Available online at www.sciencedirect.com

ScienceDirect

Advances in Space Research 73 (2024) 53–63

ADVANCES IN SPACE RESEARCH (a COSPAR publication)

www.elsevier.com/locate/asr

Uncertainty propagation in orbital dynamics via Galerkin projection of the Fokker-Planck Equation

Giacomo Acciarini [a,*], Cristian Greco [b], Massimiliano Vasile [b]

[a] Surrey Space Center, University of Surrey, GU27XH Guildford, United Kingdom
[b] University of Strathclyde, 73 Montrose Street, Glasgow G1 1XJ, United Kingdom

Received 10 November 2022; received in revised form 13 November 2023; accepted 26 November 2023
Available online 1 December 2023

Abstract

The Fokker–Planck equation is a partial differential equation that describes how the probability density function of an object state varies, when subject to deterministic and random forces. The solution to this equation is crucial in many space applications, such as space debris trajectory tracking and prediction, guidance navigation and control under uncertainties, space situational awareness, and mission analysis and planning. However, no general closed-form solutions are known and several methods exist to tackle its solution. In this work, we use a known technique to transform this equation into a set of linear ordinary differential equations in the context of orbital dynamics. In particular, we show the advantages of the applied methodology, which allows to decouple the time and state-dependent components and to retain the entire shape of the probability density function through time, in the presence of both deterministic and stochastic dynamics. With this approach, the probability density function values at future times and for different initial conditions can be computed without added costs, provided that some time-independent integrals are solved offline. We showcase the efficacy and use of this method on some orbital dynamics example, by also leveraging the use of automatic differentiation for efficiently computing the involved derivatives.

Keywords: Fokker–Planck equation; Uncertainty propagation; Uncertainty quantification; Orbital dynamics

1. Introduction

Uncertainty quantification and propagation of orbital states plays a pivotal role in Space Situational Awareness (SSA) in many ways: from orbital object tracking and monitoring to conjunction analysis, maneuver planning, and anomaly detection. The state of a satellite interacting within a dynamical system can be influenced by random effects that span from uncertainty in the initial conditions (e.g. due to measurement errors) to random fluctuations of the forces that govern the satellite dynamics (e.g. random fluctuations of atmospheric drag). All these effects give rise to stochastic dynamical systems that are described by either stochastic differential equations (SDEs), whenever random fluctuations of the dynamical forces exist, or ordinary differential equations (ODEs), whenever the sole initial conditions are uncertain. In such cases, the spacecraft state can be studied through the lenses of probability, by analyzing the time variation of the probability density function (pdf) of the state, rather than the state itself. Much work has been carried out in the field of uncertainty propagation for orbital mechanics problems. The most common approaches are linear and linearized models (e.g. extended Kalman filters), simulation-based approaches (e.g. Monte Carlo simulations), and semi-analytic methods (e.g. polynomial chaos expansion and differential algebra techniques) (Luo and

* Corresponding author.
E-mail addresses: giacomo.acciarini@gmail.com (G. Acciarini), massimiliano.vasile@strath.ac.uk (M. Vasile).

https://doi.org/10.1016/j.asr.2023.11.042
0273-1177/

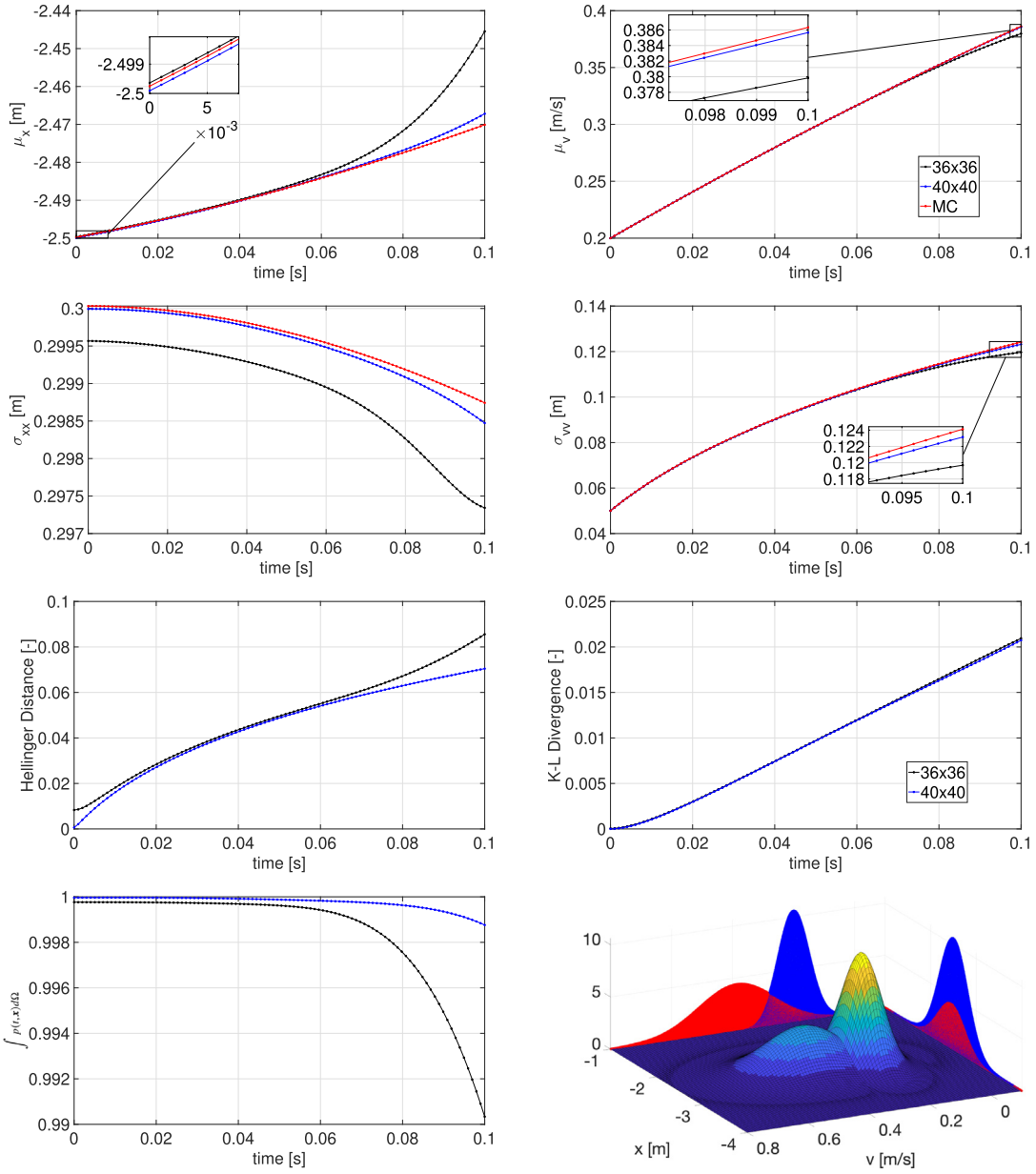

Fig. 1. First and second row: mean and square root of the diagonal of the covariance matrix of both the pdfs reconstructed via Monte Carlo and Galerkin projection using two different sets of basis functions, as a function of time; *third row*: Hellinger distance and KL-divergence between the pdfs of the two Galerkin projection approximations and the Monte Carlo reconstruction, as a function of time; *third row*: on the left, it is displayed the integral of the approximated pdfs as a function of time, while on the right a 3D plot with 2D projections of the fully reconstructed pdf via the Monte Carlo sampling approach.

