distribution

$$\hat{\mathbf{x}} = \max_{\mathbf{x}} p_{\mathbf{x}|\mathbf{y}}(\mathbf{x}|\mathbf{y}) \tag{2}$$

We notice that the denominator of Eq. (1) does not contain $\mathbf{x}$, therefore we obtain the same answer if we instead maximize the un-normalized posterior distribution, i.e. the joint distribution evaluated at the measurement value: $p_{\mathbf{x},\mathbf{y}}(\mathbf{x},\mathbf{y}) = p_{\mathbf{y}|\mathbf{x}}(\mathbf{y}|\mathbf{x})\, p_{\mathbf{x}}(\mathbf{x})$.

$$\hat{\mathbf{x}} = \max_{\mathbf{x}} p_{\mathbf{x},\mathbf{y}}(\mathbf{x},\mathbf{y}) \tag{3}$$

Rather than maximizing the joint distribution directly, we might decide to maximize its logarithm:

$$\hat{\mathbf{x}} = \max_{\mathbf{x}} \Big(\log p_{\mathbf{x}}(\mathbf{x}) + \Lambda(\mathbf{x}|\mathbf{y}) \Big) \tag{4}$$

where $\Lambda(\mathbf{x}|\mathbf{y}) = \log L(\mathbf{x}|\mathbf{y})$ is the log-likelihood function.

UNCERTAINTY EVOLUTION OF HAMILTONIAN SYSTEMS

Let's look at the evolution of the following dynamic system

$$\dot{\mathbf{x}}(t) = \mathbf{f}(t,\mathbf{x}); \qquad \mathbf{x}(t_0) = \mathbf{x}_0 \tag{5}$$

this is an ODE, and let's denote its solution flow as $\mathbf{x}(t) = \boldsymbol{\phi}(\mathbf{x}_0; t, t_0)$; clearly

$$\frac{\partial}{\partial t}\boldsymbol{\phi}(\mathbf{x}_0; t, t_0) = \mathbf{f}(t, \boldsymbol{\phi}(\mathbf{x}_0; t, t_0)) \tag{6}$$

The solution flow has the following properties

$$\boldsymbol{\phi}(\mathbf{x}_0; t_0, t_0) = \mathbf{x}_0 \tag{7}$$
$$\boldsymbol{\phi}(\boldsymbol{\phi}(\mathbf{x}_0; t, t_0); t_0, t) = \mathbf{x}_0 \tag{8}$$

this second property just states that the functional inverse is obtained by inverting the time arguments (exactly like the state transition matrix)

$$\mathbf{x}_f = \boldsymbol{\phi}(\mathbf{x}_0; t_f, t_0) \tag{9}$$
$$\mathbf{x}_0 = \boldsymbol{\phi}(\mathbf{x}_f; t_0, t_f) \tag{10}$$

From

$$\boldsymbol{\phi}(\boldsymbol{\phi}(\mathbf{x}_0; t, t_0); t_0, t) = \mathbf{x}_0 \tag{11}$$

we have that

$$\begin{aligned} \frac{d}{dt}\boldsymbol{\phi}(\boldsymbol{\phi}(\mathbf{x}_0; t, t_0); t_0, t) &= \mathbf{0} \\ &= \frac{\partial}{\partial t}\boldsymbol{\phi}(\boldsymbol{\phi}(\mathbf{x}_0; t, t_0); t_0, t) + \frac{\partial}{\partial \xi}\boldsymbol{\phi}(\boldsymbol{\xi}; t_0, t)\frac{\partial}{\partial t}\boldsymbol{\phi}(\mathbf{x}_0; t, t_0) \end{aligned} \tag{12}$$

or

$$\frac{\partial}{\partial t}\boldsymbol{\phi}(\mathbf{x};t_0,t) = -\left.\frac{\partial}{\partial \xi}\boldsymbol{\phi}(\boldsymbol{\xi};t_0,t)\right|_{\boldsymbol{\xi}=\mathbf{x}} \mathbf{f}(t,\mathbf{x}) \tag{13}$$

Taking the initial condition $\mathbf{x}_0$ as a random vector with PDF $p_{\mathbf{x}_0}(\mathbf{x})$ the PDF evolves in time according to the Fokker-Plank-Kolmorov (FPK) equation without diffusion

$$\begin{aligned}\frac{\partial p_{\mathbf{x}(t)}(\mathbf{x},t)}{\partial t} &= -\sum_{i=1}^{n}\frac{\partial\Big(f_i(\mathbf{x},t)\; p_{\mathbf{x}(t)}(\mathbf{x},t)\Big)}{\partial x_i} \\ &= -\sum_{i=1}^{n} f_i(\mathbf{x},t)\;\frac{\partial p_{\mathbf{x}(t)}(\mathbf{x},t)}{\partial x_i} - \sum_{i=1}^{n} p_{\mathbf{x}(t)}(\mathbf{x},t)\;\frac{\partial f_i(\mathbf{x},t)}{\partial x_i} \\ &= -\frac{\partial p_{\mathbf{x}(t)}(\mathbf{x},t)}{\partial \mathbf{x}}\;\mathbf{f}(\mathbf{x},t) - p_{\mathbf{x}(t)}(\mathbf{x},t)\;\operatorname{trace}\left(\frac{\partial \mathbf{f}(\mathbf{x},t)}{\partial \mathbf{x}}\right)\end{aligned} \tag{14}$$

For Hamiltonian systems the Jacobian of $\mathbf{f}$ is traceless and we are left with

$$\frac{\partial p_{\mathbf{x}(t)}(\mathbf{x},t)}{\partial t} + \frac{\partial p_{\mathbf{x}(t)}(\mathbf{x},t)}{\partial \mathbf{x}}\;\dot{\mathbf{x}} = \frac{dp_{\mathbf{x}(t)}(\mathbf{x},t)}{dt} = 0 \tag{15}$$

hence for all t

$$p_{\mathbf{x}(t)}(\boldsymbol{\phi}(\mathbf{x}_0;t,t_0),t) = p_{\mathbf{x}(t_0)}(\mathbf{x}_0,t_0) = p_{\mathbf{x}_0}(\mathbf{x}_0) \tag{16}$$

or

$$p_{\mathbf{x}(t)}(\mathbf{x},t) = p_{\mathbf{x}(t_0)}(\boldsymbol{\phi}(\mathbf{x};t_0,t),t_0) = p_{\mathbf{x}_0}(\boldsymbol{\phi}^{-1}(\mathbf{x};t,t_0)) \tag{17}$$

This is easy to show since

$$\frac{dp_{\mathbf{x}(t)}(\mathbf{x},t)}{dt} = \frac{dp_{\mathbf{x}_0}(\boldsymbol{\phi}(\mathbf{x};t_0,t))}{dt} = \frac{\partial p_{\mathbf{x}_0}(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}}\left(\frac{\partial \boldsymbol{\phi}(\mathbf{x};t_0,t)}{\partial t} + \frac{\partial \boldsymbol{\phi}(\mathbf{x};t_0,t)}{\partial \mathbf{x}}\dot{\mathbf{x}}\right) \tag{18}$$

But the term in parenthesis is identically zero because of Eq. (12), so the FPK equation is satisfied.

