UNCERTAINTY PROPAGATION MEETING SPACE DEBRIS NEEDS

Jorge Bravo Aguado[(1)], Javier López Santiago[(1)], Alberto López Yela[(1)], Pablo Martínez Olmos[(1)], Joaquín Míguez Arenas[(1)], David Morante González[(1)], Manuel Sanjurjo-Rivo[(1)], Manuel Vázquez López[(1)], Alejandro Cano[(2)], Diego Escobar[(2)], Efrén Villanueva Bonay[(2)], Pau Gago Padreny[(2)], Benjamin Bastida Virgili[(3)], Stijn Lemmens[(3)], Klaus Merz[(3)], and Jan Siminski[(3)]

[(1)]*Universidad Carlos III de Madrid, Av. de la Universidad 30, 28911 Leganés, Madrid, Spain, jobravoa@pa.uc3m.es, jalopezs@ing.uc3m.es, alyela@math.uc3m.es, pamartin@ing.uc3m.es, jmiguez@ing.uc3m.es, dmorante@ing.uc3m.es, manuel.sanjurjo@uc3m.es, mavazque@ing.uc3m.es*

[(2)]*GMV, Calle Isaac Newton 11, 28670 Tres Cantos, Madrid, Spain, alcano@gmv.com, descobar@gmv.com, evillanueva@gmv.com, pau.gago.padreny@gmv.com*

[(3)]*Space Debris Office, OPS-SD, ESA/ESOC, Robert-Bosch-Str. 5, 64293 Darmstadt, Germany, Benjamin.Bastida.Virgili@esa.int, Stijn.Lemmens@esa.int, Klaus.Merz@esa.int, Jan.Siminski@esa.int*

ABSTRACT

The provision of quality space debris services largely relies in the capacity to estimate the state vector of the space debris objects and a suitable description of its uncertainty. In recent years, emphasis has been placed on the need to improve realism in characterizing uncertainty to enhance the quality of space surveillance and tracking (SST) services. The determination of the orbit of a debris object, however, is a tracking problem with scarce available measurements. For that reason, the propagation of the uncertainty from one observation to the next one is key to track the debris accurately enough. This work addresses the analysis of uncertainty propagation methods for space debris orbit determination. An assortment of approaches are detailed, defined and explored, to cover an extensive variety of methodologies. Some of them are studied in more detail and tuned to provide the proper metrics of interest in two relevant space debris services: reentry prediction and collision avoidance. A series of pertinent study cases are analyzed for both applications and, as a result recommendations are made to improve the uncertainty characterization in SST-related problems. A subset of the investigated methodologies has been implemented and tested in a software prototype.

Keywords: Uncertainty propagation; uncertainty quantification; reentry; collision avoidance.

1. INTRODUCTION

Space debris objects are routinely tracked to provide services of re-entry and conjunction analysis, among others. Estimation and prediction of the state of the space debris is at the core of such space surveillance and tracking operations. The performance of the estimation process can be measured in terms of the correct characterization of the state uncertainty, which is, in turn, necessary for the proper operation of the space tracking system. When tracking space objects, observations are not in abundance, and, therefore, propagation of the uncertainty from the epoch of the last observation update is key for efficient tracking. A thorough review of the relevance of uncertainty realism in SST-related problems has been recently highlighted and presented by a *Working Group on Covariance Realism* [32]. In the last years, there has been a considerable effort to improve uncertainty quantification methods and enhance SST-services.

This paper explores various families of methods for uncertainty propagation (UP) to meet the needs of state-of-the-art debris tracking systems. The work has been carried out in the framework of the ESA contract of the same name and it has *not* been limited to the study of methods previously used in orbit determination problems. Indeed, it broadens scope to explore UP schemes devised for other tracking problems. The focus is placed on nonlinear methods, i.e., those in which the non-linearity of the dynamics is explicitly taken into account. Nonlinear UP techniques can be classified in three broad groups:

1) Probabilistic UP methods, that propagate the probability distribution of the state without regard to the specifics of the dynamic equation. This category includes most available schemes: unscented transform ([24]) and cubature methods in general ([26, 4]), or Gaussian mixture (GM) models ([37, 40, 14, 15]).

2) Dynamics-based UP methods, (essentially) includes state transition tensors (STTs) ([31, 19]), differential algebra (DA) propagators ([39, 7, 43]) and polynomial chaos expansions ([22, 23]).

3) Hybrid UP methods, that aim at drawing from the advantages of different schemes (probabilistic or dynamics-based) by combining them, e.g., GM models combined with STTs ([18]) or with a polynomial chaos expansion ([41]).

In addition, stochastic and deterministic dynamical models, including a stochastic description of the atmospheric density, have been tested for assessing their influence on the overall performance of the UP methods.

A series of study cases are selected to compare all the methods, taking into account both their computational cost and their accuracy (which is assessed by comparing with "brute force" Monte Carlo propagation). From the preliminary results obtained in the study, a selection of suitable UP methods is made. They are then used in two prominent applications: satellite collision risk assessment and re-entry prediction. For these two problems, the mapping of uncertainties into the quantities of interest, i.e., collision probability or re-entry time and location uncertainty, is also explored. Different techniques are proposed and specific metrics are defined for their performance assessment. A series of test cases are studied for both collision avoidance and re-entry analysis. Selected methods have been implemented in a software prototype.

The paper is organized as follows: Section 2 is a review of the UP methods that have been studied in this work; Section 3 describes how some of these methods can be used in the context of reentry prediction and collision avoidance services; Section 4 briefly describes the software prototype; Section 5 illustrates some relevant results of the analysis; finally, Section 6 is devoted to the conclusions drawn from this work.

2. UNCERTAINTY PROPAGATION METHODS

For generality, the object motion is assumed to be governed by a stochastic differential equation of the form

$$\mathsf{d}\mathbf{X} = f(\mathbf{X}, t)\mathsf{d}t + \mathbf{G}(\mathbf{X}, t)\mathsf{d}\mathbf{W}, \tag{1}$$

where t is continuous time, $\mathbf{X}(t)$ is the state of the object (a multivariate stochastic process), the drift function $f(\mathbf{X}, t)$ is deterministic, $\mathbf{G}(\mathbf{X}, t)$ is a known matrix that determines the power of the random perturbation and $\mathbf{W}(t)$ is a multivariate standard Wiener process [30]. This formulation allows us to include in a seamless way a stochastic model for drag and also covers the usual deterministic framework by simply taking $\mathbf{G}(\mathbf{X}, t) = \mathbf{0}$, which reduces (1) to an ordinary differential equation (ODE).

The numerical integration of an equation of the form in (1) involves the discretisation of the continuous-time variable t. Several schemes can be used for this purpose. Most of them translate the continuous-time SDE (1) into a discrete-time difference equation of the form

$$\mathbf{X}_n = \mathsf{f}(\mathbf{X}_{n-1}, \mathbf{U}_n), \tag{2}$$

where T is the integration step, $\mathbf{X}_n \approx \mathbf{X}(nT)$, $\mathbf{U}_n$ is a zero-mean Gaussian random vector with covariance matrix $T\mathbf{G}_{n-1}\mathbf{G}_{n-1}^\top$ and $\mathbf{G}_{n-1} = \mathbf{G}(\mathbf{X}_{n-1}, (n-1)T)$. For the simple Euler-Maruyama scheme,

$$\mathsf{f}(\mathbf{X}_{n-1}, \mathbf{U}_n) = f(\mathbf{X}_{n-1}) + \mathbf{U}_n.$$

2.1. Monte Carlo methods

Given a prior probability density function (pdf) for the state at discrete time $n = 0$, denoted $p_0(\mathbf{x}_0)$, eq. (2) generates random sequences $\mathbf{X}_0, \mathbf{X}_1, \ldots, \mathbf{X}_n, \ldots$ For conciseness, let us represent the subsequence between discrete time instants k and m as $\mathbf{X}_{k:m} := \{\mathbf{X}_k, \mathbf{X}_{k+1}, \ldots, \mathbf{X}_m\}$. One can perform standard Monte Carlo (MC) sampling for the dynamical system (2) for $n = 0, \ldots, m$, which generates N independent and identically distributed (iid) subsequences $\{\mathbf{X}_{0:m}^{(i)}\}_{i=1}^N$. Under very mild assumptions on eq. (1), if $\mathbf{X}_0$ has a pdf $p_0(\mathbf{x}_0)$ then the random vector $\mathbf{X}_m$ is distributed according to a well-defined pdf $p_m(\mathbf{x}_m)$. It is often of interest to numerically approximate moments of $\mathbf{X}_m$. The standard MC estimator, $\mathbb{E}^N[\phi(\mathbf{X}_m)] = \frac{1}{N}\sum_{i=1}^N \phi(\mathbf{X}_m^{(i)})$, where $\phi(\cdot)$ is an integrable test function, can be used for that purpose, with a convergence rate of $\mathcal{O}(N^{-\frac{1}{2}})$ [33].

The set of MC samples $\{\mathbf{X}_{0:m}^{(i)}\}_{i=1}^N$ can also be used to construct a kernel density estimator (KDE) of the pdf $p_m(\mathbf{x}_m)$, for any integer $m < \infty$. The KDE is built by taking the convolution of a kernel function $\kappa : \mathcal{X} \mapsto [0, \infty)$, where $\mathcal{X}$ is the state space, with the sequence $\mathbf{X}_m^{(1)}, \ldots, \mathbf{X}_m^{(N)}$. To be specific, the KDE of $p_m(\mathbf{x})$ is

$$\hat{p}_m^N(\mathbf{x}) := \frac{1}{N}\sum_{i=1}^N \kappa\left(\mathbf{x} - \mathbf{X}_m^{(i)}\right). \tag{3}$$

The kernel is itself a centred pdf with finite second order moments, i.e., $\int_{\mathcal{X}} \|\mathbf{x}\|^2 \kappa(\mathbf{x}) d\mathbf{x} < \infty$. Classical KDE convergence theory [35, 42] deals with the case where the samples $\{X_m^{(i)}\}_{1 \le i \le N}$ are iid, with common pdf $p_m(\mathbf{x})$. The typical figure of merit is the mean integrated square error (MISE),

$$\mathrm{MISE}(\hat{p}_m^N) = \mathbb{E}\left[\int_{\mathcal{X}} \left(p_m(\mathbf{x}) - \hat{p}_m^N(\mathbf{x})\right)^2 \mathsf{d}\mathbf{x}\right].$$

Using techniques based on the Taylor expansion of the error, it can be shown that $\lim_{N\to\infty} \mathrm{MISE}(\hat{p}_m^N) = 0$, but no convergence rates are available in general. The MISE can be optimised with respect to (w.r.t.) the kernel $\kappa(\mathbf{x})$. When the samples are iid and drawn from the actual $p_m(\mathbf{x})$, the minimum MISE is obtained using the Epanechnikov kernel [35]. Stronger results, including uniform convergence and the analysis of non-iid samples, can be found in [10].