Yang, 2017; Armellin et al., 2010; Jones and Weisman, 2019; DeMars et al., 2013; Sun and Kumar, 2016; Servadio et al., 2023). Most of these methods are used for predicting the first few moments of the pdf (e.g. mean and standard deviation) and work best for dynamical systems with small nonlinearities and/or Gaussian assumptions. For nonlinear dynamical environments, it becomes particularly interesting to study how the pdf evolves through time, as a result of the forces acting on the object. The Fokker–Planck partial differential equation accurately describes the time-evolution of the pdf, for any dynamical system subject to deterministic and/or random forces, with uncertain initial conditions and/or parameters. However, in general, this equation is not solvable in a closed form, and its numerical solution often requires a substantial computational burden (especially for nonlinear and high-dimensional dynamical systems). For instance, when finite element methods are used for solving this partial differential equation, a spatial mesh has to be generated: for nonlinear dynamics and high-dimensional systems, the complexity of the geometric structures grows fast, thus causing the computational time to quickly rise immensely.

In this work, we propose for the first time the application of a method to transform the Fokker–Planck into a set of ordinary differential equations, which can be solved in a parallelized fashion with a substantially less computational burden (Kumar and Narayanan, 2006; Kumar et al., 2010), in the context of orbital dynamics. To solve the above-mentioned issues, the pdf is first divided into a sum of time-varying coefficients and spatial basis, then, by applying Galerkin projection, then, the computations of the time coefficients are separated from the spatial support. As it is known, this approach allows us to compute the spatial terms in advance and in parallel, while the time-dependant coefficients can be computed separately, therefore allowing us to very quickly propagate any initial condition that can be represented with the chosen basis. This is a key advantage when the computing time is a bottleneck, and when families of pdf's must be propagated (e.g. epistemic uncertainties), which is often the case for orbital dynamics applications. This work stems from the authors' preliminary work on the topic, introducing several novelties, such as broadening the methodology to stochastic dynamics, mathematically identifying and leveraging the sparsity of the matrices of the integral in the case of B-spline basis functions, introducing automatic differentiation to accelerate the computation and exactly compute the derivatives at specified points, and extending the test cases to the diffusive case and to higher dimensions (Acciarini et al., 2020).

The remainder of the paper is structured as follows. In Section 2, we first discuss the background theory, introducing the Fokker–Planck equation. Then, in Section 3, we discuss the methods and key advantages, while in Section 4, we discuss the implementation details for constructing the basis functions and computing the derivatives and the integrals. Furthermore, in Section 5, we discuss the obtained results on some test cases. Finally, in Section 6, we discuss conclusions and future work. This work is freely available open-source.[1]

2. Fokker–Planck Equation

The Fokker–Planck is a partial differential equation that describes the time evolution of the probability density function for a stochastic process. Given a dynamical system governed by a set of differential equations in which at least one of the terms is a stochastic process, one can write the set of equations that describe the motion of a particle subject to deterministic and random forces as (Itô, 1951)

$$d\boldsymbol{X}_t = \boldsymbol{f}(\boldsymbol{X}_t, t)dt + \boldsymbol{\sigma}(\boldsymbol{X}_t, t)d\boldsymbol{W}_t, \tag{1}$$

where $\boldsymbol{X}_t$ is the n-dimensional random state vector, $\boldsymbol{f}$ is the deterministic force vector, and $\boldsymbol{W}_t$ is an m-dimensional standard Wiener process. The Fokker–Planck equation describes how the probability density function of the random variable $\boldsymbol{X}_t$ varies in time, whenever the motion is subject to drifting and diffusive dynamics. The equation was first developed by Fokker and Planck to investigate the Brownian motion of a particle (Risken, 1996). It can be expressed as (Gardiner, 2009)

$$\begin{aligned}\frac{\partial p(\boldsymbol{x},t)}{\partial t} = & -\sum_{i=1}^{n}\frac{\partial}{\partial x_i}(f_i(\boldsymbol{x},t)p(\boldsymbol{x},t))+ \\ & +\sum_{i=1}^{n}\sum_{t=1}^{n}\frac{\partial^2}{\partial x_i \partial x_t}(D_{it}(\boldsymbol{x},t)p(\boldsymbol{x},t)),\end{aligned} \tag{2}$$

where $\boldsymbol{f} = [f_1, .., f_n]^T$ is the vector of the deterministic forces, whereas $D = 1/2\,\boldsymbol{\sigma}\boldsymbol{\sigma}^T$ is the diffusion tensor, with $\boldsymbol{\sigma}$ being an $n \times n$ matrix which describes the diffusion forces involved in the dynamics. In this paper, we will discuss both the case in which only the initial conditions are uncertain and the dynamics is deterministic, as well as the case in which the initial conditions are uncertain and the dynamics is stochastic. In all the experiments we perform, we compare our approximated solution with the solution found by propagating a high number of samples at future times (which we refer to as Monte Carlo approach) and reconstructing the probability density function by fitting Gaussian mixture models with 10 components. This was done using the iterative expectation–maximization algorithm (McLachlan et al., 2019).

3. Methodology

3.1. Galerkin Projection

The Fokker–Planck differential equation is first transformed into a set of ordinary differential equations, by representing the pdf as a sum of time-varying coefficients and spatial basis. Then, the Galerkin projection is applied to reduce the Fokker–Planck equation into a set of ordinary differential equations.