MAP ESTIMATION OF HAMILTONIAN SYSTEMS WITH DIFFERENTIAL ALGEBRA

Let the initial state $\mathbf{x}(t_0)$ be distributed as a Gaussian random vector with mean $\hat{\mathbf{x}}_0$ and covariance matrix $\mathbf{P}_0$. We initialize the algorithm with

$$\log p_{\mathbf{x}_0}(\mathbf{x}_0) = -\frac{1}{2}(\mathbf{x}_0-\hat{\mathbf{x}}_0)^{\mathrm{T}}\mathbf{P}_0^{-1}(\mathbf{x}_0-\hat{\mathbf{x}}_0) + C_0 \tag{19}$$

we can safely ignore the constant C_0 and we recognize that the above equation is a quadratic polynomial function. Let's define

$$d\mathbf{x}_0 = \mathbf{x}_0 - \hat{\mathbf{x}}_0 \tag{20}$$

and

$$\Xi_0(d\mathbf{x}_0) = \log p_{\mathbf{x}_0}(\hat{\mathbf{x}}_0 + d\mathbf{x}_0) - C_0 = -\frac{1}{2}d\mathbf{x}_0^{\mathrm{T}}\mathbf{P}_0^{-1}d\mathbf{x}_0 \tag{21}$$

C_0 is subtracted because all normalizing constants of the PDFs are removed.

Time Propagation

Assume $\mathbf{x}$ is a Hamiltonian system evolving as

$$\dot{\mathbf{x}}(t) = \mathbf{f}(t, \mathbf{x})$$

then, from the prior results we have that (dropping the explicit time dependency)

$$p_{\mathbf{x}_k}(\mathbf{x}_k) = p_{\mathbf{x}_{k-1}}(\boldsymbol{\phi}^{-1}(\mathbf{x}_k; t_k, t_{k-1})) \tag{22}$$

Suppose we integrate $\dot{\mathbf{x}} = \mathbf{f}(t, \mathbf{x})$ from t_{k-1} to t_k using a polynomial approach and a Taylor series centered at $\hat{\mathbf{x}}_{k-1}$, we obtain that

$$\begin{aligned} \mathbf{x}_k &= \boldsymbol{\phi}(\mathbf{x}_{k-1}; t_k, t_{k-1}) = \boldsymbol{\phi}(\hat{\mathbf{x}}_{k-1} + d\mathbf{x}_{k-1}; t_k, t_{k-1}) \\ &= \phi(\hat{\mathbf{x}}_{k-1}, t_k, t_{k-1}) + \mathcal{M}_{\hat{\mathbf{x}}_{k-1}}(d\mathbf{x}_{k-1}) = \hat{\mathbf{x}}_k^- + d\mathbf{x}_k \end{aligned} \tag{23}$$

where we defined

$$\hat{\mathbf{x}}_k^- = \phi(\hat{\mathbf{x}}_{k-1}, t_k, t_{k-1}) \tag{24}$$

$$d\mathbf{x}_k = \mathcal{M}_{\hat{\mathbf{x}}_{k-1}}(d\mathbf{x}_{k-1}) \tag{25}$$

we can now invert the map to get

$$\mathbf{x}_{k-1} = \boldsymbol{\phi}^{-1}(\mathbf{x}_k; t_k, t_{k-1}) = \hat{\mathbf{x}}_{k-1} + \mathcal{M}_{\hat{\mathbf{x}}_k^-}(d\mathbf{x}_k) \tag{26}$$

so that

$$p_{\mathbf{x}_k}(\mathbf{x}_k) = p_{\mathbf{x}_{k-1}}(\boldsymbol{\phi}^{-1}(\mathbf{x}_k; t_k, t_{k-1})) = p_{\mathbf{x}_{k-1}}(\hat{\mathbf{x}}_{k-1} + \mathcal{M}_{\hat{\mathbf{x}}_k^-}(d\mathbf{x}_k)) \tag{27}$$

Using Eq. (27) and the definition of $\Xi_{k-1}(\mathbf{x}_{k-1})$

$$\begin{aligned} \Xi_{k-1}(d\mathbf{x}_{k-1}) &= \Xi_{k-1}\Big(\mathcal{M}_{\hat{\mathbf{x}}_k^-}(d\mathbf{x}_k)\Big) = \log p_{\mathbf{x}_{k-1}}(\hat{\mathbf{x}}_{k-1} + \mathcal{M}_{\hat{\mathbf{x}}_k^-}(d\mathbf{x}_k)) \\ &= \log p_{\mathbf{x}_k}(\hat{\mathbf{x}}_k^- + d\mathbf{x}_k) = \Xi_k^-(d\mathbf{x}_k) \end{aligned} \tag{28}$$

where again, additive constants are discarded. Therefore, $\Xi_k^-(d\mathbf{x}_k) = \Xi_{k-1}\Big(\mathcal{M}_{\hat{\mathbf{x}}_k^-}(d\mathbf{x}_k)\Big)$ is the Taylor series approximation of the logarithm of the prior distribution of $\mathbf{x}_k$ (modulo a constant) centered at $\hat{\mathbf{x}}_k^-$.

Measurement Update

We assume the measurement $\mathbf{y}_k$ is a (possibly) nonlinear function of the state $\mathbf{x}_k$ corrupted by zero-mean, additive Gaussian noise $\boldsymbol{\eta}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k) + \boldsymbol{\eta}_k \tag{29}$$

The likelihood and log-likelihood functions are given by

$$L(\mathbf{x}_k|\mathbf{y}_k) = \mathcal{N}\big(\mathbf{h}(\mathbf{x}_k), \mathbf{R}_k\big) \tag{30}$$

$$\Lambda(\mathbf{x}_k|\mathbf{y}_k) = -\frac{1}{2}(\mathbf{y}_k - \mathbf{h}(\mathbf{x}_k))^{\mathrm{T}}\mathbf{R}_k^{-1}(\mathbf{y}_k - \mathbf{h}(\mathbf{x}_k)) + C_k \tag{31}$$

we can once again safely ignore the constant C_k. We will now approximate $\mathbf{h}(\mathbf{x}_k)$, and hence $\Lambda(\mathbf{x}_k|\mathbf{y}_k)$, as a Taylor series expansion centered at $\hat{\mathbf{x}}_k^-$, and we will then compute

$$\Xi_k^+(d\mathbf{x}_k) = \Xi_k^-(d\mathbf{x}_k) + \Lambda(\mathbf{x}_k|\mathbf{y}_k) \tag{32}$$

therefore $\Xi_k^+(\mathbf{x}_k)$ is the Taylor series approximation of the logarithm of the posterior distribution of $\mathbf{x}_k$ (modulo a constant) centered at $\hat{\mathbf{x}}_k^-$.

MAP Estimate

To find the MAP estimate $\hat{\mathbf{x}}_k$ we need to find the value of $d\mathbf{x}_k$ that maximizes $\Xi_k^+(d\mathbf{x}_k)$ in Eq. (32), a good initial guess for the search is $d\mathbf{x}_k = 0$.

$$d\hat{\mathbf{x}}_k = \max_{d\mathbf{x}_k} \Xi_k^+(d\mathbf{x}_k) \tag{33}$$

then the MAP estimate is simply given by

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + d\hat{\mathbf{x}}_k \tag{34}$$

at this point we shift the center of the Taylor series of $\Xi_k^+(d\mathbf{x}_k)$ from $\hat{\mathbf{x}}_k^-$ to $\hat{\mathbf{x}}_k$ to obtain $\Xi_k(d\mathbf{x}_k)$; the Taylor series approximation of the logarithm of the posterior distribution of $\mathbf{x}_k$ (modulo a constant) centered at the MAP estimate $\hat{\mathbf{x}}_k$.

We are now ready to go back to the time propagation to advance to the $k+1$ step.