2.2. The Fokker-Plank equation

The Fokker-Planck equation (FPE) [25] is a partial differential equation (PDE) that describes the dynamics of a family of pdf's $\{p(\mathbf{x}, t)\}_{t \ge 0}$ where, for a given t, $p(\mathbf{x}, t)$ is the pdf of the (random) state $\mathbf{X}(t)$ conditional on the initial condition $\mathbf{X}(0)$. Typically, this approach is discarded as computationally prohibitive for practical problems [36] because the cost of the implementation grows

exponentially with the state dimension (although some tractable schemes have been proposed recently for low-dimensional problems [36]). In addition, the FPE is derived under regularity assumptions that do not hold in general, e.g., when the drag is random (other than a Wiener process).

2.3. Gaussian UP methods

Prediction and Kalman filtering methods based on the unscented transform (UT) [24] have become popular since the late 90s. Given a Gaussian pdf, the UT provides an efficient procedure to propagate both the mean and the covariance matrix (which fully characterize the Gaussian pdf) through a nonlinear function using deterministic samples. The result is an approximation of the actual covariance accurate to the second order. Algorithms that rely on the UT can be computationally fast, however, they are only useful when the target pdf can be well approximated by a Gaussian (which is not the case in orbital UP, see, e.g., Fig. 2 in [27]). UT can be combined with Kalman-like update steps if additional observations are available. This combination yields a so-called unscented Kalman filter (UKF) [29].

Other UP methods based on deterministic sampling and integration can be designed using cubature theory [26, 12]. The term "cubature" refers to a class of methods that generalize classical quadrature schemes to multidimensional settings. The use of cubature methods for UQ and filtering was originally proposed in [28] and it gained attention in the engineering community after the publication of [5]. In the framework of UP, cubature schemes are very similar to techniques based on the UT. The key feature of cubature schemes is that they are designed to be exact for a certain class of nonlinear transformations, while the UT always yields approximations. In this work, two spherical-radial cubature rules have been utilized to design UP algorithms. The third-degree spherical-radial rule (SRC3D) uses the same number of reference points as the UT transform, and has similar accuracy (but it has been reported as numerically more stable), while the fifth-degree spherical-radial rule (SRC5D) uses a larger number of reference points and attains a superior accuracy. Details on the derivation of both methods can be found in [21].

A major drawback of UT- and cubature-based UP methods is that the covariance matrix of the state variables increases quickly and, as a consequence, such methods significantly overestimate the uncertainty of the predictions after a few days. This limitation is a consequence of the nonlinearity in the propagation of the state: after a moderate interval of time (ranging from a few hours to a few days, depending on the initial uncertainty) the pdf of the state becomes non-Gaussian and the mean vector and covariance matrix do not yield sufficient information to characterize it.

2.4. Polynomial approximations of the state vector

A family of UP methods rely on the construction of polynomial approximations of the state variables with the initial condition as an argument. In other words, given an initial condition $\mathbf{x}_0$, these schemes yield an approximation of the state at time t of the form $\mathbf{x}_t \simeq \psi_t(\mathbf{x}_0)$, where $\psi_t(\cdot)$ is a suitable polynomial.

In this category, one can find techniques based on the Taylor differential algebra and state transition tensors, consisting in the propagation of polynomials. They just differ in the way of obtaining such polynomials. The third method is based on the polynomial chaos expansion, which consists in an interpolation pf the solution state with orthogonal polynomials depending on the probability distribution of the initial state.

Taylor differential algebra An algebra $\mathcal{A}$ over a field is a vector space equipped with a bilinear product [17]. The derivative of the product of two functions $f(\cdot)$ and $g(\cdot)$ can be seen as a bilinear product. Then, if we consider the Taylor expansion of these functions, the coefficients form an algebra with the bilinear product. The key idea behind the use of Taylor differential algebra or differential algebra for short (DA), for computation purposes, is using this bilinear product to construct polynomial approximations (truncated to a prescribed order) of transformations which involve much heavier computations [7].

Consider, for example, an ordinary differential equation (ODE)

$$\mathtt{d}\mathbf{x}(t) = f(\mathbf{x}, t)\mathtt{d}\mathbf{x}, \quad \text{with} \quad \mathbf{x}(0) = \mathbf{x}_0, \tag{4}$$

where the solution can be written as a function of the initial value, namely, $\mathbf{x}(t) = \varphi(\mathbf{x}_0, t)$. When we use a numerical scheme to integrate the ODE above, we compute an approximation of this function, hence we denote $\mathbf{x}_t \simeq \widehat{\varphi}(\mathbf{x}_0, t)$. The DA methodology provides the means to construct a polynomial (Taylor) approximation around a given point $\mathbf{x}_0$ in the state space, of the form

$$\widehat{\mathbf{x}}(t) = \widehat{\varphi}(\mathbf{x}_0', t) = \widehat{\varphi}(\mathbf{x}_0, t) + \widehat{\nabla}\varphi(\mathbf{x}_0, t) \cdot (\mathbf{x}_0' - \mathbf{x}_0) + \dots, \tag{5}$$

for any initial condition $\mathbf{x}_0'$ which lies within the convergence region of $\mathbf{x}_0$. If the initial value is random, $\mathbf{x}_0' = \mathbf{x}_0 + \Delta\mathbf{x}_0$, where $\Delta\mathbf{x}_0$ is a random vector, the expansion (5) becomes

$$\widehat{\mathbf{x}}(t) = \widehat{\varphi}(\mathbf{x}_0 + \Delta\mathbf{x}_0, t) = \widehat{\varphi}(\mathbf{x}_0, t) + \widehat{\nabla}\varphi(\mathbf{x}_0, t) \cdot \Delta\mathbf{x}_0 + \dots. \tag{6}$$

Hence, we can compute an approximation for the moments of $\widehat{\mathbf{x}}(t)$ simply computing the moments of the right-hand side (rhs) of (6). Again, the accuracy of the estimates depends on whether $\mathbf{x}_0'$ lies in the convergence region of $\mathbf{x}_0$. As a consequence, the estimate $\widehat{\mathbf{x}}$ can be expected to be more accurate when the covariance of $\Delta\mathbf{x}_0$ is narrow enough. Otherwise, more sophisticated schemes based on the partition of the state space (domain-splitting methods [43]) are needed.

The DA approximation can yield accurate estimates of the solution $\mathbf{x}(t)$ simply evaluating a polynomial. Computing the coefficients of the polynomial approximation can be computationally demanding, though. The process involves running the numerical scheme that generates $\widehat{\varphi}(\mathbf{x}_0, t)$, where all computations are implemented (approximated) using the bilinear product of the algebra. Truncation of the resulting polynomials at each time step is also needed to keep the degree of the approximation at time t fixed. When t is very large or the polynomial degree needed to attain good accuracy is high, the cost of computing the necessary coefficients can be prohibitive. In any case, numerical implementations of this method, demand thorough optimization. Moreover, if stochastic dynamics are considered (as in Eq. (1)), the random terms that result from $\mathbf{G}(\mathbf{x}, t)\mathrm{d}\mathbf{W}$ should be considered variables in the approximating polynomials. In that case, the number of variables would increase at every time step and become prohibitive even for short time spans..

State transition tensors The state transition tensor method (STT) relies on obtaining the Taylor coefficients of the solution of the propagation in a similar way as the DA technique does. However, it differs in the way of computing them. Instead of "translating" the operations in the algorithm into the DA framework, the STT method computes the coefficients by solving a system of ordinary differential equations of first order in time [31]. In general, these equations can not be solved analytically and solutions must be approximated by numerical integration. For that reason, the computational cost can be expected to be greater than the cost of the DA approach.

Polynomial chaos expansion Polynomial chaos expansion (PCE) methods approximate the function $\varphi(\mathbf{x}_0, t)$ using a basis of orthogonal polynomials. The most common families of orthogonal polynomials are related with common probability measures [3], including the Legendre, Laguerre and Hermite polynomials. In general, the computation of the coefficients of the expansion can not be done analytically. Instead, it requires the numerical solution of an optimization problem or the evaluation of an integral (typically by MC integration).

The PCE method, as described in the literature, is only valid for the deterministic case, i.e., with $\mathbf{G}(\mathbf{x}, t) = \mathbf{0}$ in Eq. (4). Like in the DA framework, the random terms generated by $\mathbf{G}(\mathbf{x}, t)\mathrm{d}\mathbf{W}$ otherwise would yield extra variables in the interpolating polynomial. Compared to DA, the main advantage of PCE is that the resulting polynomial approximation is valid irrespective of the initial condition $\mathbf{x}_0' = \mathbf{x}_0 + \Delta\mathbf{x}_0$ even if the perturbation $\Delta\mathbf{x}_0$ is large. It is also easier to code and, in principle, computationally lighter.

2.5. Gaussian mixture models

Given a collection of pdf's, $p_1(\mathbf{x}), p_2(\mathbf{x}), \cdots, p_N(\mathbf{x})$, all of them defined over the same probability space, the *weighted* sum

$$p(\mathbf{x}) = \sum_{i=1}^{N} w_i p_i(\mathbf{x}) \tag{7}$$

is a finite mixture pdf if all the weights, $w_i, i = 1, \cdots, N$, are non-negative and $\sum_{i=1}^{N} w_i = 1$ with $w_i \geq 0$. It can be shown [20] that if $p(\mathbf{x})$ above represents the initial pdf of the state $\mathbf{x}$ at time t_0 in a dynamic system in state-space form, a prediction step from time t_0 to time $t_0 + T$ does not affect the weights in the mixture, but only the shape of the individual pdf's. In other words, UP amounts to propagating the uncertainty in each individual component. However, the latter still remains a hard task when the system is non-linear or affected by non-Gaussian noise.

Adaptive Gaussian mixtures Let $\mathbf{x}_0$ denote the state at time t_0 and assume that every component of the mixture, N in (7) is kept fixed or can be updated over time. An example of a fixed GM approximation can be found in [20]. Adaptive methods, which increase the number N of components over time to keep the mixture approximation accurate over time have gained attention in recent years and we have investigated two of them in detail, namely, the AEGIS (adaptive entropy-based Gaussian-mixture information synthesis) [14] and DoNG (direction of non-Gaussianity) [16] algorithms. AEGIS relies on an information-theoretic criterion (the differential entropy of the nonlinearly-transformed state vector) to decide whether a given transformation introduces "too much nonlinearity" when applied to each Gaussian component of the mixture. In that case, the original Gaussian component is split into three new Gaussian pdf's with narrower covariance matrices. Hence, the AEGIS algorithm works in two steps: detection of nonlinearity and, if appropriate, splitting of the corresponding Gaussian.

The DoNG [16] method follows a similar scheme, but it relies on a different criterion for splitting the Gaussian terms of the mixture. Specifically, it introduces a metric of *non-Gaussianity* that quantifies how much a random vector departs from its original Gaussian distribution when going through a nonlinear transformation. In the DoNG algorithm, the non-Gaussianity along each principal axis (eigenvectors of the covariance matrix) of the original random vector variable is computed, and the splitting direction is then given by the linear combination of the different axes weighted by their corresponding non-Gaussianity metric. The specific metric suggested in [16] is the skewness of the transformed random variable. Since the pdf of the latter cannot be computed analytically, the skewness (third order central moment) which we have computed using a 5th degree cubature scheme. Both DoNG and AEGIS use the "3-component splitting library" in [14] for splitting the components that have been diagnosed as non-Gaussian.