First, we write the probability density function as

$$p(\boldsymbol{x}, t) = \sum_{j=1} a_j(t)\Phi_j(\boldsymbol{x}), \tag{3}$$

where $\boldsymbol{x} = [x_i, .., x_n]^T : \Omega_i \rightarrow \mathbb{R}^n$ are the independent state variables, $a_j(t) : \omega_t \rightarrow \mathbb{R}$ are the time-varying coefficients, and Φ_j are N basis functions chosen to represent the pdf. Moreover, the pdf is defined over the domain $\Omega = \omega_t \times \prod_{i=1}^{n}\Omega_i$. In general, using Eq. (3) does not guarantee that the approximated pdf maintains the properties of a probability density function (i.e., nonnegativity everywhere and that the area under the entire curve is one). While this is something one can use to monitor the goodness of the pdf representation (e.g., checking how closely to one the integral is), it might be a problem if one wants to sample from the distribution or use the approximated pdf for other applications. Hence, it is important to have a method to guarantee both these properties. A way to have these two properties automatically verified would be to directly choose a formulation that accounts for these

[1] https://github.com/Sceki/fpe_orbital_dynamics, date of access: November 2022.

constraints by construction. For instance, by selecting the basis functions and coefficients so that the pdf is always positive and it is enforced that the integral is one. An example is a weighted mixture of Gaussians, whose weights are imposed to sum to one. Instead, if one does not have these properties imposed by construction, another option is to post-process the approximated pdf to match these constraints. As we will detail later in Section 4.1, in our case, we use B-spline basis functions without any added constraints: this approach does not enforce by construction either of those constraints. Hence, to re-establish these pdf properties, it is necessary to post-process the approximated pdf, by normalizing (dividing by its integral) the probability density function and by ensuring that it is never below zero (e.g. clipping at a certain threshold). While these procedures would restore the pdf properties, they also negatively impact the quality of the approximation. In a broader context, whether a constrained construction of the basis functions is preferable over a more flexible one, which might violate the pdf properties, depends on the specific problem at hand, as well as on factors such as the number and type of basis functions chosen.

The multivariate basis functions, $\Phi_j(\boldsymbol{x})$, are constructed via Kronecker product of the monodimensional ones. Once the number of basis for each dimension is chosen (i.e., $N_i, \forall i$), then the jth is computed as

$$\Phi_j(\boldsymbol{x}) = \prod_{i=1}^{n} \phi_{j,i}(x_i), \tag{4}$$

where the index of the multivariate basis (j) ranges from 1 to the product of all the monodimensional basis (i.e., N). Therefore, the total number of multivariate basis can be found as

$$N = \prod_{i=1}^{n} N_i. \tag{5}$$

In general, we expect Eq. (3) to be precise only for an infinite number of basis. However, in practice, a finite set of basis is often enough to represent the pdf at high levels of accuracy. The choice of basis functions is not a trivial task, and the chosen basis will be discussed in Section 4.

By leveraging the above-mentioned equations and by assuming that the diffusion tensor only depends on the state variables and does not directly depend on time, we can rewrite the Fokker–Planck equation as follows

$$\begin{aligned} \sum_{j=1}^{N} \Phi_j \tfrac{da_j}{dt} = \; & -\sum_{j=1}^{N} a_j \left(\sum_{i=1}^{n} \tfrac{\partial (f_i \Phi_j)}{\partial x_i} \right) + \\ & + \sum_{j=1}^{N} a_j \sum_{i=1}^{n} \sum_{t=1}^{n} \tfrac{\partial^2}{\partial x_i \partial x_t} (D_{it} \Phi_j). \end{aligned} \tag{6}$$

Then, we define the scalar product between two functions u and v, with respect to a weight function w, as

$$\langle u, v \rangle = \int_{\Omega_x} u(x) v(x) w(x) dx. \tag{7}$$

Finally, we apply Galerkin projection (Rowley et al., 2004) on Eq. (6), obtaining

$$\begin{aligned} \sum_{j=1}^{N} \langle \Phi_k, \Phi_j \rangle \tfrac{da_j}{dt} = \; & -\sum_{j=1}^{N} a_j \left\langle \Phi_k, \sum_{i=1}^{n} \tfrac{\partial (f_i \Phi_j)}{\partial x_i} \right\rangle + \\ & + \sum_{j=1}^{N} a_j \left\langle \Phi_k, \sum_{i=1}^{n} \sum_{t=1}^{N} \tfrac{\partial^2 (D_{it} \Phi_j)}{\partial x_i \partial x_t} \right\rangle, \end{aligned} \tag{8}$$

for $k = 1, \ldots, N$. As can be seen, as a result of projecting both sides of the FPE, we have managed to decouple the scalar product performed over the spatial support, from the contribution of the time-varying coefficients. Therefore, defining the following two $(N \times N)$ matrices

$$B_{kj} = \langle \Phi_k, \Phi_j \rangle \tag{9}$$

$$M_{kj} = \left\langle \Phi_k, -\sum_{i=1}^{n} \frac{\partial (f_i \Phi_j)}{\partial x_i} + \sum_{i=1}^{n} \sum_{t=1}^{n} \frac{\partial^2 (D_{it} \Phi_j)}{\partial x_i \partial x_t} \right\rangle, \tag{10}$$

Eq. (6) can be transformed into a set of N ordinary differential equations, where the spatial support and time contributions can be computed separately

$$B \dot{\boldsymbol{a}}(t) = M \, \boldsymbol{a}(t). \tag{11}$$

This represents a linear system of ordinary differential equations, whose solution can be computed in closed form from given initial conditions. For instance, by knowing the initial probability density function of the state: $p(\vec{x}, t_0)$, we can compute the coefficients at the initial time by projecting the spatial support onto the initial pdf, as follows

$$a_{0k} = \langle \Phi_k, p(\boldsymbol{x}, t_0) \rangle \; \forall k = 1, \ldots, N. \tag{12}$$

Once the initial conditions for the coefficients are computed, the time-varying coefficients at future times can be found by solving the following exponential matrix

$$\boldsymbol{a}(t) = e^{B^{-1} M t} \boldsymbol{a}_0, \tag{13}$$

where B^{-1} is the inverse of the B matrix. In the case of orthonormal basis functions, the B matrix becomes an identity matrix. An important aspect to mention is that this method is only applicable for cases in which both the diffusion tensor and the vector of deterministic forces do not explicitly depend on time.

3.2. Automatic Differentiation

Several ways exist for computing and evaluating derivatives of mathematical functions. Manual differentiation involves the manual computation and coding of derivatives: this can practically be very time-consuming and error-prone. On the other hand, numerical differentiation can be easily implemented, but only offers an approximation of the derivatives at the chosen points, whose accuracy is related to truncation and round-off errors. As a valid alternative to the abovementioned methods, symbolic differentiation can be employed. However, due to the derivative product rule, this technique can result in very complex

and difficult to handle expressions: something known as "expression swell" (Corliss, 1988). To balance both accuracy and control flow, researchers have developed a powerful technique known as automatic differentiation: a set of numerical techniques to exactly evaluate the derivative of a function specified through a computer program, at specific points (Griewank and Walther, 2003). It is important to stress that this technique does not return a symbolic form of the derivatives, but only its precise value at the requested data points.

In our work, the forward mode of automatic differentiation has been used for computing the partial derivatives of the dynamics arising from Eq. (10). A forward mode algorithm essentially breaks down the mathematical function as a computational graph, where each node represents an intermediate step of the computation with its corresponding derivative value: by applying the chain rule to the intermediate results, the corresponding derivative at the point of interest is generated. This method can easily be generalized for vector-valued functions, and it can be shown that it can be used for returning the Jacobian of a function with n independent variables, at given points. For a thorough description of these methods, the interested reader is referred to the literature (Clifford, 1871; Baydin et al., 2018).