THE DIFFERENTIAL ALGEBRA MAXIMUM A POSTERIORI FILTER (DAMAP)

This section deeply analyses how the Differential Algebra Maximum A Posteriori filter (DAMAP) has been implemented using the Differential Algebra Core Engine (DACE) [11, 12].

Differential Algebra

A brief review on DA is now presented and it follows the development in Ref. [13].

The key idea of DA is to define algebraic operations between polynomials similar to those commonly used with real numbers [14] by suppling the tools to compute the derivatives of functions within a computer environment [15]. More specifically, by substituting the classical implementation of real algebra with the implementation of a new algebra of Taylor polynomials, any function $\mathbf{f}[\cdot]$ of n variables is expanded into its Taylor polynomial up to an arbitrary order c.

Rather than approximating analytic functions with a collection of points, they can be represented via their Taylor coefficients. For dynamic systems, this approach results in potentially replacing thousands of integrations with evaluations of the Taylor expansion of the flow. As a result, in many applications, the computational time reduces considerably without any significant loss in accuracy [16].

DA techniques find their origin in the attempt to solve analytical problems by an algebraic approach. Rather than representing a function as a collection of its values at specific points, DA approximate it with the coefficients of a monomial basis functions. Using this framework in a computer environment allows to treat functions and their operations similarly to real numbers. Of particular interest is the efficient calculation of Taylor series expansions of the flow of ordinary differential equations (ODEs) in terms of its initial conditions. Numerical ODE solvers are based on algebraic operations, involving the evaluation of the derivative at several integration points. Starting from the DA representation of the initial conditions and carrying out all the evaluations in the DA framework, the flow of an ODE is obtained at each step as its Taylor expansion centered at the initial conditions.

Figure 1 depicts a graphical representation of the DA framework. Consider two real numbers a and b: in order to operate in a computer environment, they are represented by their floating points (FP) representation, $\bar{\mathbf{a}}$ and $\bar{\mathbf{b}}$ respectively. Then, given any operation "$\times$" in the set of the real numbers, an adjoint operation "$\oplus$" is defined in the set of the FP numbers such that the diagram in Figure 1 commutes. Consequently, the same result is obtained if the real numbers a and b are represented in their FP representation prior to operating on them, or if the operation is carried out between the real numbers and subsequently the result expressed in its FP representation (refer to the left portion of Figure 1). Analogously, starting from two different, sufficiently regular, functions f and g, using the DA framework, the computer environment operates on them by representing them with their Taylor series expansions, referred respectively as F and G. Similarly to the representation of real numbers with their FP expression, we now substitute represent the functions f and g

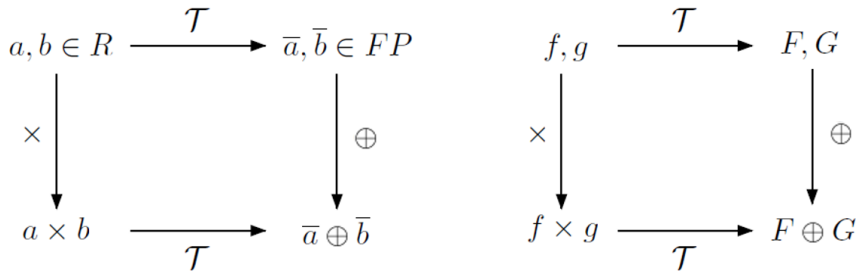

Figure 1. Analogy between the floating point representation of real numbers in a computer environment (left figure) and the introduction of the algebra of Taylor polynomials in the differential algebraic framework (right figure). Image from [16, 17, 18]

with their truncated Taylor series expansion up to a user-selected order c. For each operation in the space of function, an adjoint operation in the space of Taylor polynomials is defined such that the diagram in Figure 1 commutes. As for real numbers, the same end result is obtain by extracting the Taylor expansions of f and g and operating on them in the set of Taylor polynomial, then operating on f and g in the original space and then extracting the Taylor expansion of the resulting function.

For a more complete and detailed explanation on Differential Algebra, please refer to the references.

DAMAP

Given the following filtering statement

$$\mathbf{x}_{k+1} = \phi(\mathbf{x}_k) \tag{35}$$

$$\mathbf{z}_{k+1} = h(\mathbf{x}_{k+1}) + \nu_{k+1} \tag{36}$$

where ϕ is the process model, $\mathbf{x}_k$ is the n components long state vector of the system at time-step k, $\mathbf{z}_{k+1}$ is the m components long measurement vector at time-step $k+1$, and h is the measurements function. Measurement noise ν is a Gaussian random sequence with covariance

$$\mathbb{E}\left\{\nu(i)\nu^T(k)\right\} = \mathbf{R}\delta_{ik} \tag{37}$$

The initialization of the filtering algorithm is performed by assuming a normal prior distribution of the state with mean $\hat{\mathbf{x}}_\mathbf{0}$ and covariance $\mathbf{P}_0$.

$$p_{\mathbf{x_0}}(\mathbf{x}) = \frac{1}{(2\pi)^{n/2}\sqrt{\det \mathbf{P}_0}} \exp\left(-\frac{1}{2}(\mathbf{x}_0 - \hat{\mathbf{x}}_\mathbf{0})^T \mathbf{P}_0^{-1}(\mathbf{x}_0 - \hat{\mathbf{x}}_\mathbf{0})\right) \tag{38}$$

Differential Algebra (DA) works with deviations from the mean: the initialization of the DA variable is

$$d\mathbf{x_0} = \mathbf{x}_0 - \hat{\mathbf{x}}_\mathbf{0} \tag{39}$$

where, indeed, $\mathbf{x}_0$ is the Taylor expansion series of the state at time step 0 centered in $\hat{\mathbf{x}}_\mathbf{0}$. By considering the logarithm of the distribution, the prior can be expressed as a quadratic form.

$$\Xi_0(d\mathbf{x}_0) = -\frac{1}{2} d\mathbf{x}_0^{\mathrm{T}} \mathbf{P}_0^{-1} d\mathbf{x}_0 \tag{40}$$

where all constants have been safely ignored. It is important to point out that after selecting the expansion order c of the Taylor series, the creation of the prior requires to set the DA order to $2c$ since each time the polynomial multiplies itself the order increases of a factor c. Equation (40) is the representation of the Gaussian exponent as a $2c$ order polynomial where each DA variable is the deviation of the state from the initial mean.

Propagation

The prediction part of the filter starts with the time propagation of the distribution. The expansion order of the DACE is set to be c.

$$\mathbf{x}_{k+1}^{-} = \phi(\mathbf{x}_{k}^{+}) \tag{41}$$

The propagated polynomial can be divided into its constant part and the map that connects the deviations at time step k to time step $k+1$.

$$\mathbf{x}_{k+1}^{-} = \hat{\mathbf{x}}_{k+1}^{-} + \mathcal{M}_{(k\to k+1)}^{\hat{\mathbf{x}}_{k}^{+}}(d\mathbf{x}_{k+1}) \tag{42}$$

where the superscript indicates where the map is centered and the subscripts the expansion direction: forward or backwards. Indeed, the map can be now inverted:

$$\mathcal{W}_{(k+1\to k)}^{\hat{\mathbf{x}}_{k+1}^{-}}(d\mathbf{x}_{k+1}) = \left(\mathcal{M}_{(k\to k+1)}^{\hat{\mathbf{x}}_{k}^{+}}(d\mathbf{x}_{k+1})\right)^{-1} \tag{43}$$

The Taylor series approximation of the logarithm of the prior distribution of $\mathbf{x}_{k+1}$ is the polynomial evaluation of $\Xi_{k}^{++}(d\mathbf{x}_{k})$ using the inverted map.

$$\Xi_{k+1}^{-}(d\mathbf{x}_{k+1}) = \Xi_{k}^{++}\left(\mathcal{W}_{(k+1\to k)}^{\hat{\mathbf{x}}_{k+1}^{-}}(d\mathbf{x}_{k+1})\right) \tag{44}$$

where the expansion order is doubled due to the quadratic nature of the exponent, and $\Xi_{k}^{++}(d\mathbf{x}_{k})$ is the updated distribution from the previous step.