Kernel density estimators We may use the KDE scheme of Section 2.1 to construct a GM approximation with a fixed number of components in a straightforward

way. Let $p_t(\mathbf{x}_t)$ denote the pdf of the state at time t. At time $t = 0$, we construct a KDE of the initial distribution by independently sampling $\mathbf{m}_0^{(i)}$ from the initial pdf $p_0(\mathbf{x}_0)$, $i = 1, \ldots, N$. Then, we define

$$p_0^N(\mathbf{x}_0) := \frac{1}{N} \sum_{i=1}^{N} K_{h,0}^i(\mathbf{x}_0),$$

where the kernel $K_{h,0}^i(\mathbf{x}) = \mathcal{N}(\mathbf{x}|\mathbf{m}_0^{(i)}, h\mathbf{C}_0)$ is the Gaussian pdf with mean $\mathbf{m}_0^{(i)}$ and covariance matrix $h\mathbf{C}_0$, and $h > 0$ is a bandwidth parameter. Typically, $\mathbf{C}_0$ is the initial covariance matrix of $\mathbf{x}_0$, while h is a parameter to be tuned. The pdf of $\mathbf{x}_t$ is then approximated by propagating forward in time each kernel $K_{h,0}^{(i)}$. For this purpose, we have used a 3rd-degree cubature method, but other techniques can be suitable as well.

This KDE UP scheme is simple and flexible enough to represent non-Gaussian distributions. Its main drawback is that the approximation of the tails of the distribution of the target pdf at time t, $p_t(\mathbf{x}_t)$, is often poor as t increases. Also, the performance depends considerably on the choice of the bandwidth h, which is not easy to adjust in general. We recommend the simple rule $h \propto N^{-\frac{1}{2(2d+1)}}$, where d is the dimension of $\mathbf{x}_t$, obtained in [10]. The proportionality constant for h can typically be chosen in the interval $\left(\frac{1}{4}, 1\right)$.

3. SPACE DEBRIS SERVICES

The provision of meaningful space-debris-related services is based on furnishing relevant information to the stakeholders in concise terms, easy to interpret and consistent with the current and historic operation. The Draft Recommendations for Space Data System Standards of the Consultative Committee for Space Data Systems (CCSDS) is a reference for defining the quantities of interest. In particular, the Re-entry Data Message (Red Book) [8] and Conjunction Data Message (Blue Book) [34] are of use, because each space-debris service has its own set of relevant parameters, and they can be obtained from the stochastic description of the state vectors. Therefore, collision avoidance and re-entry deserve a separate analysis which is gathered hereafter.

3.1. Reentry prediction

The Red Book [8] recommends the use of the RDM KVN data format to define a probable atmospheric reentry. The keywords listed in Table 3-3 of that document can be directly derived from the probability distribution of the reentry time, hereafter denoted T_r, which is characterized as a real and positive random variable. We discuss three methods to approximate the pdf of T_r subject to the uncertainty in the state of the decaying object.

Standard Monte Carlo In the standard MC method, N iid sample trajectories (also termed *particles*) are generated, with the form

$$\mathbf{x}_t^{(i)} = \begin{bmatrix} \mathbf{r}_t^{(i)} \\ \mathbf{v}_t^{(i)} \end{bmatrix}, \quad i = 1, ..., N,$$

where $\mathbf{r}_t^{(i)}$ is the position vector and $\mathbf{v}_t^{(i)}$ is the velocity vector. Assuming a Cartesian coordinate framework with origin at the centre of the Earth, the altitude over the Earth surface for the i-th particle is $h_t^{(i)} := \|\mathbf{r}_t^{(i)}\| - R_E$, where R_E is the Earth radius.

Let γ be a threshold altitude. A reentry occurs for the i-th particle at time t_0 when $h_{t_0}^{(i)} < \gamma$. After a reentry event, the particle is discarded (not propagated anymore), the re-entry time is recorded as $T_r^{(i)} = t_0$ and we keep propagating the remaining sample trajectories. The simulation finishes when all particles have crossed the threshold γ. At that moment, we have a collection of re-entry times $\{T_r^{(i)}\}_{i=1}^N$ that we can use to: 1) generate a histogram, an approximate pdf or an approximate cdf of the reentry time; 2) estimate the mean and mode of the reentry time; 3) estimate the variance, standard deviation or other dispersion parameters of the reentry time; 4) compute a time window (an interval where the reentry time is contained with some prescribed high probability); 5) compute probabilities of reentry for specific time intervals; etc.

In particular, $\mu(\mathrm{d}s) = \frac{1}{N} \sum_{i=1}^{N} \delta_{T_r^{(i)}}(\mathrm{d}s)$, where $\delta_T(\mathrm{d}s)$ denotes the Dirac delta measure located at T, is a random probability measure that approximates the actual probability law of the reentry time. Moreover, at any given time t, the surviving particles yield an empirical approximation of the probability distribution of the state of the object conditional on $T_r > t$, i.e., conditional on the object not to have decayed yet.

Gaussian methods: UT and cubature Gaussian UP methods can yield a Gaussian approximation for the probability distribution of $\mathbf{x}_t$ at any time t of interest, i.e., we endow $\mathbf{x}_t$ with a normal pdf, $p(\mathbf{x}_t) \approx \mathcal{N}(\mathbf{x}_t|\mathbf{m}_t, \mathbf{C}_t)$ where $\mathbf{m}_t$ is a mean vector and $\mathbf{C}_t$ is a covariance matrix. Detecting a re-entry from a Gaussian distribution is not straightforward. Starting from $\mathcal{N}(\mathbf{x}_t|\mathbf{m}_t, \mathbf{C}_t)$, one may use the UT or a cubature method to approximate the mean ($\bar{h}_t$) and variance (σ_h^2) of the altitude over the surface, $h_t = \|\mathbf{r}_t\| - R_E$, which is itself random. Then, for a threshold altitude γ, the Chebyshev inequality [13]

$$\mathbb{P}(|h_t - \bar{h}_t| > |\gamma - \bar{h}_t|) < \frac{\sigma_h^2}{|\gamma - \bar{h}_t|^2} \tag{8}$$

provides an upper bound for the probability of a reentry at time t. However, this bound is often loose and does not provide a good estimate of the reentry time. An alternative is to approximate the probability distribution of the altitude h_t as a Gamma distribution with mean $\bar{h}_t$ and variance $\sigma_{h,t}^2$, denoted $\mathrm{Ga}(h_t|\bar{h}_t, \sigma_{h,t}^2)$ and then approxi-

mate the probability of reentry at time t as

$$\mathbb{P}(h_t < \gamma) \simeq \int_0^{\gamma} \mathrm{Ga}(h_t|\bar{h}_t, \sigma^2_{h,t})\mathrm{d}h_t, \qquad (9)$$

which can be computed efficiently with most software packages.

However, these approaches are not practical because they accumulate the error in the approximation of $\mathbf{C}_t$ with the error in the approximation of $\sigma^2_{h,t}$. As a consequence, the estimate of $\sigma^2_{h,t}$ degenerates after a few days. This phenomenon is illustrated in Fig. 1 (left), which shows the evolution of the mean and standard deviation of the altitude of a decaying satellite (GOCE) as computed from $\mathcal{N}(\mathbf{x}_t|\mathbf{m}_t, \mathbf{C}_t)$. After a few days, the Earth's surface is within a single standard deviation of the mean altitude.

The estimation of the altitude can be significantly improved if we compute a Gaussian approximation directly on the space of h_t. Let $\mathbf{x}_0$ be distributed as $\mathcal{N}(\mathbf{x}_0|\mathbf{m}_0, \mathbf{C}_0)$. We can compute σ-points $\mathbf{x}^i_0$ and weights w^i, $i = 1, ..., 2d+1$ (where $d = 6$ is the space dimension), and propagate them to time t using the dynamical equation, denoted as $\mathbf{x}^i_t = \phi_t(\mathbf{x}^i_0)$, for $i = 1, ..., 2d+1$. We can project the time t σ-points to the space of altitudes as $h^i_t = \|\mathbf{r}^i_t\| - R_E$, where $\mathbf{r}^i_t$ is the position corresponding to the σ-point $\mathbf{x}^i_t$. Then the expected value and the standard deviation of the altitude can be approximately computed as

$$\bar{h}_t = \sum_{i=1}^{2d+1} w^i h^i_t \quad \text{and} \quad \sigma_{t,h} = \sqrt{\sum_{i=1}^{2d+1} (h^i_t - \bar{h}_t)^2},$$

respectively. Both $\bar{h}_t$ and σ_t can be efficiently computed and remain numerically stable for many days in a stable orbital trajectory, which allows to compute approximate probabilities of the form of Eqs. (8) or (9) more accurately.

However, when the object is decaying the standard deviation σ_t increases sharply and prevents any useful computation of re-entry probabilities. This is illustrated in Figure 1. The plot in the middle shows the mean altitude of GOCE (black) $\pm 2\sigma_{t,h}$ (red-coloured). The estimate of the altitude remains tight while $\bar{h}_t$ remains large. However, as the altitude decreases, the standard deviation $\sigma_{t,h}$ increases sharply (right-hand plot). Even if h_t falls well below a generous threshold of $\gamma = 150$ km, since $\sigma_{h,t}$ becomes large, $\mathbb{P}(h_t < \gamma)$ does not approach 1.

Actually, the UP algorithm breaks down as the object decays. This is because the sigma points spread out quickly and some of them land into regions of the state space where the dynamical equation stops yielding meaningful results (e.g., positions close to the Earth's surface). The sharp increase of σ_t is just a consequence of this more general issue.

Hybrid Monte Carlo At time t, the KDE method described in Section 2.5 yields an approximation of the pdf of the state vector, denoted $p(\mathbf{x}_t)$, of the form

$$p(\mathbf{x}_t) \approx \frac{1}{n} \sum_{i=1}^{n} \mathcal{N}(\mathbf{x}_t|\mathbf{m}^{(i)}_t, \mathbf{C}^{(i)}_t). \qquad (10)$$

It has been tested that the KDE remains stable for a longer time than a Gaussian approximation. This suggests a combination of the KDE and MC methods into the following hybrid scheme:

1. Run a KDE method for UP. At time instants t_k, $k = 1, 2, 3, ...$ (e.g., every hour or every two hours) use the approximation (10) and the Chebyshev inequality to obtain an upper bound for the probability of reentry. Denote this bound as $U(t_k)$.
2. Let $\pi_* << 1$ be a small positive threshold (e.g., 5×10^{-4}). If $U(t_k) < \pi_*$ then go back to step 1 and keep running the KDE approximation.
3. Otherwise, if $U(t_k) \geq \pi_*$ then draw N iid samples $\mathbf{x}^{(i)}_t$, $i = 1, ..., N$, from the KDE in (10) and use them to run a standard MC procedure to estimate the law of the reentry time T_r.