The use of automatic differentiation through the JAX Python library allows us not only to very quickly compute all the derivatives at the points of interest, even for high-dimensional cases, but also parallelize the computations on CPU, GPU, or TPU, according to the user's preference, with small coding effort (Bradbury et al., 2018).

In terms of computational complexity, the cost of computing automatic differentiation depends on the chosen algorithm (e.g. forward or reverse mode), the number of dimensions of the problem, and the complexity of the function whose derivatives are taken. For more details on the computational cost of applying automated differentiation, the interested reader can refer to Margossian (2019).

4. Implementation

4.1. Basis Functions

As already discussed in Section 2, the basis functions constitute an important aspect of the proposed method. Better basis functions could mean a more accurate representation of the pdf with less computational cost, as well as fewer numerical instabilities.

In our case, the B-spline basis functions of order 3 were selected. This is related to the fact that these bases are semi-definite positive, which is a fundamental characteristic of the probability density function. Moreover, increasing the number of basis functions, in this case, only implies increasing the number of the same degree polynomials in a given interval, without actually increasing the degree of the polynomials. Finally, these basis functions allow for easily treating both nonperiodic and periodic boundary conditions (De Vylder, 1978).

An order k B-spline can be constructed by joining several polynomials of degree $k-1$. By choosing N nondescending breaking points and collecting them into a knot vector: $\vec{t} = (t_0, t_1, \dots, t_N)^T$, where $t_0 \leqslant t_1 \leqslant t_2 .. \leqslant t_N$, we can construct the B-spline basis functions as follows

$$N_{i,k}(t) = \frac{t - t_i}{t_{i+k-1} - t_i} N_{i,k-1}(t) + \frac{t_{i+k} - t}{t_{i+k} - t_{i+1}} N_{i+1,k-1}(t), \tag{14}$$

where

$$N_{i,1}(t) = \begin{cases} 1 & \text{for } t_i \leqslant t \leqslant t_{i+1} \\ 0 & \text{otherwise.} \end{cases} \tag{15}$$

The first and second derivatives of the B-spline can also be analytically computed

$$\frac{dN_{i,k}(t)}{dt} = \frac{k-1}{t_{i+k-1} - t_i} N_{i,k-1}(t) - \frac{k-1}{t_{i+k} - t_{i+1}} N_{i+1,k-1}(t) \tag{16}$$

$$\begin{aligned} \frac{d^2 N_{i,k}(t)}{dt^2} = & \frac{k-1}{t_{i+k-1} - t_i} \frac{k-1}{t_{i+k-2} - t_i} N_{i,k-2}(t) + \\ & - \frac{k-1}{t_{i+k-1} - t_i} \frac{k-2}{t_{i+k-1} - t_{i+1}} N_{i+1,k-2}(t) + \\ & - \frac{k-1}{t_{i+k} - t_{i+1}} \frac{k-2}{t_{i+k-1} - t_{i+1}} N_{i+1,k-2}(t) + \\ & + \frac{k-1}{t_{i+k} - t_{i+1}} \frac{k-2}{t_{i+k} - t_{i+2}} N_{i+2,k-2}(t). \end{aligned} \tag{17}$$

In our work, we select the basis functions of degree $k = 3$.

As already discussed, the multivariate case is generated through tensor products of monodimensional basis functions. As shown in Eq. (4), this means that each basis function can be constructed as a product of basis functions associated with each dimension. These basis functions have a very appealing property that makes the matrices M and B sparse, and allows less and faster computations. By expressing the multivariate basis functions Φ_j and Φ_k as shown in Eq. (4), it can be shown that whenever a pair of indexes of the same basis (i.e., j_i and k_i) exists, whose difference is higher than 2, then the product between two basis functions, or their first or second derivatives, of indexes j and k is zero, as one can deduct by looking at Eq. (14), for $k = 3$. Mathematically, this can be expressed as follows

$$\begin{cases} \Phi_k \Phi_j & = 0 \\ \Phi_k \sum_{i=1}^{n} \frac{\partial \Phi_j}{\partial x_i} & = 0 \\ \Phi_k \sum_{i=1}^{n} \sum_{t=1}^{n} \frac{\partial^2 \Phi_j}{\partial x_i \partial x_t} & = 0 \end{cases} \quad \text{if} j_i,\ k_i \ \exists\ s.t.\ |j_i - k_i| \geqslant 3.$$

As shown in Eq. (8), when the above terms are zero, the corresponding B_{kj} and M_{kj} matrix elements are zero as well. This allows us to greatly reduce the number of computations and avoid the rapid insurgence of the curse of dimensionality. In fact, the relative number of zero elements w.r.t. non-zero ones increases as the matrix size increases (since the upper bound of the k_{x_i} and j_{x_i} indexes enlarges). While

this is partially alleviated by the increased number of zeros in the B and M matrices, it is important to point out that constructing the multivariate basis functions as products of monovariate basis, causes the number of basis functions to grow exponentially as the number of dimensions increases. Therefore, depending on the application and the number of basis functions required to approximate the dynamical system of interest, it might be necessary to change how basis functions are constructed (e.g. selecting orthogonal basis functions).

4.2. Integrals computation

As shown in Section 8, the Galerkin projection involves the computation of high-dimensional integrals between the basis functions and the left and right-hand sides of the FP equation. The set of integrals to be computed is shown in Eq. (10): while the ones associated with the B matrix can easily be reduced to products between one-dimensional integral (due to the fact that the basis functions are separable), the ones associated with the M_{kj} matrix have high-dimensionality (unless the dynamics of the system is also separable). Furthermore, these integrals cover the full domain, which makes their computation even harder. In this work, in order to perform numerical integration we use a quasi-Monte Carlo method. Our goal is to approximate a multi-dimensional integral over an n-dimensional domain

$$I(f) = \int_{\Omega} f(\boldsymbol{x}) d\boldsymbol{\Omega}, \tag{18}$$

While several methods exist for approximating this integral (e.g., quadrature rule, Monte Carlo methods, sparse grid methods, etc.), we have opted for a quasi-Monte Carlo scheme (Dick et al., 2013). This is an equal-weight quadrature rule similar to Monte Carlo methods, where the above-mentioned integral is approximated as follows

$$I(f) \approx \frac{V}{n} \sum_{i=0}^{n-1} f(\boldsymbol{x}_i), \tag{19}$$

V is the volume of the n-dimensional cube defined by the intervals of integration. The difference is that in quasi-Monte Carlo methods, the points $\boldsymbol{x}_0, \boldsymbol{x}_1, .., \boldsymbol{x}_{n-1}$ are chosen deterministically in order to obtain guaranteed error bounds and a better convergence speed w.r.t. random grids. Several possibilities exist for selecting the points such as the van der Croput sequence, Kronecker sequence, and Hammersley point set. In this work, the Halton sequence has been used: this enables us to compute the abovementioned high-dimensional integrals without using a number of samples as high as pure Monte Carlo methods.