Measurements Update

The prior is now centered at $\hat{\mathbf{x}}_{k+1}^{-}$, thus a new DA variable initialization around the propagated mean is requested:

$$\mathbf{x}_{k+1}^{-} = \hat{\mathbf{x}}_{k+1}^{-} + d\mathbf{x} \tag{45}$$

and the predicted measurements can be computed in the DA framework (order c)

$$\mathbf{z}_{k+1} = h(\mathbf{x}_{k+1}^{-}) \tag{46}$$

Given the measurements from the sensors $\mathbf{y}_{k+1}$, the log-likelihood is computed (order $2c$)

$$\Lambda(\mathbf{x}_{k+1}|\mathbf{y}_{k+1})(d\mathbf{x}_{k+1}) = -\frac{1}{2}(\mathbf{y}_{k+1} - \mathbf{z}_{k+1})^{\mathrm{T}}\mathbf{R}_{k+1}^{-1}(\mathbf{y}_{k+1} - \mathbf{z}_{k+1}) \tag{47}$$

where, again, constants can be safely ignored. MAP estimators consist in the maximization of the posterior distribution of the state; however, since the posterior PDF is proportional to the joint distribution, maximizing the latter means finding the estimate. Working with logarithms, the joint distribution is evaluated easily as the sum of the exponents; thus

$$\Xi_{k+1}^{+}(d\mathbf{x}_{k+1}) = \Xi_{k+1}^{-}(d\mathbf{x}_{k+1}) + \Lambda(\mathbf{x}_{k+1}|\mathbf{y}_{k+1})(d\mathbf{x}_{k+1}) \tag{48}$$

is the Taylor series expansion of the posterior distribution of $\mathbf{x}_{k+1}$ centered in $\hat{\mathbf{x}}_{k+1}^{-}$.

Maximization

In order to find the MAP estimate, we need to find the deviation that maximizes the posterior distribution:

$$\bar{d\mathbf{x}} = \max_{d\mathbf{x}_{k+1}} \Xi_{k+1}^{+}(d\mathbf{x}_{k+1}) \tag{49}$$

Maximization in the DA framework is performed at first by evaluating the gradient of the function it is desired to maximize.

$$\Theta(d\mathbf{x}_{k+1}) = \nabla \Xi^{+}_{k+1}(d\mathbf{x}_{k+1}) \tag{50}$$

Any solver algorithm can be used. In this case, it has been decided to implement a straightforward Newton's method to find the root of the gradient $\Theta(d\mathbf{x}_{k+1})$. After evaluating, at each iteration j, the Jacobian $\mathcal{J}$ of the gradient at $d\mathbf{x}^{[j]}_{k+1}$, the solution is found by applying iteratively the following algorithm

$$d\mathbf{x}^{[j+1]}_{k+1} = d\mathbf{x}^{[j]}_{k+1} - \mathcal{J}^{-1}\Theta(d\mathbf{x}^{[j]}_{k+1}) \tag{51}$$

until a certain level of tolerance is reached. The deviation that maximizes the posterior distribution is therefore evaluated as the output of the Newton's method. Indeed, the MAP estimate is given by

$$\hat{\mathbf{x}}^{+}_{k+1} = \hat{\mathbf{x}}^{-}_{k+1} + \bar{d\mathbf{x}} \tag{52}$$

Before starting the next time-step, the filter needs to recenter the Taylor expansion series around the updated estimate, performing an additional shift of the polynomials. Therefore, a new DA variable is initialized with center $\bar{d\mathbf{x}}$.

$$d\mathbf{x}^{+}_{k+1} = \bar{d\mathbf{x}} + d\mathbf{x}_{k+1} \tag{53}$$

such that the state Taylor expansion series $\mathbf{x}^{-}_{k+1}$ and the distribution $\Xi^{+}_{k+1}(d\mathbf{x}_{k+1})$ can perform the center shift to their final versions.

$$\mathbf{x}^{+}_{k+1} = \mathbf{x}^{-}_{k+1}(d\mathbf{x}^{+}_{k+1}) \tag{54}$$

$$\Xi^{++}_{k+1}(d\mathbf{x}_{k+1}) = \Xi^{+}_{k+1}(d\mathbf{x}^{+}_{k+1}) \tag{55}$$

Equation 54 and Equation 55 are simple polynomial evaluation where the variable is substituted by the center shift value added to the new DA variable. A check that the gradient of the updated distribution is null at its center is performed before starting the following step.

NUMERICAL EXAMPLES

The proposed filter has been tested on an orbit determination problem. The equations of motion governing the spacecraft are associated to the Keplerian dynamics

$$\frac{d\ddot{\mathbf{r}}}{dt} = -\frac{\mu}{r^3}\mathbf{r} \tag{56}$$

where $\mathbf{r}$ is the position vector and μ is the Earth gravitational parameter. Following the example from [17], the problem has been normalized to be non-dimensional with length scaled by the orbit semi-major axis, $a = 8788\ km$, and time by the scaled orbital period $\sqrt{\frac{a^3}{\mu}}$. The initial conditions and uncertainties values are coherent with previous works [15, 19], where the initial estimates for the state have a 10% offset from the true initial state

$$\mathbf{x}_0 = \begin{pmatrix} \mathbf{r}_0 \\ \mathbf{v}_0 \end{pmatrix} = \begin{pmatrix} -0.68787 \\ -0.39713 \\ 0.28448 \\ -0.51331 \\ 0.98266 \\ 0.37611 \end{pmatrix} \tag{57}$$

As regarding the measurements equations, range and bearing angles are taken with respect of the center of planet at different acquisition frequencies.

$$y_1 = r + \eta_1 \tag{58}$$

$$y_2 = arctan\left(\frac{x_2}{x_1}\right) + \eta_2 \tag{59}$$

$$y_3 = arcsin\left(\frac{x_3}{r}\right) + \eta_3 \tag{60}$$

where η_i, $i = 1, 2, 3$, is the measurement noise, assumed to be Gaussian. The standard deviation of the error is assumed to be $0.1m$ in range and $0.1arcsec$ for the angles. The initial uncertainty is assumed to be Gaussian as well, with a diagonal initial error covariance matrix: the value of the variance for the position part of the state vector components is 0.01, while the variance for the velocity components is 10^{-4}. Therefore

$$\sigma_r = 10^{-2}a \tag{61}$$

$$\sigma_v = 10^{-4}\sqrt{\frac{\mu}{a}} \tag{62}$$

In this examples, DAMAP is compared to the DA-Based High Order Extended Kalman Filter (EKFDA) implemented in previous works [19, 17]. This comparison is of importance since it underlines the main differences between a MAP estimator (DAMAP) and a Linear Minimum Mean Square Error (LMMSE) one. After performing a high order prediction step using DA techniques, the update step of the EKFDA resemble the classical Kalman filter. Performance of different orders of Taylor series approximation are investigated. In the figures below, the filter referred as DAMAP2 indicates that all polynomials have been truncated to the second power, while DAMAP1 performs a simple linearization of the dynamics. Therefore, DAMAP1 (1^{st} order truncation) behaves exactly like the classical Extended Kalman Filter (EKF), and EKFDA1.