Similar hybrid methods can be implemented using UT or cubature schemes for the UP stage instead of the KDE. However, these algorithms have been found useful mainly for short-term reentry predictions (less than 3 days, as a rule of thumb). While Gaussian UP is computationally fast, one can expect the transition to the MC phase to occur earlier than with KDE.

3.2. Collision avoidance

The key metric for assessing conjunction is the time of closest approach (TCA) and the miss distance (MD), both of them obligatory information in the Conjunction Data Message [34]. These quantities correspond to the averaged motion of spacecraft and debris. Additionally, a metric related to collision risk (or collision probability) can also be provided [2]. This latter value is usually compared against an accepted collision probability level (ACPL) to decide whether a collision avoidance manoeuvre should be carried out. Nevertheless, the way in which the statistical information should be used by satellite operators is a subject of current controversy [6]. The way in which collision probability is defined depends on the nature of the encounter between spacecraft and debris. In case of a "short term" encounter, in which the relative velocity between the objects is high enough, the conjunction is usually considered instantaneous and the collision risk is defined as an instantaneous metric [9]. On the other hand, in case of a "long term" encounter, target and chaser are close to each other during a non-negligible length of time. Thus, relative dynamics and uncertainty evolution cannot be ignored [9]. The former corresponds to the typical situation in LEO and permits great simplifications in the calculations. The latter is the usual case in GEO orbit, or in formation flying configurations.

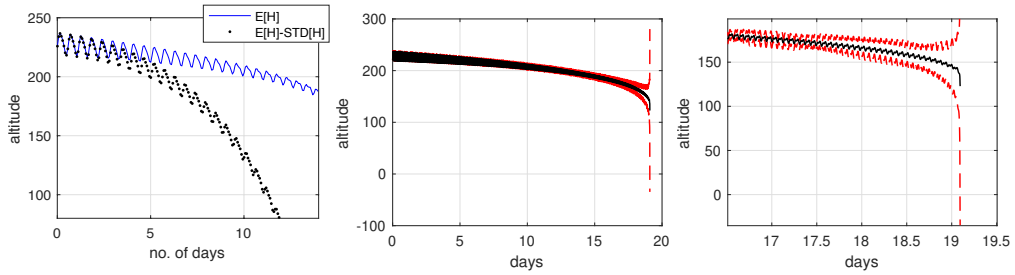

Figure 1. ***Left:*** *Evolution of the mean altitude $\bar{h}_t$ (blue) of GOCE and the mean altitude minus one standard deviation $\bar{h}_t - \sigma_{h,t}$ (black) over time when $\mathcal{N}(\mathbf{x}_t|\mathbf{m}_t, \mathbf{C}_t)$ is computed via a spherical-radial cubature rule of 5th degree.* ***Middle:*** *Evolution of $\bar{h}_t$ (black) of GOCE and $\bar{h}_t \pm \sigma_{h,t}$ (red) over time when using a direct UT scheme (results with cubature are similar).* ***Right:*** *Zoom into the last three days of propagation.* ***All:*** *The starting day is October 21, 2013 @ 00:00:00. Altitude is in km, with 0 indicating the Earth's surface.*

Monte Carlo analysis MC has been used in previous works as the benchmark to establish the reference probability of collision, taken as truth [23]. MC involves, mainly, UP without performing any geometrical computation or transformation other than checking that the distances between particles simulating each object are higher than a certain threshold. In [22], a detailed method on how to proceed to compute the collision probability using an MC approach is described. For completeness, the details of the approach implemented in this work are gathered below.

Let us assume that the initial conditions for the state of the two potentially colliding objects is given by Gaussian pdf's,

$$\mathbf{x}_0^A \sim \mathcal{N}(\boldsymbol{\mu}_0^A, \mathbf{C}_0^A)\,, \qquad \mathbf{x}_0^B \sim \mathcal{N}(\boldsymbol{\mu}_0^B, \mathbf{C}_0^B)\,,$$

where superscripts A and B stand for the first and second object, respectively, $\boldsymbol{\mu}$ is the mean and $\mathbf{C}$ is the covariance. The two objects are subject to the same dynamics. Let us generate a set of N independent pairs of trajectories,

$$\mathbf{z}_{0:T}^i = \left\{ \begin{bmatrix} \mathbf{x}_0^{A,i} \\ \mathbf{x}_0^{B,i} \end{bmatrix}, \begin{bmatrix} \mathbf{x}_1^{A,i} \\ \mathbf{x}_1^{B,i} \end{bmatrix}, ..., \begin{bmatrix} \mathbf{x}_T^{A,i} \\ \mathbf{x}_T^{B,i} \end{bmatrix} \right\}, \quad i = 1, ..., N.$$

The statistics of interest in this experiment are: the time of collision and probability of collision (assuming that a collision occurs when the distance between the objects is less than a threshold η), the TCA and the MD. They can be defined using the function

$$T_\eta^i := \begin{cases} \min \left\{ 0 \le t \le T : d(\mathbf{x}_t^{A,i}, \mathbf{x}_t^{B,i}) < \eta \right\}, \text{or} \\ \infty, \quad \text{if} \quad d(\mathbf{x}_t^{A,i}, \mathbf{x}_t^{B,i}) > \eta \; \forall t \end{cases}$$

where $d(\mathbf{x}_t^{A,i}, \mathbf{x}_t^{B,i})$ is the Euclidean distance between the objects A and B at time t. The time of collision is a random variable denoted T_η. Its law can be approximated by the discrete probability distribution

$$\mathbb{P}^N(T_\eta \le K) = \frac{1}{N} \sum_{i=1}^{N} u[K - T_\eta],$$

where $u[k]$ is the unit-step function ($u[k] = 1$ for $k \ge 0$ or $u[k] = 0$ otherwise). Assume that we intend to analyze the collision probability in an interval $[0, T]$. The function $P_{c,\eta}^N(t) = \mathbb{P}^N(T_\eta \le t)$ is an estimator of $P_{c,\eta}(t) = \mathbb{P}^N(T_\eta \le t)$, the probability of a collision ocurring up to time $t \le T$. The final collision probability $P_{c,\eta} = \mathbb{P}(T_\eta \le T)$ is similarly estimated as $P_{c,\eta}^N := P_{c,\eta}^N(T)$.

Let T_A denote the TCA. For the i-th MC trajectory we obtain the estimate

$$T_A^i = \begin{cases} T_\eta^i, \quad \text{if} \quad T_\eta^i < \infty, \text{ or} \\ \arg \min\limits_{0 \le t \le T} d(\mathbf{x}_t^{A,i}, \mathbf{x}_t^{B,i}), \text{ if } T_\eta^i = \infty. \end{cases}$$

The probability of the TCA occurring up to time t can be estimated from the empirical distribution $\mathbb{P}^N(T_A < t) = \frac{1}{N} \sum\limits_{i=1}^{N} u[t - T_A^i]$, and the resulting MC estimator of T_A is $T_A^N = \frac{1}{N} \sum\limits_{i=1}^{N} T_A^i$. Finally, the MD estimate for the i-th trajectory is $D_\eta^i = d(\mathbf{x}_{T_A^i}^{A,i}, \mathbf{x}_{T_A^i}^{B,i}) - \eta$. The empirical probability distribution is $\mathbb{P}^N(D_\eta < d_0) = \frac{1}{N} \sum\limits_{i=1}^{N} u[d_0 - D_\eta^i]$, and the MC estimator of the MD is $D_\eta^N = \frac{1}{N} \sum\limits_{i=1}^{N} D_\eta^i$.

We remark that the treatment of different encounters for the same propagation time span should be coupled. Samples that "collide" in the first encounter are not included in the population for the following encounters. In an operational context, however, the analysis can be performed for individual encounters under certain simplifying assumptions.

UT and KDE methods The theoretical basis and use of the UT and KDE methods for UP have been explained in Section 2. In the context of conjunction analysis, both methods are used for obtaining an estimate of the TCA and the MD (as a computationally less-demanding alternative to the MC estimators T_A^N and D_η^N).

Let us define the TCA and MD in terms of the Gaussian pdf's generated by the UT scheme. In particular, we assume that

$$\mathbf{x}_t^A \sim \mathcal{N}(\boldsymbol{\mu}_t^A, \mathbf{C}_t^A)\,, \qquad \mathbf{x}_t^B \sim \mathcal{N}(\boldsymbol{\mu}_t^B, \mathbf{C}_t^B)$$

and the time of collision, for a threshold distance η, is

$$T_\eta^{\mathsf{UT}} := \begin{cases} \min\left\{0 \le t \le T : d(\boldsymbol{\mu}_t^A, \boldsymbol{\mu}_t^B) < \eta\right\}, & \text{or} \\ \infty, \quad \text{if} \quad d(\boldsymbol{\mu}_t^A, \boldsymbol{\mu}_t^B) > \eta \quad \text{for all } t. \end{cases}$$

Accordingly, the time of closest approach is defined as

$$T_A^{\mathsf{UT}} = \begin{cases} T_\eta^{\mathsf{UT}} \quad \text{if} \quad T_\eta^{\mathsf{UT}} < \infty, \quad \text{or} \\ \arg \min\limits_{0 \le t \le T} d(\boldsymbol{\mu}_t^A, \boldsymbol{\mu}_t^B) \end{cases}$$

and the corresponding MD is $D_\eta^{\mathsf{UT}} = d(\boldsymbol{\mu}^A_{T_A^{\mathsf{UT}}}, \boldsymbol{\mu}^B_{T_A^{\mathsf{UT}}}) - \eta$. A similar method can be applied to obtain estimates of the TCA and MD from the KDEs of the two objects, instead of the Gaussian densities generated by the UT method.

Once T_A^{UT} and D_η^{UT} (as well as $\boldsymbol{\mu}^A_{T_A^{\mathsf{UT}}}$, $\boldsymbol{\mu}^B_{T_A^{\mathsf{UT}}}$, $\mathbf{C}^A_{T_A^{\mathsf{UT}}}$ and $\mathbf{C}^B_{T_A^{\mathsf{UT}}}$) are identified, the collision probability is computed using: a) the B-plane method, where the mean state and covariance matrix are projected and the collision probability is computed as in a deterministic case; or b) Patera's integral method. For the latter, an additional threshold distance is defined in order to establish the starting and finishing instants of time for the computation of the collision probability. For the former, two approaches are considered: 1a) the covariance matrix is obtained from the σ-points before projecting it onto the B-plane and 1b) the σ-points are projected onto the B-plane and the planar covariance matrix is computed from the projections.