4.3. Probability metrics

The solution to the FP equation is a probability density function. Therefore, any method that attempts to solve this equation in an approximate way needs to, first of all, have a ground truth pdf to compare against (since the FP is not solvable in closed form, in general) and, secondly, have some metrics that evaluate how well the approximated pdf is representing the underlying distribution. We reconstruct the "true" pdf by solving the underlying stochastic differential equation for many samples, and reconstructing from them (with a fit of multivariate kernels of Gaussian distributions) the evolving pdf at each timestamp. However, it is also necessary to find ways to compare the approximated and "true" pdf. One idea could be to use the first k-moments of the distribution, but the problem is that these do not fully capture the pdf complexity. Instead, probability metrics such as distance and divergence metrics allow us to assess how close two probability density functions are. The definition and use of distance and divergence metrics are crucial in many statistics fields that involve assessing how well a given model approximates the real probability density function. A probability metric is, by definition, a measure that quantifies how dissimilar two random quantities (i.e., two probability measures) are: this can be quantified in two ways. The first one is by employing distance measures (e.g. Hellinger distance, Bhattacharyya distance, Wasserstein-Kantorovich distance, etc.) that generalize the general concept of metric spaces to probability distributions. On the other hand, a weaker but similar concept to distance measures is divergence, which is a form of distance measure between probability distributions but does not necessarily satisfy either the symmetric property or the triangle inequality. In this work, we use the Hellinger distance and KL-divergence for quantitatively assessing the proximity of probability distributions. By denoting as p and q the reference and approximated discrete density functions, respectively, the Hellinger distance can be defined as

$$H(p,q) = \frac{1}{\sqrt{2}} \sqrt{\sum_{\vec{x} \in \Omega} \left(\sqrt{p(\boldsymbol{x})} - \sqrt{q(\boldsymbol{x})}\right)^2}. \tag{20}$$

Interestingly, the Hellinger distance satisfies the property: $0 \leqslant H(p,q) \leqslant 1$, being zero if the two distributions are identical. On the other hand, the KL divergence can be computed as (Kullback, 1997)

$$D_{KL}(p||q) = \sum_{\vec{x} \in \Omega} p(\boldsymbol{x}) \log\left(\frac{p(\boldsymbol{x})}{q(\boldsymbol{x})}\right). \tag{21}$$

In order to interpret the values of these metrics, it has to be ensured that the pdf has the property of a probability density function (i.e., it is always nonnegative and it integrates to one). Hence, if these properties are not fulfilled by the approximated pdf, it is first essential to normalize and post-process the pdf to meet these requirements.

5. Experiments

5.1. Stochastic dynamics: 2D test case

As a stochastic dynamics test case, we chose the harmonic oscillator with damping and noise, whose equations of motion are (Zorzano et al., 1999)

$$\dot{x} = v$$
$$\dot{v} = -Kx - \gamma v + \sqrt{2\sigma}\eta(t),$$

where $\eta(t)$ is the time derivative of the Wiener process, which is demonstrated to be a white noise with mean: $\mathbb{E}[\eta(t)] = 0$ and variance: $\mathbb{E}[\eta(t)\eta(s)] = \delta(t-s)$, where δ is the Dirac delta function (Evans, 2012). For this dynamical system, the FP can be written as

$$\frac{\partial p}{\partial t} = \left(-\frac{\partial v}{\partial x}\right)p + \left(\frac{\partial}{\partial v}(\gamma v + Kx) + \sigma\frac{\partial^2}{\partial v^2}\right)p \tag{22}$$

$$\frac{\partial p}{\partial t} = \left(-\frac{\partial (vp)}{\partial x}\right) + \left(\frac{\partial}{\partial v}[(\gamma v + Kx)p] + \sigma\frac{\partial^2 p}{\partial v^2}\right) \tag{23}$$

We solved the Fokker–Planck equation with two sets of basis (36 and 40 for each dimension) and by choosing the following parameters: $K = 1, \gamma = 2.1, \sigma = 0.08, x_0 = -4, v_0 = 0.001, t_0 = 0.95, t_f = 3$; the results are shown in Fig. 1 and compared against a Monte Carlo run with 500 thousand samples (considered as ground truth). In the top row, we show the time evolution of the first moments of position (on the left) and velocity (on the right) for both cases. While on the second row, the evolution of the second moments of the distribution is reported. Then, in the third row, the Hellinger and KL-divergence evolution are displayed (computed w.r.t. the MC solution) and in the forth row both the time evolution of the integral of the probability density function and the 3D plot of the initial and final pdf's are plotted. As we can observe, the method manages to capture both the drift and diffusive behavior of the dynamics, while accurately representing the first two moments of the distributions. Moreover, the Hellinger distance, KL divergence, and the integral of the probability density function show that the method is able to represent the pdf across time, although, in this case, the accuracy degrades as time passes. As observed, the approximated pdf does not integrate perfectly to one: hence, as explained already in Section 4.3, it is first essential to normalize the pdf, before computing the Hellinger distance and KL-divergence. We performed this post-processing step in this and all the other experiments performed in this work. The degradation of the quality of the pdf as a function of time can be alleviated by increasing the number of basis functions: 40 basis functions maintain the error at lower levels than 36. Finally, in the plot on the bottom-right part, we show the initial (in red) and final (in blue) probability density functions. As can be seen, the final distribution has drifted away from the initial one, while the velocity standard deviation has evidently increased.

5.2. Deterministic dynamics with uncertain initial conditions

Having already introduced this approach for deterministic dynamics on one and two-dimensional cases in our previous work, we here focus on a three-dimensional case (Acciarini et al., 2020). We consider the motion of a satellite in LEO, expressed in terms of averaged equations of motion in modified equinoctial elements. These equations allow us to express the average variation of the orbital elements for long periods of time when the satellite motion is affected by the central gravity term and various disturbances. In this case, we consider that the drag term is the only disturbing force acting on the satellite, and we assume that its component is only directed along the radial and along-track components of the satellite motion. With these assumptions, the following equations of motion apply