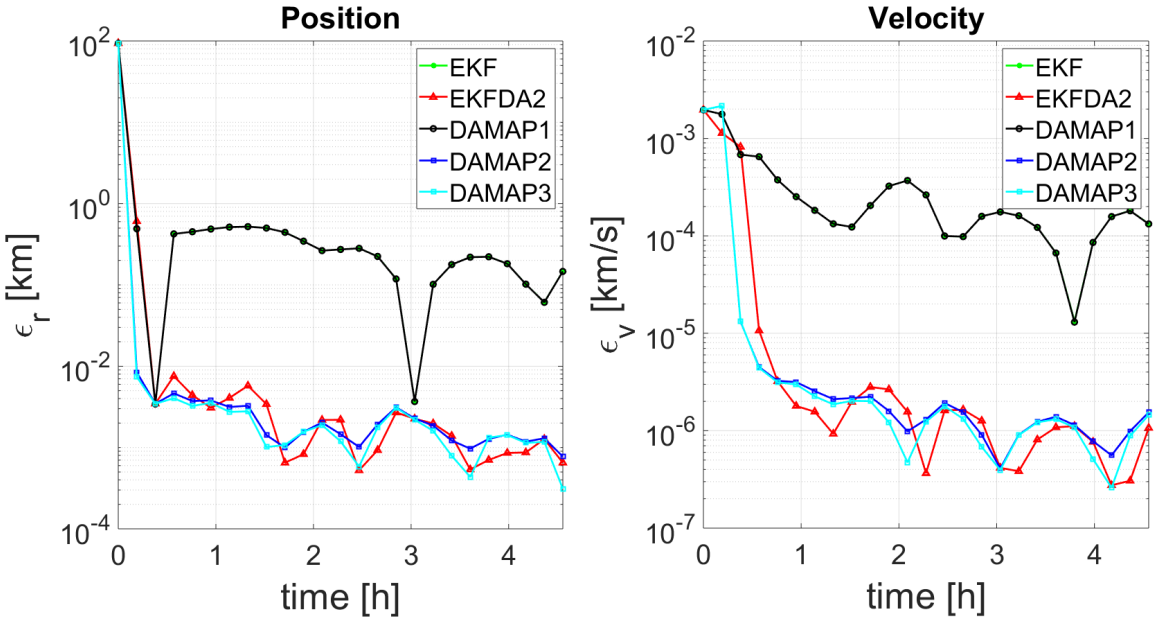

Figure 2. Orbit Determination test: 2 orbits, 12 observations per orbit. Position and Velocity error profiles of DAMAP1, DAMAP 2, EKFDA1 (EKF) and EKFDA2.

Figure 2 shows the position and velocity error profiles obtained with the two filters using first, second and third order in a single test run. The position and velocity errors, expressed respectively as ϵ_r and ϵ_v, are defined as the Euclidean norm of the difference vector between the estimated position, and velocity,

with the corresponding true values. The figure represents the time duration of two orbits, with a total of 12 observations each orbit separated by the same time interval. As expected, the DAMAP1 line is equivalent to the performance of the classical EKF since the two filters share the same approximation of the propagated PDF. On the contrary, second order filters show a significant improvement in the state estimation accuracy, having both DAMAP2 and EKFDA2 settling on the same error level, two orders of magnitude below with respect to filters based on linearization. Indeed, high order filters are able to extract more information form nonlinear equations with respect to the first order. However, by looking at DAMAP3, there appears to be no influential benefits by upgrading to a third order filter. Furthermore, EKFDA3 would have given the same estimates of EKFDA2 because the filter describes the propagated statistics using only the first and second order moments, that is intrinsic in the Kalman approach [19].

The performances of DAMAP2 have then been tested with a Monte Carlo analysis using 800 samples and compared to the EKFDA2 in order to visualize, again, the differences between a high order MAP estimator and a LMMSE one. The Monte Carlo analysis has been performed starting each sample with the same 10% offset of the state. Figure 3 and Figure 4 show the simulation results for the position part of the state, left

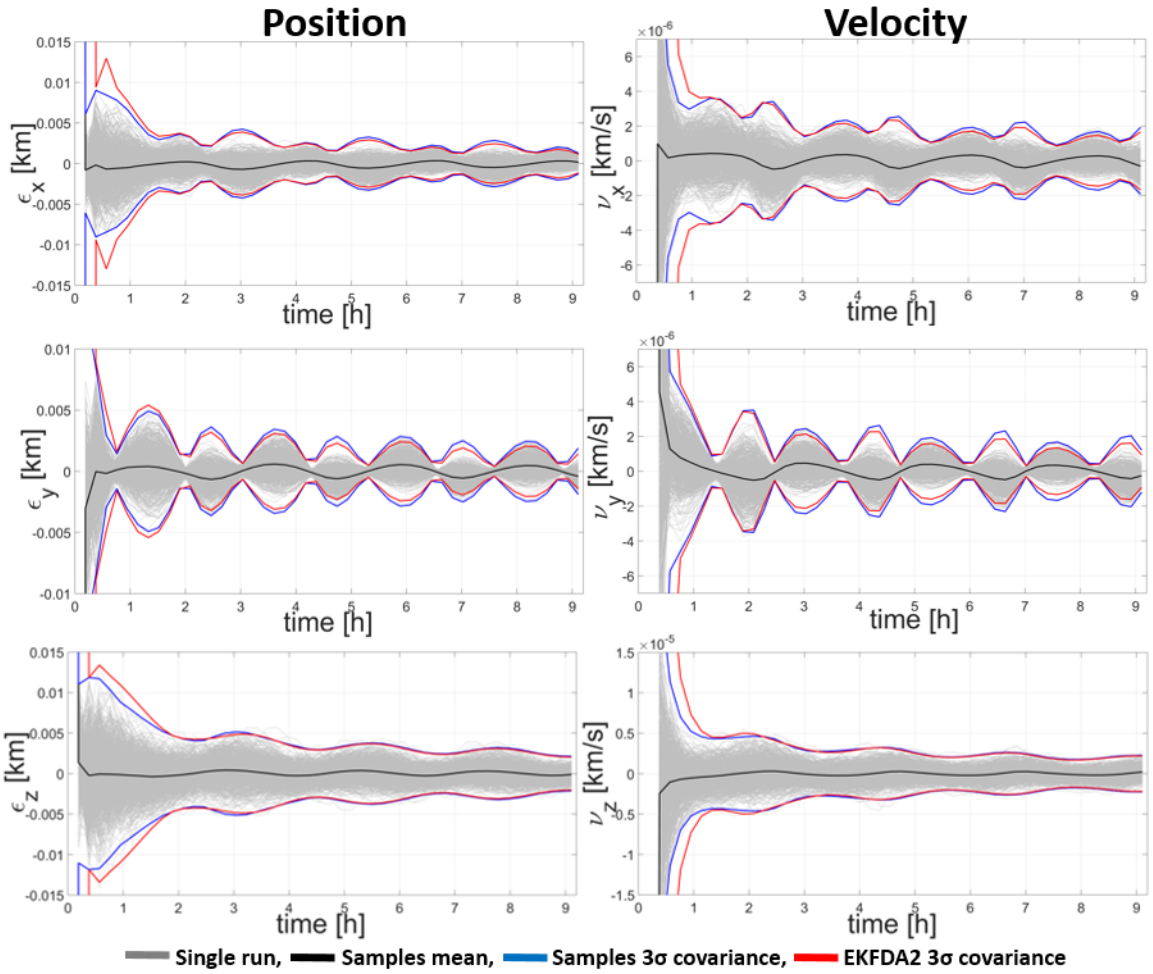

Figure 3. DAMAP2 position and velocity profiles errors. 4 orbits with 12 observations per orbit. 800 runs.