4. SOFTWARE PROTOTYPE

A software prototype has been developed including the most promising algorithms for UP and the different space debris services. The propagation libraries implemented are the ones from NAPEOS, where new modules have been added to implement the stochastic propagation methods. The prototype is structured in four different chains or workflows:

- Uncertainty Propagation of an object based on its initial state and covariance
- Re-entry prediction of an object based on its initial state and covariance and the altitude threshold γ.
- Collision risk computation of two objects based on their initial states and covariances
- Atmospheric uncertainty computations based on space weather data

The propagation of the state can be performed both integrating an SDE or an ODE. Thus, the stochastic effects can be analyzed and results compared with the deterministic case. The following methods are available using both SDE and ODE integration for each of the three main chains:

- Uncertainty propagation: MC, UT, spherical-radial cubature of 3rd degree (SRC3D), spherical-radial cubature of the 5th degree (SRC5D), KDE, linear propagation (for ODE only) and AEGIS (for ODE only).
- Re-entry prediction: MC and hybrid KDE-MC.
- Collision risk computation: MC and UT.

The atmosphere uncertainty has also been researched within this project. It can been characterized using different models (Gaussian distribution, Gamma distribution, constant coefficient of variation, and time and space correlated distributions). All these models (except the constant coefficient of variation) are based on the atmospheric density standard deviation as a function of altitude and solar activity. This input can be computed prior to the propagation by executing the atmospheric uncertainty chain of the software using space weather recorded data.

Most of the execution-dependent information is passed to the software through clearly defined file-based interfaces. Furthermore, the software has been designed to comply with standard interfaces (CCSDS OPM, OEM and OMM or CCSDS CDM and RDM) whenever possible. The configuration of each execution can be modified manually on a configuration file or, alternatively, it can be done via a HMI.

Apart from the propagation of an initial state and covariance using different algorithms, the software is designed to provide post-processing outputs of the variables of interest depending on the selected execution chain, being the most relevant ones: 1) UP (final state and covariance, covariance ellipsoid and pdf of all propagated variables), 2) re-entry prediction (time of re-entry pdf and re-entry window, re-entry ground-track, re-entry latitude/longitude pdf), and 3) collision risk (TCA and MD with their corresponding variances, B-plane uncertainty and the same outputs as in the uncertainty propagation chain for both objects).

5. RESULTS

5.1. Re-entry prediction

The scenarios that have been chosen for re-entry analysis are the decay of GOCE, because of the availability of an extended data set,and the decay of Iridium 60, using

TLE data, for the application of particle filtering in the estimation of the drag term.

The hybrid technique that combines a stage of KDE-based UP with an MC simulation stage for the counting of reentries has been applied to the study of the decay of GOCE. The KDE scheme uses $M = 50$ random samples, $\mathbf{m}_0^{(i)}$, $i = 1, ..., 50$, at time $t = 0$, to approximate the initial pdf as a mixture of Gaussians. The kernel used for the approximation is Gaussian, with the i-th component centered at $\mathbf{m}_0^{(i)}$ and covariance $\mathbf{C}_0^{(i)} = h_i \mathbf{C}_0$, where the sample-specific bandwidth factor α_i is computed following one of the schemes suggested in [38], namely, $h_i = 10 \times M^{-\frac{1}{d}} p_0(\mathbf{m}_0^{(i)})^{-\frac{1}{d}}$, with $d = 6$, the dimension of the Cartesian frame, and $p_0(\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_0|\mathbf{m}_0, \mathbf{C}_0)$ the Gaussian distribution of the state at time $t = 0$. The resulting KDE at time $t = 0$ is $\hat{p}_0^M(\mathbf{x}_0) = \frac{1}{M} \sum_{i=1}^M \mathcal{N}(\mathbf{x}_0|\mathbf{m}_0^{(i)}, \mathbf{C}_0^{(i)})$.

The pdf $\hat{p}_0^M(\mathbf{x}_0)$ is propagated over time using the SCR3D rule to update the Gaussian components. Every 3 hours, the algorithm uses the Chebyshev inequality to obtain an upper bound for the probability of reentry, denoted $U(t')$ if computed at time t'. If $U(t') > \pi_* = 8 \times 10^{-4}$, then the algorithm generates $N = 4,000$ iid samples from the KDE $\hat{p}_{t'}^M(\mathbf{x}_{t'})$ and uses standard MC to predict the reentry.

We have applied this method to track the decay of GOCE with three setups:

- Deterministic dynamics and deterministic atmospheric density (NRLMSISE00 model).
- Stochastic dynamics and stochastic atmospheric density: the state dynamics are governed by the SDE (1), where $\mathbf{G}(\mathbf{X}, t)\mathrm{d}\mathbf{W} = \sigma_w \mathrm{d}\mathbf{W}$ and the constant scale parameter is set as $\sigma_w = 10^{-8}$, which accounts for the order of magnitude of the truncation in the differential equation. The atmospheric density is modelled as a Gamma random variable with mean given by the NRLMSISE00 model and standard deviation depending on the altitude and solar activity.

We declare a reentry when the altitude of the satellite is predicted to fall below a threshold $\gamma = 150$ km. Note that crossing this threshold does not imply an immediate decay.

Figure 2 shows the histograms and cumulative distribution functions (cdf's) of the reentry time T_r for a single simulation with deterministic dynamics and deterministic atmospheric density (the two panels on the left) and the same graphs, histograms and cdf's, when the dynamics and the atmospheric density are stochastic (two panels on the right). The approximate distributions of the T_r obtained with the hybrid KDE-MC method have a slightly broader support than the reference MC, implying that the uncertainty is (slightly) overestimated. Similar results are obtained for the deterministic and stochastic cases, although a larger uncertainty (longer tails) can be observed in the latter scenario.

A summary of performance metrics for the experiments in Figure 2 is provided in Table 1. In general, the estimated distribution of the reentry time using the hybrid method has longer tails compared to standard MC. This affects the calculation of the reentry window. We suggest that windows containing > 0.99 probability of reentry should be adopted, as shown in Table 1.

	mean(T_r)	Std(T_r)	Reentry window
Hybrid KDE (deterministic model)	18.795	0.351	(17.636; 20.238)
Monte Carlo (deterministic model)	18.832	0.175	(18.026; 19.636)
Hybrid KDE (stochastic model)	18.861	0.681	(17.187; 21.992)
Monte Carlo (stochastic model)	18.832	0.215	(17.841; 19.884)

Table 1. Performance metrics for the hybrid KDE-MC method versus the $N = 10^6$ MC reference method. T_r is the reentry time in days. For the MC method, the reentry window contains the whole support of T_r. For the hybrid KDE-MC method, T_r lies within the re-entry window with probability > 0.99.

Figure 3 shows the normalized run-times for the hybrid KDE-MC method and the MC reference simulation. Note the difference of three orders of magnitude.

Particle filtering for drag estimation The ballistic coefficient and the atmospheric density determine the drag and they are, therefore, key to the accuracy of reentry predictions. Hereafter, we illustrate the application of a particle filtering algorithm with 2 time scales (PF2S) [11] to estimate the probability distribution of the ballistic coefficient (B) using TLE data.

For this experiment, we consider the problem of predicting the reentry of Iridium 60. We assume an altitude threshold $\gamma = 120$ km and work with a sequence of 12 TLEs that start on March 9, 2019, at 14:15:51 and finish on March 12, 2019, at 22:23:27, to estimate the probability distribution of the ballistic coefficient B. The PF2S propagates the state samples (particles) using a 4th-order Runge-Kutta (RK4) integrator of the underlying SDE, while the ballistic coefficient is assumed to follow a random walk driven by Gaussian noise with variance $\sigma^2 = 4 \times 10^{-8}$, i.e.,

$$B_k = B_{k-1} + u_k, \tag{11}$$

where u_k is a sequence of iid Gaussian random variables with common distribution $\mathcal{N}(u_k|0, \sigma^2)$. The time scale of k in Eq. (11) is assumed much larger than the step size of the RK4 integrator. In practice, for the simulation we generate initial particles $B_0^{(i)}$, $i = 1, ..., N$, at time 0 from a uniform distribution on the interval $[-0.05, 1]$ and then we generate a new sample B_k from B_{k-1} at the time of the k-th available TLE (i.e., we sample new coefficients

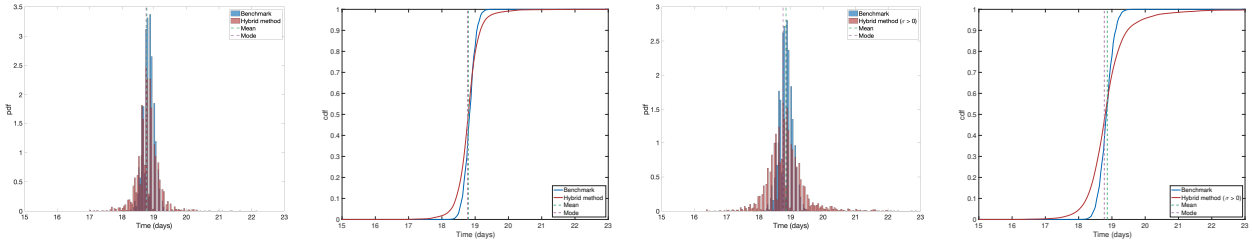

Figure 2. Re-entry times distribution for the hybrid KDE-MC method and the GOCE satellite scenario. Comparison with an $N = 10^6$-sample MC scheme for reference. ***Left:*** *histograms and cumulative distribution functions (cdf's) of the re-entry time T_r with the two methods when the dynamical equation is deterministic, including the atmospheric density (nominal NRLMSISE00 values). The vertical lines indicate the mean (green) and the mode (brown) for the hybrid KDE-MC algorithm.* ***Right:*** *histograms and cdf's for the two methods when the dynamical equation is an SDE and the atmospheric density is random (Gamma distribution centred around the NRLMSISE00 output, and altitude-dependent variance). We observe a larger spread of the distributions due to the additional uncertainty.*

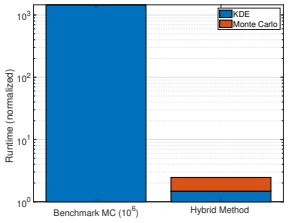

Figure 3. Comparison of normalized run-times for the hybrid KDE-MC method and the $N = 10^6$-sample MC used as reference. The two colors in the bar on the right indicate the run-times associated to the KDE stage (bottom) and MC stage (top) of the hybrid algorithm. The first stage typically propagates the initial distribution over 15-16 days.

right after processing each observation). Each sample $B_k^{(i)}$ is linked to a sequence of state particles $\mathbf{x}_t^{(i)}$, i.e., we propagate the Cartesian coordinates $\mathbf{x}_t^{(i)}$, $i \in \{1, ..., N\}$, using, precisely, the ballistic coefficient $B_k^{(i)}$. Each time a TLE is available, we compute new weights for the particles $\{\mathbf{x}_t^{(i)}, B_k^{(i)}\}_{i=1}^N$. We extract the altitude from the state information the k-th TLE, say h_k, and assume a Gaussian conditional pdf,

$$p(h_k|\mathbf{x}_t^{(i)}) = \mathcal{N}(h_k|\|\mathbf{r}_t^{(i)}\| - R_E, \sigma_h^2),$$

where $\sigma_h^2 = 4$. This pdf enables the computation of the particle weights. After updating the weights with the TLE altitude, we resample to reset the weights, propagate the coefficient particles $B_k^{(i)}$ using Eq. (11), and regenerate the state particles by generating a new set samples with mean the position indicated by the TLE and the initial covariance matrix.