$$\begin{aligned}
\frac{da}{dt} &= \frac{1}{2\pi}\int_{-\pi}^{\pi} \frac{(1-P_1^2-P_2^2)^2}{(1+P_2\sin L+P_1\cos L)^2}\frac{1}{\sqrt{1-P_1^2-P_2^2}} \\
&\quad \frac{2a^2}{h}\left[(P_2\sin L - P_1\cos L)a_r + \right. \\
&\quad \left. + \frac{p}{r}a_\theta\right]dL \\
\frac{dP_1}{dt} &= \frac{1}{2\pi}\int_{-\pi}^{\pi} \frac{(1-P_1^2-P_2^2)^2}{(1+P_2\sin L+P_1\cos L)^2}\frac{1}{\sqrt{1-P_1^2-P_2^2}} \\
&\quad \frac{r}{h}\left\{-\frac{p}{r}\cos L a_r + \right. \\
&\quad + \left[P_1 + (1+\frac{p}{r})\sin L\right]a_\theta + \\
&\quad \left. - P_2[Q_1\cos L - Q_2\sin L]a_h\right\}dL \\
\frac{dP_2}{dt} &= \frac{1}{2\pi}\int_{-\pi}^{\pi} \frac{(1-P_1^2-P_2^2)^2}{(1+P_2\sin L+P_1\cos L)^2}\frac{1}{\sqrt{1-P_1^2-P_2^2}} \\
&\quad \frac{r}{h}\left\{\frac{p}{r}\sin L a_r + \right. \\
&\quad + \left[P_2 + (1+\frac{p}{r})\cos L\right]a_\theta + \\
&\quad \left. + P_1(Q_1\cos L - Q_2\sin L)a_h\right\}dL,
\end{aligned} \tag{24}$$

where the following relations hold

$$\frac{p}{r} = 1 + P_1\sin L + P_2\cos L$$
$$\frac{r}{h} = \frac{h}{\mu(1+P_1\sin L + P_2\cos L)}$$
$$h = nab$$
$$b = a\sqrt{1-P_1^2-P_2^2}.$$

Furthermore, a_r and a_θ are the radial and transverse (i.e., perpendicular to r and in the orbital plane) components of the perturbing acceleration. As shown in (Carlo et al., 2017), by assuming a zero wind velocity and that the aerodynamic forces only act in the orbital plane and in the opposite direction w.r.t. the velocity vector of the spacecraft, then drag acceleration can be written as

$$a_r = \frac{1}{2}\rho C_d\frac{A}{m}\frac{\mu}{a}\left(\frac{2\Phi}{b^2} - 1\right)\frac{(P_2\sin L - P_1\cos L)}{D} \tag{25}$$

$$a_\theta = \frac{1}{2}\rho C_d\frac{A}{m}\frac{\mu}{a}\left(\frac{2\Phi}{b^2} - 1\right)\frac{(1+P_1\sin L + P_2\cos L)}{D} \tag{26}$$

$$a_h = 0, \tag{27}$$

where

$$\Phi = 1 + P_1\sin L + P_2\cos L$$
$$D = \sqrt{1 + P_1^2 + P_2^2 + 2(P_2\cos L + P_1\sin L)}.$$

We assumed uncertainty in the initial conditions expressed as a multivariate Gaussian distribution with mean: $\boldsymbol{\mu}(t_0) = [\mu_a(t_0), \mu_{P1}(t_0), \mu_{P2}(t_0)] = [6665.15\text{km}, 0\text{rad}, 0\text{rad}]$

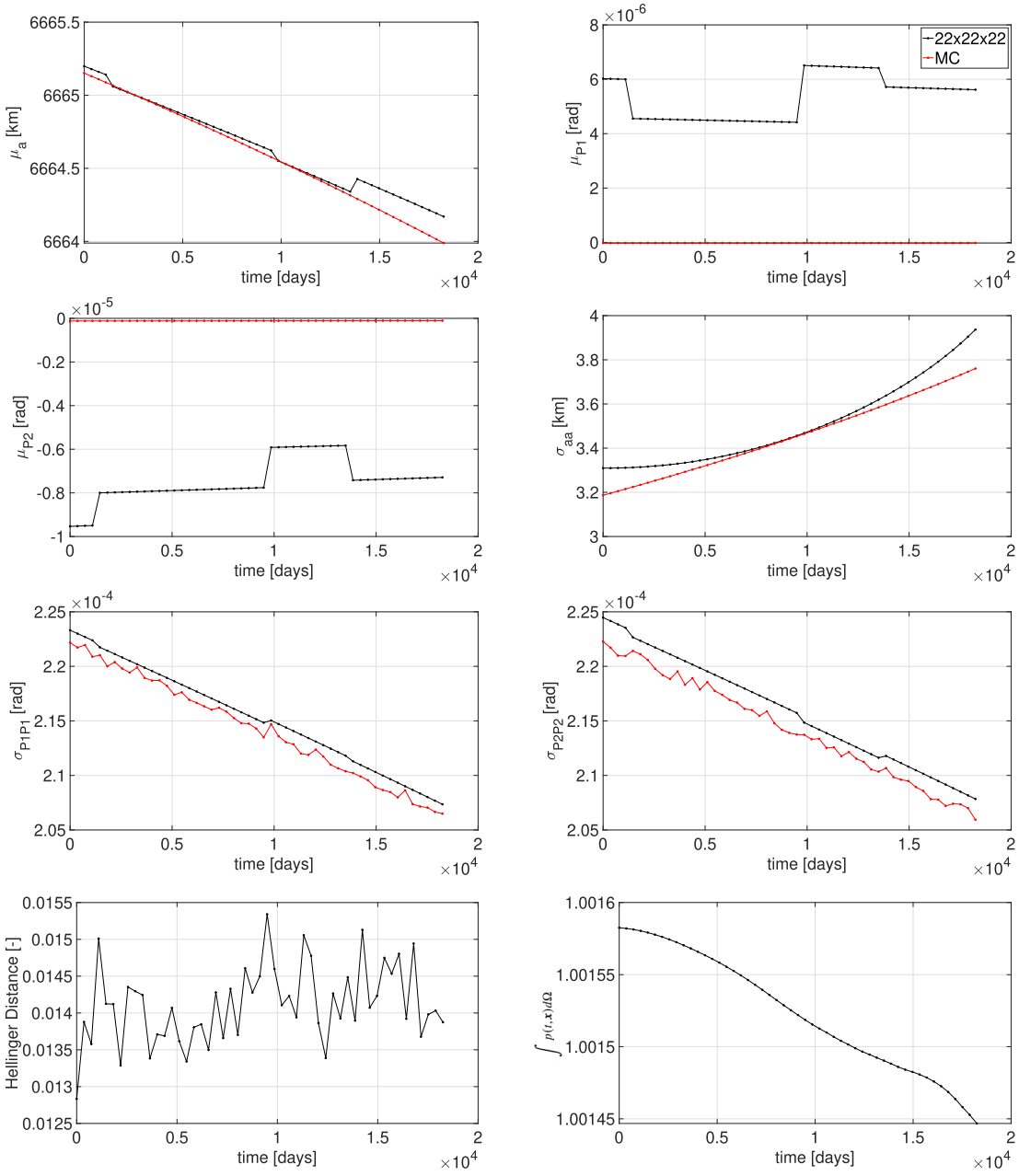

Fig. 2. *First, second and third rows*: comparison between mean and diagonal elements of the square root of the diagonal terms of the covariance matrix using Monte Carlo and approximated via Galerkin projection of the FP; *fourth row*: Hellinger distance between the two distributions and integral of the approximated pdf via Galerkin projection as a function of time.