column, and the velocity part, right column, of the robustness analysis in a simulation with time duration of 4 orbits with a total of 12 observations each orbit (separated by the same time step). The estimation errors are zoomed in to show the reach of steady state. In the figures, the black lines show the means of the

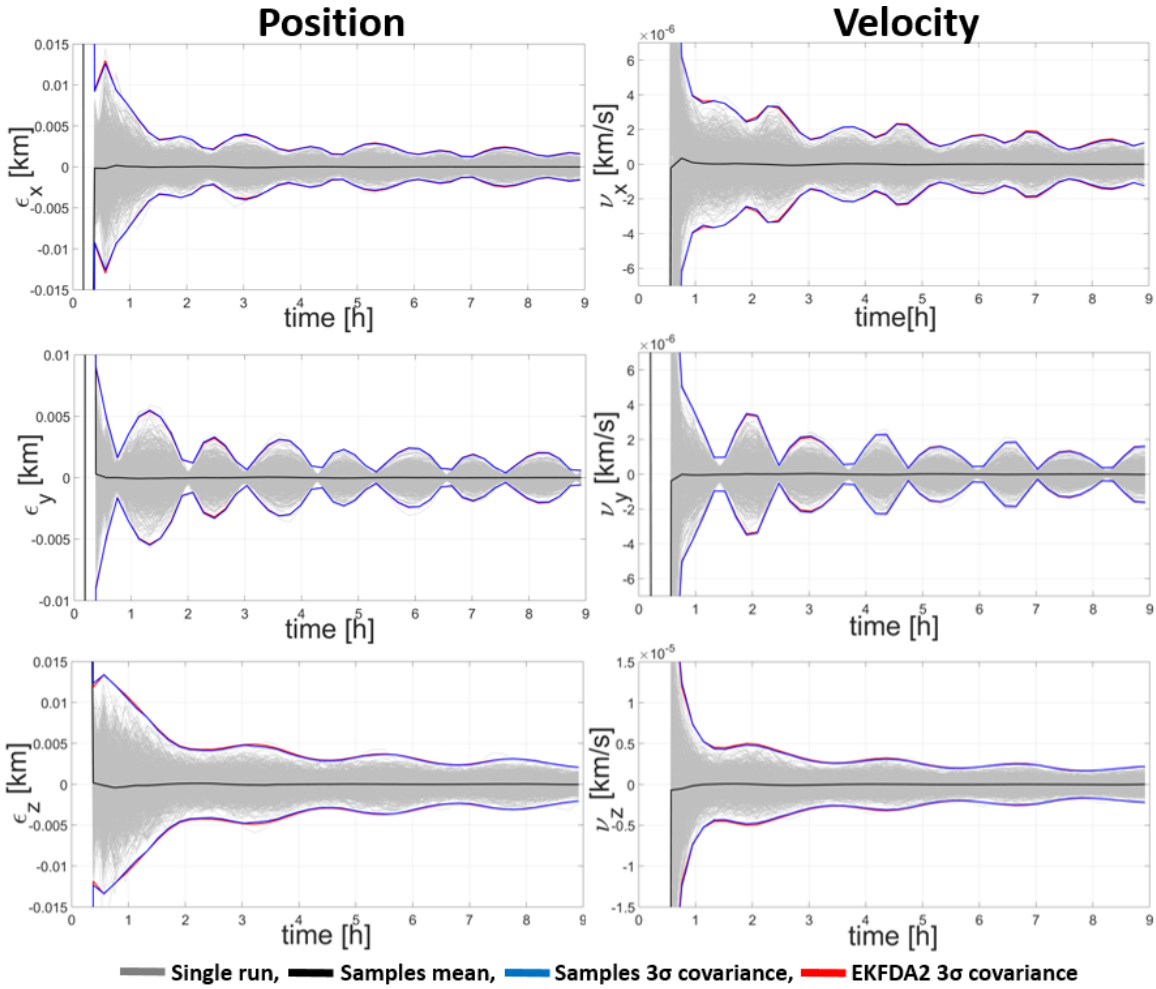

Figure 4. EKFDA2 position and velocity profiles errors. 4 orbits with 12 observations per orbit. 800 runs.

estimation errors among all the 800 test runs. The blue lines represent the square root of the mean square of 800 samples, i.e. $\sqrt{(x_k - \hat{x}_k)^2/N_{monte}}$, while the red lines are the EKFDA2's estimated 3σ standard deviations. The EKFDA2's estimated 3σ standard deviations are shown in Figure 3 for comparison purposes. It can be noted how the two filters reach approximately the same level of accuracy and they behave similarly. However, after an initial transient period, it can be seen that the LMMSE solution has a lower mean square error than the MAP solution, as theoretically expected.

In the next example, the measurement acquisition frequency is reduced to 4 observations per orbit (Figures 5 and 6); it can be noted how the mean of the EKFDA2 error is zero (as theoretically expected, which results in variance and mean square coinciding) while the mean of the DAMAP2 error is non-zero, i.e. it has a bias that follows the position of the spacecraft in the orbit.

The MAP estimate is the most likely realization of the posterior distribution, which does not necessarily coincide with the condition mean, resulting in a, possibly, bias estimate. In Figure 5 it can be noticed more clearly that, since EKFDA2 minimizes the mean square error (LMMSE), after having reached steady state, the red lines are noticeably tighter than the blue ones. However, DAMAP2 reaches steady state level faster with respect to EKFDA2.

At simulation time $\bar{t} = 10.82$ hours of simulation time, the EKFDA2 position error has sample mean (in