Twelve TLEs are processed in this way, the last of them from March 12, 2019, at 22:23:27. After that, each state particle $\mathbf{x}_t^{(i)}$ $(i = 1, ..., N)$ is propagated forward over time with its own coefficient $B^{(i)}$ that is kept fixed for the rest of the simulation. Re-entry events, with a threshold $\gamma = 120$ km are counted to obtain an empirical distribution of T_r in the usual way.

The first plot in Figure 4 displays the histogram, and estimated posterior pdf, of the ballistic coefficient B obtained after processing the 12 TLEs, between March 9 at 14:15:51 and finish on March 12 at 22:23:27, using the PF2S algorithm. The dashed vertical line indicates the conventional estimate of $B = 0.0081$ computed from the $B_\star$ data in TLE of March 12 at 22:23:27. The probability distribution generated by the PF2S (with $N = 2,000$ particles) given the available data leaves this value on the left tail. The posterior mean value is $\sum_{i=1}^N B^{(i)} = 0.0102$ and its standard deviation is $\text{std}^N(B) = 8.19 \times 10^{-4}$.

The second plot in Figure 4 shows the empirical probability distribution of altitudes predicted by the PF2S algorithm with $N = 2,000$ particles for March 12 at 22:23:27, right before processing a TLE. The vertical line indicates the actual altitude computed from the TLE at that time, which is close to the mean of the predictive distribution. The third plot in the same Figure 4 shows the predicted distribution of altitudes computed using standard MC (with $N = 2,000$ samples as well) starting from the information (state and ballistic coefficient) in the immediately previous TLE (March 12 at 19:19:46). We see that the histogram is offset and the actual altitude lies on the left tail.

Finally, the third plot in Figure 4 displays the histogram and estimated pdf for the time of reentry T_r of Iridium 60. The vertical dashed line indicates the time of the last available TLE (March 18, 2019, at 02:21:40) which provides an altitude of 123.98 km (our threshold is 120 km). The mean re-entry time is $\bar{T}_r = 9.511$ days (after March 9 at 14:15), the standard deviation is $\sigma(T_r) = 0.482$ days and the reentry window is $(8.439; 11.009)$ (with probability > 0.99). Standard MC using $N = 2,000$ samples and initial information from the TLE on March 12 at 22:23:27 yields the reentry window $(10.320; 11.448)$ with probability > 0.99, which is clearly offset to the right.

Short-term prediction, 24-hour time frame In order to explore the limits of the hybrid approaches, we have

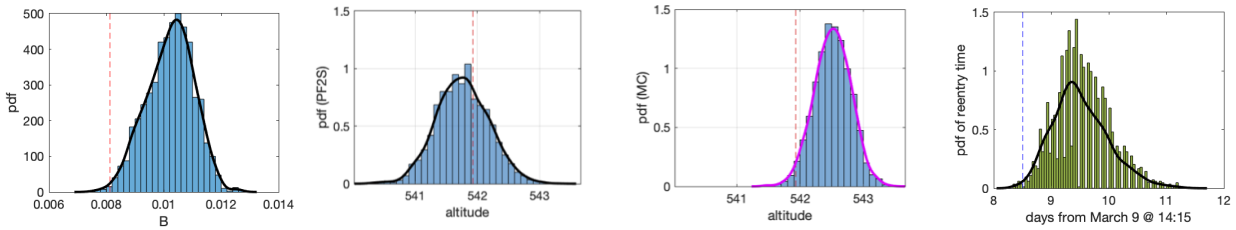

Figure 4. ***First plot:*** *posterior pdf for the ballistic coefficient B computed by the PF2S algorithm with $N = 2,000$ particles using 12 consecutive TLEs over 2 days. The vertical red line indicates the ballistic coefficient that would be conventionally estimated from the last processed TLE.* ***Second plot:*** *predictive pdf for the altitude of Iridium 60 on March 12 at 22:23:27 computed by the PF2S algorithm with $N = 2,000$ particles. The vertical red line indicates the actual altitude as extracted from the TLE corresponding to that exact time.* ***Third plot:*** *predictive pdf for the altitude of Iridium 60 on March 12 at 22:23:27 computed by a standard MC method that starts running at the immediately previous TLE (March 12 at 19:19:46).* ***Fourth plot:*** *Histogram of re-entry times for Iridium 60 obtained using the PF2S algorithm with $N = 2,000$ particles.*

repeated the computer experiment allowing a larger number of TLEs for the PF2S algorithm to estimate the ballistic coefficient B and then run prediction algorithms for T_r over a short time frame of a few hours. To be specific, we have used a sequence of 38 TLEs for Iridium 60, starting on March 9, 2019, at 14:15 h and extending up to March 17, 2020, at 21:35. The PF2S scheme with $N = 2,000$ particles yields an estimate of the ballistic coefficient $\hat{B}^N = 0.0098$, which is then used to predict the re-entry time T_r using MC, the hybrid KDE-MC scheme and a hybrid SRC3D-MC method. The number of components and the number of samples per component in the MC stage of the hybrid KDE-MC algorithm are $M = 40$ and $N_{kde} = 50$, respectively. The number of samples in the MC stage of the hybrid SRC3D-MC method is $N_{src} = 2,000$. Therefore, the three algorithms use the same number of independent trajectories for estimating T_r, namely $N = MN_{kde} = N_{src} = 2,000$.

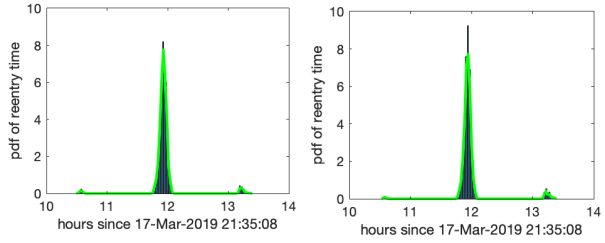

Figure 5. ***Left:*** *histogram and estimate of the pdf of the re-entry time T_r for Iridium 60 using the MC method.* ***Right:*** *Histogram and estimate of the pdf of the re-entry time T_r for Iridium 60 using the hybrid SRC3D-MC method. In both cases, the prediction is carried out from March 15, 2019, at 12:42. The ballistic coefficient is estimated using a PF2S algorithm with 2,000 particles that uses 38 TLEs as data (first one on March 9, 2019, at 14:15; last one on March 17, 2019, at 21:35 h). The altitude threshold to declare a re-entry is 120 kms over the Earth's surface.*

Figure 5 shows the results (histogram and estimated pdf of T_r) attained by the MC (left plot) and hybrid SRC3D-MC (right plot) schemes, respectively. The units in the horizontal axes are hours since March 17, 2020, at 21:35 (the time of the last TLE processed by the particle filter). We observe that the two schemes yield similar results, with a clear mode after 12 hours and a small variance. Both methods also identify smaller modes at around 11 hours and 13 hours. The specific means and standard deviations output by the two algorithms (plus the hybrid KDE-MC method) are gathered in Table 2.

Method	$\mathbb{E}[T_r]$ (h)	std(T_r) (h)
Monte Carlo (MC)	11.95	0.27
Hybrid KDE-MC	11.94	0.26
Hybrid SRC3D-MC	11.98	0.27

Table 2. Performance of MC and hybrid methods in short-term re-entry prediction

5.2. Collision avoidance

Six scenarios have been selected using information from actual CDMs. Two of them correspond to short term encounters with different geometry and the other four to long-term encounters. The geometries of the two short-term encounters correspond to head-on and lateral encounters. For each of the six test cases, an MC scheme is run and used as benchmark for comparison with other methods. All the relevant statistical information can be obtained from the MC analysis. Additionally, the collision probability is computed using UT and KDE for uncertainty propagation jointly with B-plane-based methods and/or the integral Patera's method. One low-velocity case and one high-velocity encounter are discussed in more detail below.

Low-velocity encounter A close encounter between two geostationay objects, Artemis and Insat 2B, is analysed. The initial date is August 26th, 2017, at 16:35:01 UTC. The reference TCA, provided by the CDM is 5.4024 days ahead. The ballistic coefficient is zero for

	Minimum	Mean	Mode	Std
MD (m)	20.4	3095.5	600	2287.9
TCA (s)	14.95	12.33	11.6	15.23

Table 3. Case 1. MC benchmark, $N = 100,000$. TCA with respect to the nominal time provided by CDM.

both objects, because no atmospheric drag acceleration is considered. The initial covariance matrices for both objects have maximum values in the order of 10^6 m for position and 10^{-2} m/s for velocity.

Monte Carlo benchmark An MC algorithm has been run starting from the initial states and covariances and using $N = 10^5$ samples. Results for the distributions of MD and TCA are shown in Figure 6. Mean, mode and standard deviation of the distributions can be seen in Table 3. In addition, the TCA for the minimum achieved distance is displayed in the first column of Table 3. Accordingly, the collision probability for objects with a combined radius of less than 20 m is less than 10^{-5}. Assessing smaller collision probabilities leads to an increasing number of samples. The TCA is close to the one predicted by the CDM, with a distance of less than one standard deviation from the mean and the mode. In turn, the MD distribution has a standard deviation larger than 2 km with a non-normal behaviour (expected due to the distance definition).

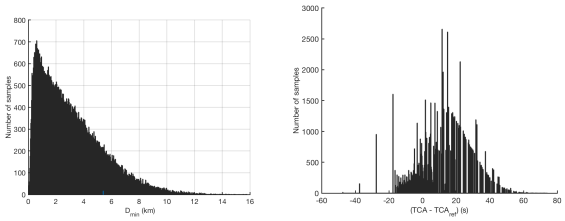

Figure 6. Case 1. Distribution of MD and TCA.

UT and KDE propagation The first step in the process is the identification of the TCA (and corresponding MD). In Figure 7, the evolution of the MD between the objects in time is shown. Additionally, the process noise related to the truncation error (σ_w) has been used as a parameter to assess the influence of its variation in the predicted TCA. The shape of the evolution of the distance is similar in all cases, with a small dispersion in TCA but relevant differences in the computed MD. The KDE, in turn, predicts larger MDs for the same noise level (σ_w) with a small dispersion in TCA.

Tables 4 and 5 show numerical values for UT and KDE propagation of MD, TCA and corresponding collision probabilities. As it can be seen, for UT, the TCA varies between -10 and 10 seconds, whereas the MD changes in a factor of 20. Collision probabilities are small, as it corresponds to the MD obtained, and for all cases the order of magnitude is 10^{-8}. KDE values of the MD are in the

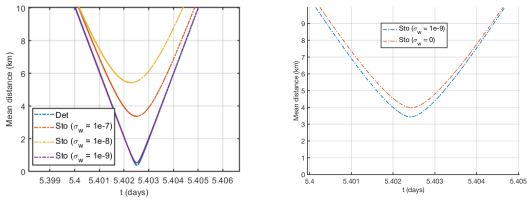

Figure 7. Case 1. UT and KDE propagation

order of km, with a deviation of TCA of 2.5 seconds with respect the value provided by the CDM.