and a diagonal covariance matrix with values $\sigma_{aa}(t_0) = 3.189$ km, $\sigma_{P1P1}(t_0) = 10^{-4}$ rad, $\sigma_{P2P2}(t_0) = 10^{-4}$ rad. We used 22 basis functions for each of the three dimensions, and we then propagated the dynamics, for 50 years, with a $\rho C_d A/m$ value of 10^{-6}m^{-1}. In the first three rows of Fig. 2, we show the evolution of the first and second moments of the distribution (i.e., mean values and diagonal terms of covariance matrix). We compare the results against a Monte Carlo run with one and a half million samples. As we can see, the method manages to accurately capture both the trend and values. For instance, in the case of the semi-major axis, at the end of the propagation, there is an error of fewer than 200 meters on the estimated mean and less than 15 meters in the estimated covariance. Finally, in the fourth row of Fig. 2, we show the Hellinger distance and the integral of the approximated distribution as a function of time. As we can observe, the integral of the distribution is always maintained within 0.2% of its true value (i.e., 1). An interesting aspect is that the integral value improves over time and that the Hellin-

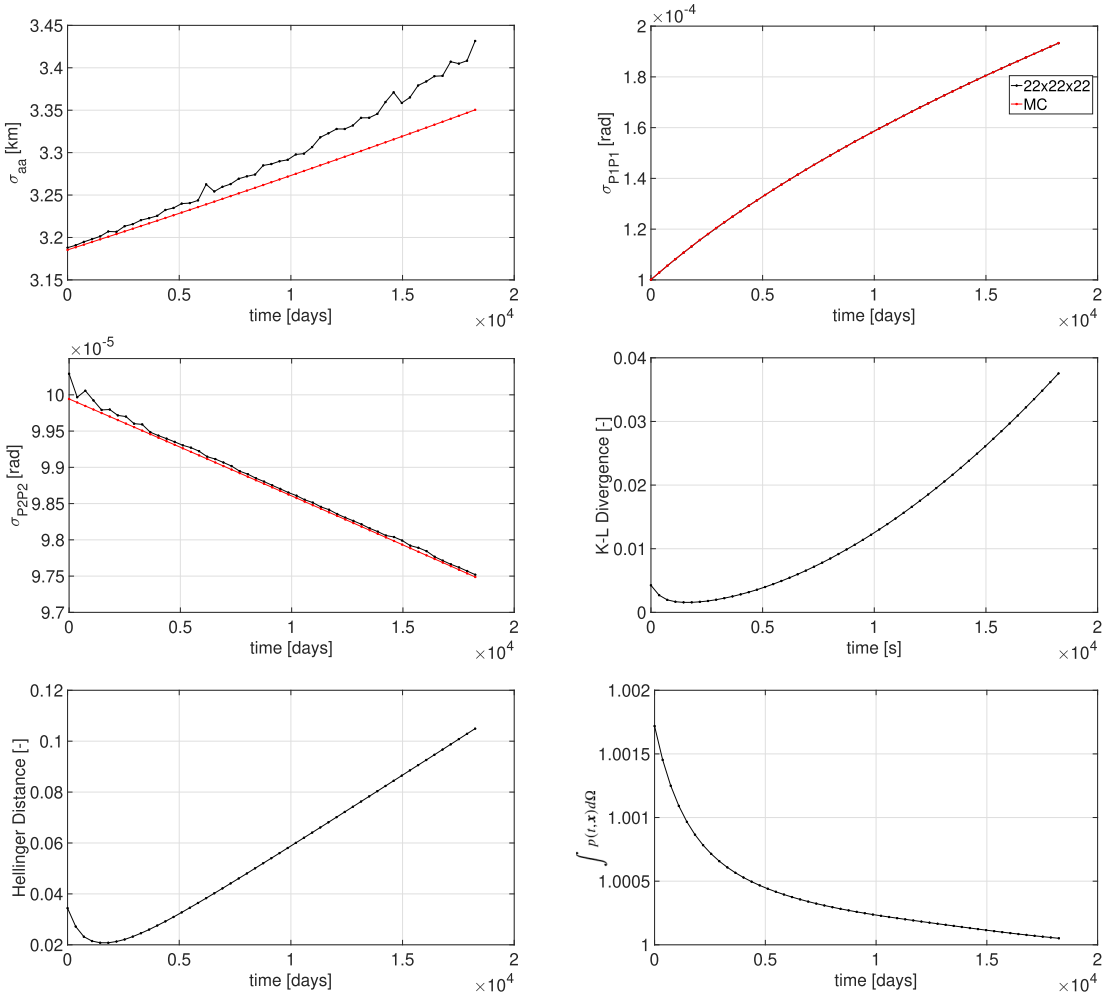

Fig. 3. *First row and left figure of second row*: square root of the diagonal elements of the covariance matrix for the pdf reconstructed via Monte Carlo sampling and Galerkin projection, as a function of time; *right figure of second row*: KL-divergence of the two pdfs as a function of time; *third row*: Hellinger distance between the two pdfs and integral of the approximated pdf via Galerkin projection as a function of time.

ger distance does not steadily increase (as would happen for most uncertainty propagation methods, where the error grows with time). This is because, in this method, the quality of the propagation does not depend on the time, but only on the basis functions' ability to approximate the evolved distribution.

5.3. Stochastic dynamics with uncertain initial conditions

As a final experiment, we also investigate the same dynamical system presented in Section 5.2, but with a diffusive term in the second equation of motion

$$\frac{dP_1}{dt} = \frac{dP_1}{dt}|_d + \sqrt{2\sigma}\eta(t),$$

where $\sqrt{2\sigma} = 1.25 \times 10^{-6}$, $dP_1/dt|_d$ refers to the expression of the derivative term in the deterministic system shown in Eq. (24), and $\eta(t)$ refers to the white noise with zero mean, as already explained for the 2D case in Section 5.2. We use the same basis functions, parameters, and initial conditions as the ones used in Section 5.2, increasing the number of Monte Carlo samples to one million (to better represent the introduced diffusive behavior). Similarly to what has been done for the other experiments, we compare the approximated results with the pdf reconstructed via Monte Carlo. In Fig. 3, we show the results in terms of the square root of the diagonal elements of the covariance matrix as a function of time, as well as the Hellinger distance, KL-divergence, and integral of the pdf as a function of time. As we can see, also in this case the approximated pdf manages to approximate well the underlying ground truth pdf, also maintaining an approximation of the integral within 0.2% of the true value. Finally, as we would expect, the diffusive term triggers an increase in the covariance element associated with the P_1 variable, which is the variable directly affected by diffusion (this was not happening in Section 5.2 without the diffusive term, where the covariance was instead decreasing).