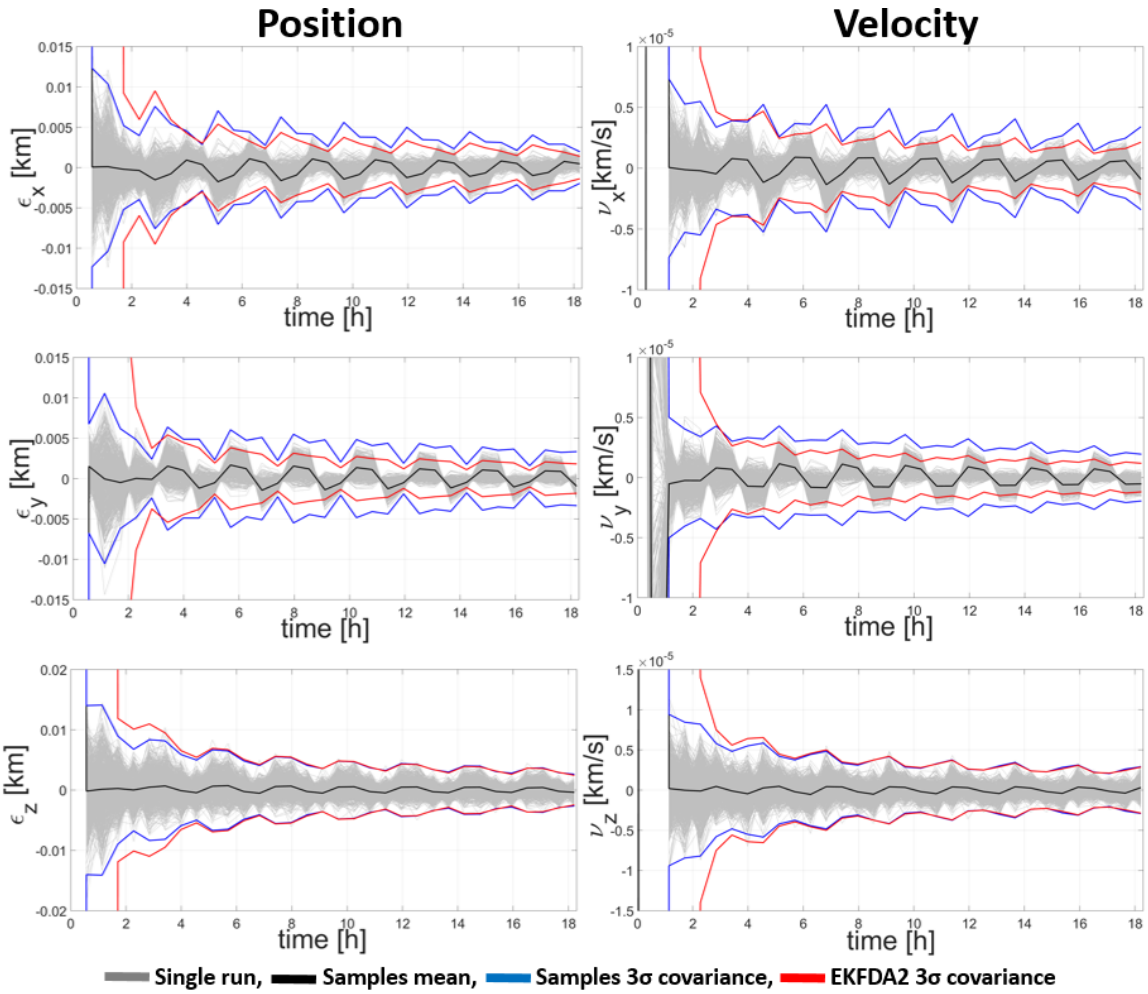

Figure 5. DAMAP2 position and velocity profiles errors. 8 orbits with 4 observations per orbit. 800 runs.

km) and sample covariance matrix (in km^2) :

$$\bar{\boldsymbol{\epsilon}}_{\boldsymbol{r}EKFDA2} = 10^{-4}\begin{bmatrix} 0.0395 \\ 0.1234 \\ -0.0848 \end{bmatrix} \quad \mathbf{P}_{EKFDA2} = 10^{-5}\begin{bmatrix} 0.0677 & 0.0339 & 0.0691 \\ 0.0339 & 0.0702 & -0.0365 \\ 0.0691 & -0.0365 & 0.1677 \end{bmatrix}$$

while DAMAP2 has sample mean and covariance matrix:

$$\bar{\boldsymbol{\epsilon}}_{\boldsymbol{r}DAMAP2} = \begin{bmatrix} 0.0010 \\ 0.0011 \\ -0.0002 \end{bmatrix} \quad \mathbf{P}_{DAMAP2} = 10^{-5}\begin{bmatrix} 0.0629 & 0.0315 & 0.0639 \\ 0.0315 & 0.0610 & -0.0278 \\ 0.0639 & -0.0278 & 0.1463 \end{bmatrix}$$

These means and covariance matrices are depicted in Figure 7: EKFDA2, in red and DAMAP2 in blue. The figure shows 3σ ellipsoids centered at the mean, as well as their projection on the xy, xz and yz planes. These data shows a common characteristic of MAP vs. LMMSE estimators. The MAP estimate is often biased (non-zero mean error) but it often has a smaller spread of the error distribution (i.e. a smaller variance which represents the spread around the mean). The LMMSE, on the other hand, is unbiased (i.e. zero-mean estimation error), therefore its estimation error variance coincides with the mean square. The LMMSE mean square error (or variance) is smaller than the MAP's mean square error (sum of the variance and the square

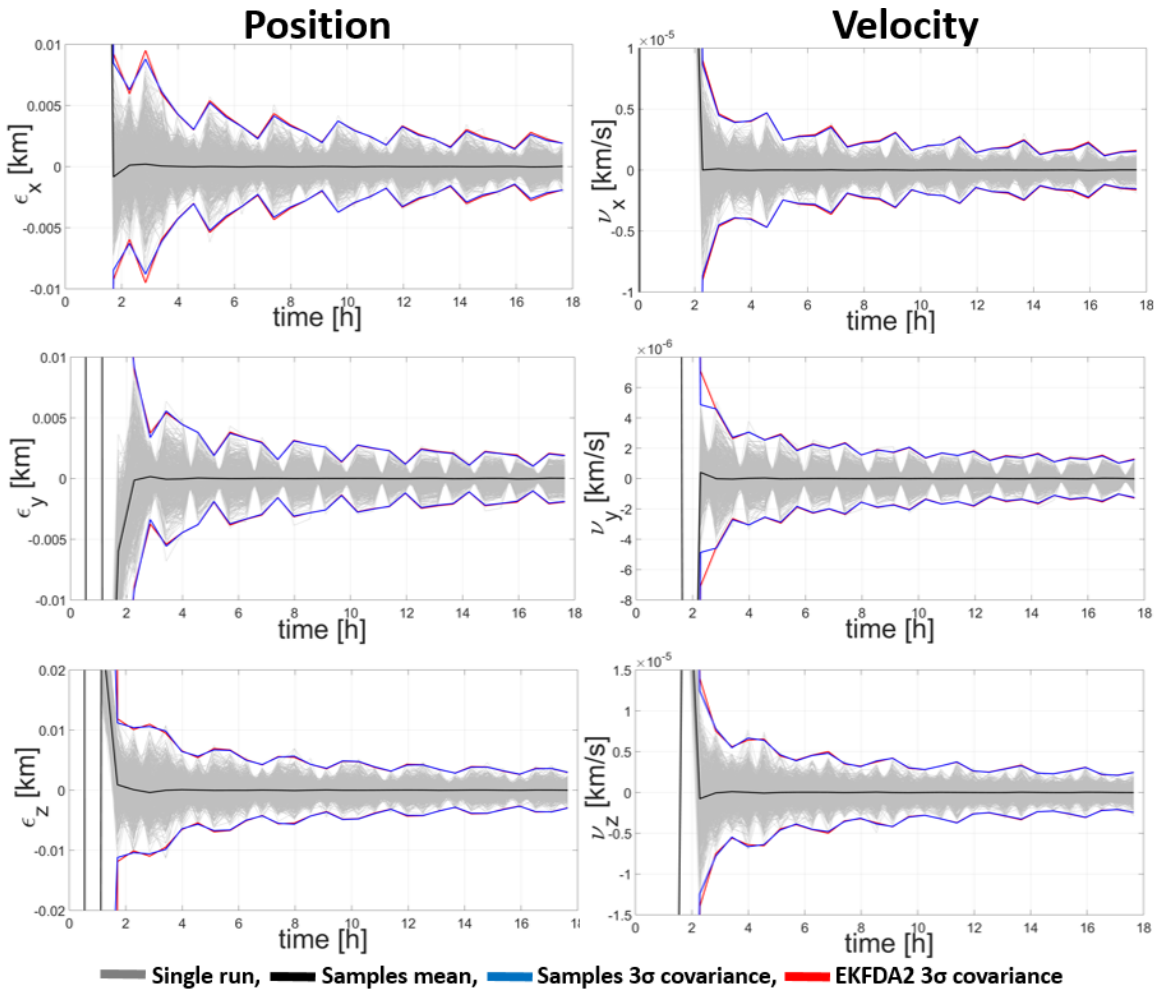

Figure 6. EKFDA2 position and velocity profiles errors. 8 orbits with 4 observations per orbit. 800 runs.

of the mean). In summary: LMMSE has, on average, less estimation error, while MAP has, often, a tighter distribution because it returns the most likely outcome.