	Det.	Stoch. ($\sigma_w = 0$)	Stoch. ($\sigma_w = 10^{-9}$)
MD (m)	387.6	275.3	535.6
TCA (s)	10.36	11.01	9.43
p_c	$7.83 \cdot 10^{-8}$	$7.44 \cdot 10^{-8}$	$7.28 \cdot 10^{-8}$
p_c (Patera's)	$9.91 \cdot 10^{-8}$	$7.65 \cdot 10^{-8}$	$4.21 \cdot 10^{-8}$
	Det. ($\sigma_w = 10^{-8}$)	Stoch. ($\sigma_w = 10^{-7}$)	Stoch.
MD (m)	5441.0	3381.6	
TCA (s)	-10.15	9.86	
p_c	$2.13 \cdot 10^{-8}$	$1.06 \cdot 10^{-8}$	
p_c (Patera's)	$2.94 \cdot 10^{-8}$	$1.10 \cdot 10^{-8}$	

Table 4. Case 1. UT propagation. Reference TCA: 5.4024 days

	Stoch. ($\sigma_w = 0$)	Stoch. ($\sigma_w = 10^{-9}$)
MD (m)	3982.0	3429.8
TCA (s)	2.51	2.50

Table 5. Case 1. KDE propagation. Reference TCA: 5.4024 days.

High-velocity encounter The close encounter between two LEO satellites, Rapid Eye 1 and CZ-CB, is examined. The initial date is October 6th, 2018, at 23:08:21 UTC. The reference TCA, provided by the CDM is 5.3132 days ahead. The ballistic coefficient is given by the CDM, it is the same for both objects, namely $B = 0.01374057$. The initial covariance matrices for both objects have maximum values in the order of 10^2 m for position and 10^{-4} m/s for velocity.

Monte Carlo benchmark An MC scheme has been run starting from the initial states and covariances and using $N = 10^5$ samples. Results for the distributions of MD and TCA are shown in Fig. 8. Mean, mode and standard deviation of the distributions can be seen in Table 6. In addition, the TCA for the minimum achieved distance is displayed in the first column of Table 6. The number of MC samples that reach a distance less than the chosen threshold (8 m) is 8, so the collision probability is $8 \cdot 10^{-5}$. The TCA presents a discrete behaviour before the reference TCA, in accordance to the graph of the previous paragraph. The mode corresponds with the predicted TCA from CDM, whereas the standard deviation is about 1 minute. In turn, the MD distribution has

	Minimum	Mean	Mode	Std
MD (m)	2.2	5877.1	519	4438.0
TCA (s)	32.10	43.70	-0.85	67.07

Table 6. Case 3. Benchmark Monte Carlo. $N = 100000$. TCA with respect to the nominal time provided by CDM

a standard deviation of about 4.5 km with a notably non-normal behaviour. The mode of the distribution is 520 m, although a second mode is closer, 10 m.

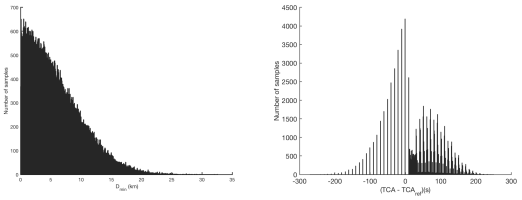

Figure 8. Case 3. Distribution of MD and TCA.

UT and KDE propagation The first step in the process is the identification of the TCA (and corresponding MD). In Figure 9, the evolution in time of the mean distance between the objects is shown. Additionally, the process noise related to the truncation error (σ_w) has been used as a parameter to assess the influence of its variation in the predicted TCA. Predictions with $\sigma_w < 10^{-8}$ are similar, with $\sigma_w < 10^{-8}$ the MD grows in a factor of about 4.

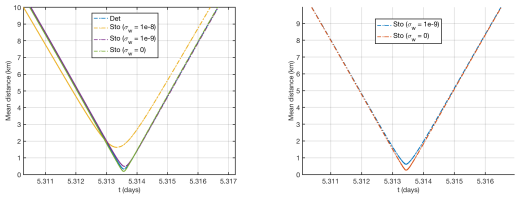

Figure 9. Case 3. UT and KDE propagation

Tables 7 and 8 show numerical values for UT and KDE propagation of MD, TCA and corresponding collision probabilities. As it can be seen, for UT, σ_w has a relevant role, mainly in the MD. The shift in TCA is less relevant (although small changes in TCA lead to important variations of MD). The three cases: $\sigma_w = 0, \sigma_w = 10^{-9}$ and deterministic UP give a similar location in time and distance of the closest approach. Nevertheless, the estimated collision probability for the stochastic cases are closer to the MC estimate, whereas the deterministic probability of collision is significantly smaller. The reason for that lies in the smaller uncertainty when no process noise is considered However, none of the methods is able to identify the close approach that appears in the MC to lead to a collision probability of 10^{-4}. KDE values of the MD are in the same range as their UT counterparts, and the deviation of TCA is 20 seconds with respect the value provided by the CDM, whereas the predicted TCA with UT is in the 31-33 seconds range.

	Det.	Stoch. ($\sigma_w = 0$)
MD (m)	323.9	190.4
TCA (s)	32.45	31.12
p_c	$2.10 \cdot 10^{-9}$	$3.52 \cdot 10^{-6}$
p_c (Patera's)	$6.85 \cdot 10^{-9}$	
	Stoch. ($\sigma_w = 10^{-9}$)	Stoch. ($\sigma_w = 10^{-8}$)
MD (m)	489.9	1627.5
TCA (s)	33.90	12.14
p_c	$7.25 \cdot 10^{-6}$	–

Table 7. Case 3. UT propagation. Reference TCA: 5.313202 days.

	Stoch. ($\sigma_w = 0$)	Stoch. ($\sigma_w = 10^{-9}$)
MD (m)	261.6	615.4
TCA (s)	20.22	20.33

Table 8. Case 3. KDE propagation. Reference TCA: 5.313202 days.

5.3. Prototype results

Uncertainty propagation To illustrate the operation of the UP chain of the prototype, we consider an object in a geosynchronous transfer orbit (GTO). Its initial state and covariance are derived from an orbit determination process. Figure 10 (left) displays the 3σ covariance containment of two of the implemented algorithms, namely the UT and the SRC5D rule, and how they compare to the traditional linear UP. This indicates the percentage of MC samples which are appropriately captured by the distribution provided by each of the algorithms by means of its aggregated state vector and covariance. For a propagation time of 3 days, the two considered algorithms still agree with the theoretical containment of a Gaussian distribution with 3 degrees of freedom.

Additionally, Fig. 11 shows 1D-marginal pdf's and 2D-scatter plots of the position state variables after 3 days of propagation for the same case described above. The plots on the diagonal present a good agreement of the 1D distribution of each of the coordinates, while the off-diagonal plots indicate both that the σ-points of the SRC5D algorithm populate well the domain explored by the MC benchmark, and that subsequent covariance reconstruction produces an adequate representation of the MC distribution. The performance of these methods is expected to degrade with longer propagation times as the resulting MC benchmark no longer features a Gaussian distribution.

Re-entry predictions The ground-track of one example of the re-entry prediction chain of the Prototype is shown in Figure 10 (middle). The red line is the averaged ground-track of the re-entry samples, green dots are the samples at re-entry epoch and the purple line shows the previous orbit of the object. All samples decay in between 8 and 9 days of propagation. There is a significant variance in the along-track direction, as expected due to

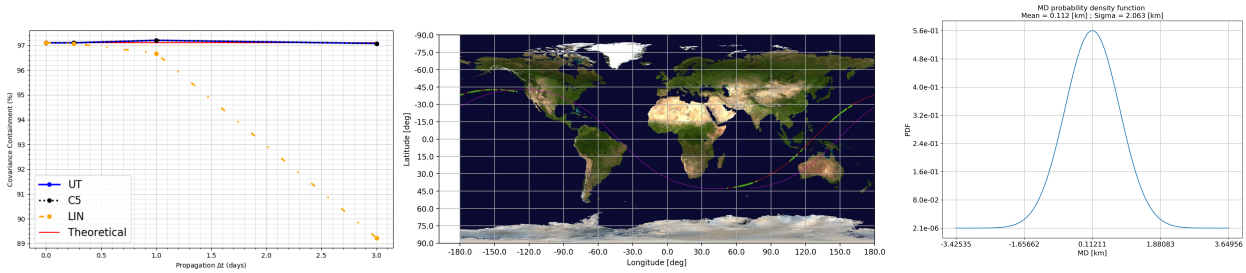

Figure 10. ***Left:*** *3σ covariance containment comparison between UT, SRC5D and linear UP methods. Object in a GTO, with deterministic dynamics and deterministic NRLMSISE00 atmosphere. Benchmark is a $N = 4 \cdot 10^3$-sample MC.* ***Middle:*** *Re-entry ground-track of an object using hybrid KDE algorithm.* ***Right:*** *MD pdf for a collision risk analysis with UT and stochastic dynamics (10 independent runs).*

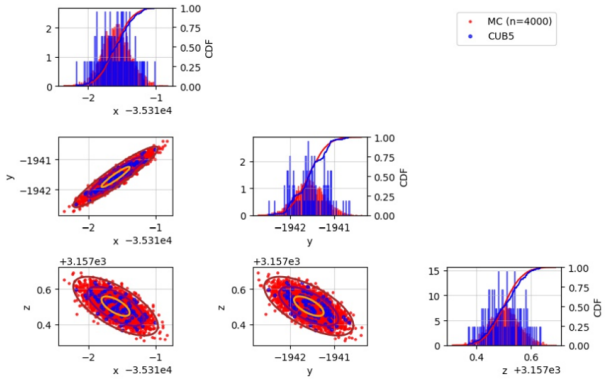

Figure 11. Position 1D-marginal pdf's and 2D-scatter plots of the SRC5D method and $N = 4 \times 10^3$-sample MC benchmark. Object in a GTO, with deterministic dynamics and deterministic NRLMSISE00 atmosphere. Includes $1,\ 2,\ 3\sigma$ Covariance ellipses.

the uncertainty in the drag force. One can also notice that re-entered samples seem to be grouped in different bands, due to the nature of the underlying hybrid algorithm. When a given kernel satisfies the criteria for re-entry detection, its aggregated state is sampled by means of an MC scheme producing a family of green points.

Collision risk Figure 10 (right) is an example of the post-processing collision risk analysis of the software for the MD. In this case, a UT method with stochastic dynamics was applied, having a time to TCA from the beginning of the propagation of 5 days approximately. A probability of collision of $8.13 \cdot 10^{-4}$ was obtained, computed using the Akella-Alfriend algorithm [1].

6. SUMMARY AND DISCUSSION

Uncertainty propagation We have compared the Monte Carlo (MC), Gaussian (UT and cubature), polynomial-based (DA, PCE), GM-based (AEGIS and DoNG) and KDE schemes for UP over three criteria: performance (accuracy of the predictions), computational cost of the algorithms and flexibility (whether the method can be adapted to stochastic models).