6. Conclusions

In this work, we have applied a known method to solve the Fokker–Planck equation for the time propagation of probability density functions that quantify the uncertainty of the state and/or the dynamics in nonlinear dynamical systems, with a focus on orbital dynamics. We have showed that the discussed method has several advantages, which can be pivotal in many orbital dynamics applications.

First of all, once the number of basis functions is established and several matrices of integrals are computed, one can propagate any initial probability density function at any given future time, without added computational costs. The only requirement for the propagation to maintain its accuracy is that the given basis can approximate well the pdf at future times, in the prescribed spatial domain. This means that one can pre-compute the matrix of integrals once, and then store those values and use them for real-time uncertainty propagation applications, for approximating the pdf evolution at future times, with very little computational costs. This is fundamentally different from most of the numerical and simulation-based techniques, which force users to perform from scratch most of the computations, for different initial conditions and whose computation time is proportional to the time horizon of the simulation (the longer the propagation time, the slower the computational time). Secondly, we discussed that B-spline basis functions have the interesting property of making the matrix of the integrals highly sparse, therefore consistently reducing the computational burden. Moreover, such sparsity increases as the number of basis functions are increased, therefore allowing to delay the insurgence of the curse of dimensionality. Then, from the implementation perspective, we formulated the problem dynamics in terms of differentiable expressions and made use of automatic differentiation, which allows to quickly and exactly compute first and high order partial derivatives. Finally, we extended our previous preliminary work by presenting results for nonlinear diffusive dynamics and nonlinear three-dimensional deterministic dynamics with uncertain initial conditions, in the context of orbital mechanics. The results show that this method can successfully be used to solve the Fokker–Planck equation and propagate the full probability density function of the state (and not only the first two moments). The work is available open-source at https://github.com/Sceki/fpe_orbital_dynamics and we encourage more practitioners and researchers in the field of astrodynamics and celestial mechanics to consider and improve upon this method (especially for high dimensional cases).

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.

References

Acciarini, G., Greco, C., Vasile, M., 2020. On the solution of the fokker-planck equation without diffusion for uncertainty propagation in orbital dynamics. In: 2020 AAS/AIAA Astrodynamics Specialist Conference.

Armellin, R., Di Lizia, P., Bernelli-Zazzera, F., et al., 2010. Asteroid close encounters characterization using differential algebra: the case of apophis. Celest. Mech. Dynam. Astron. 107 (4), 451–470. https://doi.org/10.1007/s10569-010-9283-5.

Baydin, A.G., Pearlmutter, B.A., Radul, A.A., et al., 2018. Automatic differentiation in machine learning: a survey. J. Mach. Learn. Res. 18.

Bradbury, J., Frostig, R., Hawkins, P., et al., 2018. Jax: composable transformations of python+ numpy programs. Version 0.2, 5, 14–24.

Clifford, M., 1871. Preliminary sketch of biquaternions. Proc. London Math. Soc. 1 (1), 381–395.

Corliss, G.F., 1988. Applications of differentiation arithmetic. In: Reliability in Computing. Elsevier, pp. 127–148.

De Vylder, F., 1978. A practical solution to the problem of ultimate ruin probability. Scand. Actuar. J. 1978 (2), 114–119. https://doi.org/10.1080/03461238.1978.10419484.

DeMars, K.J., Bishop, R.H., Jah, M.K., 2013. Entropy-based approach for uncertainty propagation of nonlinear dynamical systems. J. Guid., Control, Dynam. 36 (4), 1047–1057. https://doi.org/10.2514/1.58987.

Di Carlo, M., Martin, J.M.R., Vasile, M., 2017. Automatic trajectory planning for low-thrust active removal mission in low-earth orbit. Adv. Space Res. 59 (5), 1234–1258. https://doi.org/10.1016/j.asr.2016.11.033.

Dick, J., Kuo, F.Y., Sloan, I.H., 2013. High-dimensional integration: the quasi-monte carlo way. Acta Numer. 22, 133. https://doi.org/10.1017/S0962492913000044.

Evans, L.C., 2012. An introduction to stochastic differential equations, volume 82. American Mathematical Soc.

Gardiner, C., 2009. Stochastic methods, volume 4. Springer, Berlin, pp. 117–174.

Griewank, A., Walther, A., 2003. Introduction to automatic differentiation. PAMM: Proceedings in Applied Mathematics and Mechanics, volume 2. Wiley Online Library, pp. 45–49. https://doi.org/10.1002/pamm.200310012.

Itô, K., 1951. On stochastic differential equations, 4. American Mathematical Soc, pp. 1–4.

Jones, B.A., Weisman, R., 2019. Multi-fidelity orbit uncertainty propagation. Acta Astronaut. 155, 406–417. https://doi.org/10.1016/j.actaastro.2018.10.023.

Kullback, S., 1997. Information theory and statistics. Courier Corporation, pp. 6–7.

Kumar, M., Chakravorty, S., Junkins, J.L., 2010. A semianalytic meshless approach to the transient fokker–planck equation. Probab. Eng. Mech. 25 (3), 323–331. https://doi.org/10.1016/j.probengmech.2010.01.006.

Kumar, P., Narayanan, S., 2006. Solution of fokker-planck equation by finite element and finite difference methods for nonlinear systems. Sadhana 31 (4), 445–461. https://doi.org/10.1007/BF02716786.

Luo, Y.-Z., Yang, Z., 2017. A review of uncertainty propagation in orbital mechanics. Prog. Aerosp. Sci. 89, 23–39. https://doi.org/10.1016/j.paerosci.2016.12.002.

Margossian, C.C., 2019. A review of automatic differentiation and its efficient implementation. Wiley Interdiscip. Rev.: Data Mining Knowledge Discov. 9 (4), e1305.

McLachlan, G.J., Lee, S.X., Rathnayake, S.I., 2019. Finite mixture models. Annu. Rev. Stat. Its Appl. 6, 355–378.

Risken, H., 1996. Fokker-planck equation. In: The Fokker-Planck Equation. Springer, pp. 63–95.

Rowley, C.W., Colonius, T., Murray, R.M., 2004. Model reduction for compressible flows using pod and galerkin projection. Physica D 189 (1–2), 115–129. https://doi.org/10.1016/j.physd.2003.03.001.

Servadio, S., Parker, W., Linares, R., 2023. Uncertainty propagation and filtering via the koopman operator in astrodynamics. J. Spacecr. Rock., 1–17

Sun, Y., Kumar, M., 2016. Uncertainty propagation in orbital mechanics via tensor decomposition. Celestial Mech. Dynam. Astron. 124 (3), 269–294. https://doi.org/10.1007/s10569-015-9662-z.

Zorzano, M.P., Mais, H., Vazquez, L., 1999. Numerical solution of two dimensional fokker—planck equations. Appl. Math. Comput. 98 (2–3), 109–117. https://doi.org/10.1016/S0096-3003(97)10161-8.