CONCLUSIONS

A filter based on the Maximum A Posteriori principle has been presented. The capability of propagating and updating the PDF by working directly on the exponent of the distribution had given DAMAP2 the ability to reach satisfying accuracy levels. Moreover, thanks to the polynomial approximation of the exponent through DA techniques, the filter is computational light and is able to reach convergence level faster than high order filters based on the LMMSE, such has EKFDA2.

At high expansion orders, the curvilinear approximation of the distribution, during maximization, creates a bias in the state estimate. The behavior of the bias is directly connected to the orbit parameters and future developments are focused on its estimation in order to further reduce the state estimation error.

REFERENCES

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

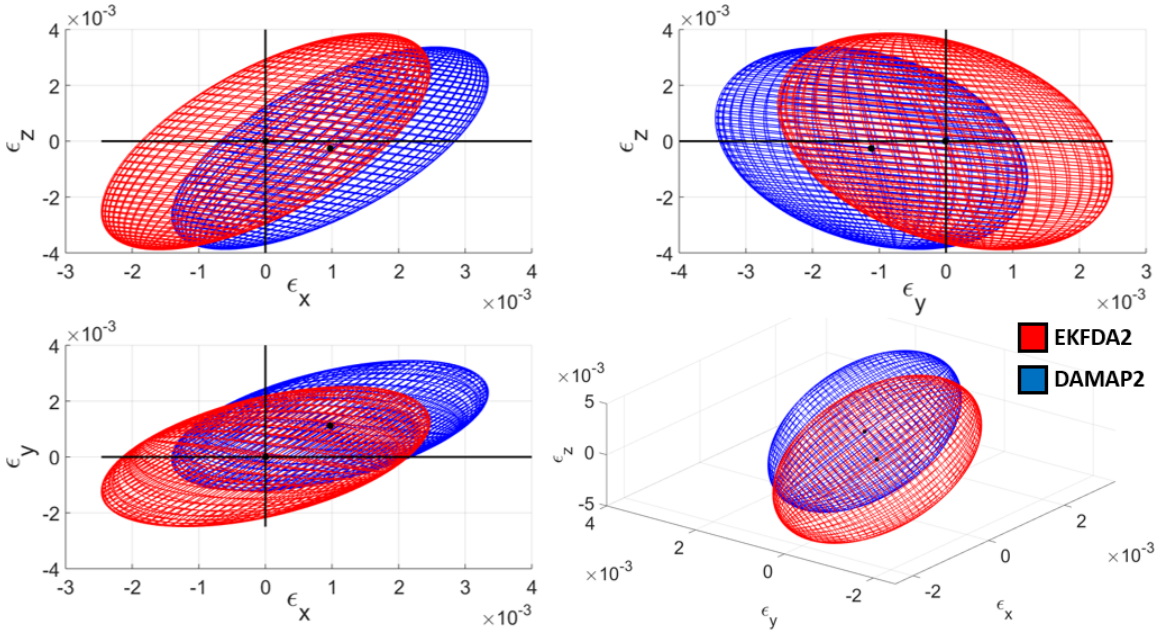

Figure 7. Position estimation error distribution (in km) approximated as a Gaussian Distribution for DAMAP2 and EKFDA2 at time $\bar{t} = 10.82$ h.

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

[3] Tapley, B. D., Schutz, B. E., and Born, G. H., *Statistical Orbit Determination*, Elsevier Academic Press, 2004, ISBN: 9780080541730.

[4] Gelb, A., editor, *Applied Optimal Estimation*, The MIT press, Cambridge, MA, 1974, ISBN:9780262200271.

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

[6] Julier, S. J. and Uhlmann, J. K., "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, Vol. 92, No. 3, 2004, pp. 401–422.

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

[8] Park, R. and Scheeres, D., "Nonlinear mapping of Gaussian statistics: theory and applications to spacecraft trajectory design," *Journal of guidance, Control, and Dynamics*, Vol. 29, No. 6, 2006, pp. 1367–1375, doi: 10.2514/1.20177.

[9] Park, R. S. and Scheeres, D. J., "Nonlinear Semi Analytic Methods for Trajectory Estimation," *Journal of Guidance Control and Dynamics*, Vol. 30, No. 6, 2007, pp. 1668–1676, doi: 10.2514/1.29106.

[10] Valli, M., Armellin, R., Di Lizia, P., and Lavagna, M., "Nonlinear mapping of uncertainties in celestial mechanics," *Journal of Guidance, Control, and Dynamics*, Vol. 36, No. 1, 2012, pp. 48–63, doi: 10.2514/1.58068.

[11] Rasotto, M., Morselli, A., Wittig, A., Massari, M., Di Lizia, P., Armellin, R., Valles, C., and Ortega, G., "Differential algebra space toolbox for nonlinear uncertainty propagation in space dynamics," 2016.

[12] Massari, M., Di Lizia, P., Cavenago, F., and Wittig, A., "Differential Algebra software library with automatic code generation for space embedded applications," *2018 AIAA Information Systems-AIAA Infotech@ Aerospace*, 2018, p. 0398, doi: 10.2514/6.2018-0398.

[13] Servadio, S., *High Order Filters For Relative Pose Estimation Of An Uncooperative Target*, Master's thesis, Politecnico di Milano, Milan,Italy,20156, 2017.

[14] Witting, A., Lizia, P. D., and R. Armellin et al., "An introduction to Differential Algebra," Dinamica Innovating Technology, European Space Agency.

[15] Valli, M., Armellin, R., Di Lizia, P., and Lavagna, M. R., "Nonlinear filtering methods for spacecraft navigation based on differential algebra," *Acta Astronautica*, Vol. 94, No. 1, 2014, pp. 363–374, doi: 10.1016/j.actaastro.2013.03.009.

[16] Armellin, R., Di Lizia, P., Bernelli-Zazzera, F., and Berz, M., "Asteroid close encounters characterization using differential algebra: the case of Apophis," *Celestial Mechanics and Dynamical Astronomy*, Vol. 107, No. 4, 2010, pp. 451–470, doi: 10.1007/s10569-010-9283-5.

[17] Cavenago, F., Di Lizia, P., Massari, M., Servadio, S., and Wittig, A., "DA-based nonlinear filters for spacecraft relative state estimation," *2018 Space Flight Mechanics Meeting*, 2018, p. 1964, doi: 10.2514/6.2018-1964.

[18] Valli, M., Armellin, R., Di Lizia, P., and Lavagna, M., "Nonlinear mapping of uncertainties in celestial mechanics," *Journal of Guidance, Control, and Dynamics*, 2013, doi: 10.2514/1.58068.

[19] Di Lizia, P., Massari, M., and Cavenago, F., "Assessment of onboard DA state estimation for spacecraft relative navigation," 2017, Final report, ESA .