MC, DA and PCE methods can be very accurate. PCE and, especially, DA, can be computationally very costly, they require sophisticated and careful programming and are limited to deterministic models. In particular, they do not admit a random drag or a proper SDE in (1) with $\mathbf{G}(\mathbf{X}, t) \neq \mathbf{0}$. While MC is arguably the method with the higher computational cost, it can attain asymptotically optimal accuracy (and so we use it for benchmarking), it is easy to code, straightforwad to parallelize (which reduces runtimes drastically) and can be applied to all kinds of models, deterministic or stochastic.

Gaussian methods are relatively simple to code, they are computationally light and can be used reliably with deterministic and stochastic models. They are, however, the least accurate and errors can be very significant as the pdf of the state departs from the initial Gaussian distribution over time.

GM and KDE schemes are in an intermediate position in terms of performance: they can approximate non-Gaussian distributions (where UT and cubature methods fail) but they are outperformed by MC, DA and PCE schemes. AEGIS and DoNG are difficult to code and tune (they depend on sensitive parameters) and they are constrained to deterministic models, the same as DA and PCE. The KDE scheme is easier to code and can be used with deterministic or stochastic models, but suffers from significant errors in the estimation of the distribution tails and is sensitive to the choice of the bandwidth parameter.

Re-entry prediction UP propagation methods have been applied to the problem of long-term reentry prediction (8-30 days). The reentry of GOCE has been used as the main scenario for testing. Overall, it has been found that a robust and accurate estimate of the reentry time usually demands the (possibly partial use) of MC methods. To be specific, it has been found that UP methods combined with last-stage MC schemes for reentry prediction present a good performance on accuracy and reliability. The uncertainty accumulated over several days of UP typically makes the estimation of reentry times inaccurate when using Gaussian or GM approximations directly. DA and PCE could eventually overcome this difficulty but (i) their computational cost can be similar to MC schemes (which are easier to implement and more robust)

and (ii) DA/PCE methods are only available for deterministic models at this time, which implies that uncertainties in the drag and/or other acceleration terms have to be ignored.

Two schemes have been implemented for the prototype:

- *MC reentry prediction.* This is the most accurate and robust method, which fully exploits the dynamical and uncertainty models. The computational cost is high because it typically demands the simulation of several thousands of independent trajectories. However, the run times can be brought down to the scale of minutes by proper implementations on multicore computers and, especially, massively parallel GPUs.
- *Hybrid prediction.* Various hybrid schemes have been assessed and it has been found that the best performance-to-complexity trade-off is attained by using a KDE approximation of the target probability distribution (with a fixed number of components) for the initial UP stage. The KDE is easy to sample in order to switch to the MC stage. The reentry windows that we have found for the GOCE experiments are very similar to the windows obtained in the same conditions when running Monte Carlo trajectories from time t_0. Typical simulations of the KDE-MC method over 30 days, running Matlab on a single core of a 2016 MacBook Pro laptop computer (no parallelisation), take 60-70 minutes. Hybrid Gaussian-MC schemes can also be efficient for short-term re-entry predictions, as shown by experiments using TLE data for the decay of Iridium 60.

Collision risk We have tackled the analysis of conjunctions over a time horizon of about 7 days. The challenge for accurate estimation of low collision risks consists in being able to characterize the tail of the distribution of the minimum distance between the two objects. The accumulated error after the propagation time span on the pdf of the position of both objects, makes approximate methods not fully reliable. Gaussian approximations fail to reproduce the shape of the probability distribution of the positions, while KDE methods fail to estimate the tails accurately. Based on our experiments, two methods were selected for implementation in the prototype:

- *MC collision probability computation* is the most robust and accurate method. It has been used as benchmark. However, its computational cost is high, especially in scenarios where the risk to be assessed is relatively low.
- *UT and B-plane collision probability computation:* propagation methods based on Gaussian approximations combined with collision probability computation based on the B-plane provide a first and fast approximation to the collision risk. UT has been selected for collision probability computation because it balances speed and performance.

ACKNOWLEDGMENTS

This work has been supported by the European Space Agency, ESOC contract no. 4000126151/19/D/SR, "Uncertainty propagation meeting space debris needs (T711-501GR)".

REFERENCES

1. M. R. Akella and K. T. Alfriend. Probability of collision between space objects. *Journal of Guidance, Control, and Dynamics*, 23:769–772, 2000.
2. S. Alfano and D. Oltrogge. Probability of collision: Valuation, variability, visualization and validity. In *AIAA/AAS Astrodynamics Specialist Conference*, page 5654, 2016.
3. G. E. Andrews, R. Askey, and R. Roy. *Special functions*, volume 71. Cambridge university press, 2000.
4. I. Arasaratnam and S. Haykin. Cubature kalman filters. *IEEE Transactions on automatic control*, 54(6):1254–1269, 2009.
5. I. Arasaratnam and S. Haykin. Cubature kalman filters. *IEEE Transactions on Automatic Control*, 54(6):1254–1269, 2009.
6. M. S. Balch, R. Martin, and S. Ferson. Satellite conjunction analysis and the false confidence theorem. *arXiv preprint arXiv:1706.08565*, 2017.
7. M. Berz. Modern map methods in particle beam physics. *Adv. Imaging Electron Phys.*, 108:1–318, 1999.
8. R. BOOK. Re-entry data message. 2018.
9. J. R. Carpenter. Conservative analytical collision probability for design of orbital formations. 2004.
10. D. Crisan and J. Miguez. Particle-kernel estimation of the filter density in state-space models. *Bernoulli*, 20(4):1879–1929, 2014.
11. D. Crisan and J. Miguez. Nested particle filters for online parameter estimation in discrete-time state-space Markov models. *Bernoulli*, 24(4A):3039–3086, 2018.
12. D. Crisan and S. Ortiz-Latorre. A Kusuoka–Lyons–Victoir particle filter. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 469(2156):20130076, 2013.
13. M. H. DeGroot and M. J. Schervish. *Probability and statistics*. Pearson Education, 2012.
14. K. J. DeMars, R. H. Bishop, and M. K. Jah. Entropy-based approach for uncertainty propagation of nonlinear dynamical systems. *Journal of Guidance, Control, and Dynamics*, 36(4):1047–1057, 2013.
15. K. J. DeMars, Y. Cheng, and M. K. Jah. Collision probability with Gaussian mixture orbit uncertainty. *Journal of Guidance, Control, and Dynamics*, 37(3):979–985, 2014.

16. J. Duník, O. Straka, B. Noack, J. Steinbring, and U. D. Hanebeck. Directional splitting of Gaussian density in non-linear random variable transformation. *IET Signal Processing*, 12(9):1073–1081, 2018.

17. I. Etherington. Richard d. schafer, an introduction to nonassociative algebras (academic press inc., new york and london, 1966), x+ 166 pp., 64s. *Proceedings of the Edinburgh Mathematical Society*, 16(3):267–268, 1969.

18. K. Fujimoto and D. Scheeres. Tractable expressions for nonlinearly propagated uncertainties. *Journal of Guidance, Control, and Dynamics*, 38(6):1146–1151, 2015.

19. K. Fujimoto, D. J. Scheeres, and K. T. Alfriend. Analytical nonlinear propagation of uncertainty in the two-body problem. *Journal of Guidance, Control, and Dynamics*, 35(2):497–509, 2012.

20. J. T. Horwood, N. D. Aragon, and A. B. Poore. Gaussian sum filters for space surveillance: theory and simulations. *Journal of Guidance, Control, and Dynamics*, 34(6):1839–1851, 2011.

21. B. Jia, M. Xin, and Y. Cheng. High-degree cubature kalman filter. *Automatica*, 49(2):510–518, 2013.

22. B. A. Jones and A. Doostan. Satellite collision probability estimation using polynomial chaos expansions. *Advances in Space Research*, 52(11):1860–1875, 2013.

23. B. A. Jones, A. Doostan, and G. H. Born. Nonlinear propagation of orbit uncertainty using non-intrusive polynomial chaos. *Journal of Guidance, Control, and Dynamics*, 36(2):430–444, 2013.

24. S. J. Julier and J. K. Uhlmann. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3):401–422, March 2004.

25. M. Kumar, S. Chakravorty, and J. L. Junkins. A semianalytic meshless approach to the transient fokker–planck equation. *Probabilistic Engineering Mechanics*, 25(3):323–331, 2010.

26. C. Litterer, T. Lyons, et al. High order recombination and an application to cubature on wiener space. *The Annals of Applied Probability*, 22(4):1301–1327, 2012.

27. Y.-z. Luo and Z. Yang. A review of uncertainty propagation in orbital mechanics. *Progress in Aerospace Sciences*, 89:23–39, 2017.

28. T. Lyons and N. Victoir. Cubature on wiener space. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 460(2041):169–198, 2004.

29. H. M. Menegaz, J. Y. Ishihara, G. A. Borges, and A. N. Vargas. A systematization of the unscented Kalman filter theory. *IEEE Transactions on Automatic Control*, 60(10):2583–2598, 2015.

30. B. Øksendal. *Stochastic differential equations, 6th edition*. Springer, 2007.

31. R. S. Park and D. J. Scheeres. Nonlinear mapping of gaussian statistics: theory and applications to spacecraft trajectory design. *Journal of guidance, Control, and Dynamics*, 29(6):1367–1375, 2006.

32. A. B. Poore, J. M. Aristoff, J. T. Horwood, R. Armellin, W. T. Cerven, Y. Cheng, C. M. Cox, R. S. Erwin, J. H. Frisbee, M. D. Hejduk, et al. Covariance and uncertainty realism in space surveillance and tracking. Technical report, Numerica Corporation Fort Collins United States, 2016.

33. C. P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer, 2004.

34. C. Secretariat. Draft recommendation for space data system standards: Conjunction data message. *Draft recommended standard CCSDS*, 2012.

35. B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman & Hall/CRC, Boca Raton (FL, USA), 1986.

36. Y. Sun and M. Kumar. Uncertainty propagation in orbital mechanics via tensor decomposition. *Celestial Mechanics and Dynamical Astronomy*, 124(3):269–294, 2016.

37. G. Terejanu, P. Singla, T. Singh, and P. D. Scott. Uncertainty propagation for nonlinear dynamic systems using gaussian mixture models. *Journal of Guidance, Control, and Dynamics*, 31(6):1623–1633, 2008.

38. G. R. Terrell, D. W. Scott, et al. Variable kernel density estimation. *The Annals of Statistics*, 20(3):1236–1265, 1992.

39. M. Valli, R. Armellin, P. Di Lizia, and M. Lavagna. Nonlinear mapping of uncertainties in celestial mechanics. *Journal of Guidance, Control, and Dynamics*, 36(1):48–63, 2012.

40. V. Vittaldev and R. P. Russell. Collision probability for space objects using gaussian mixture models. In *Proceedings of the 23rd AAS/AIAA Space Flight Mechanics Meeting*, volume 148, pages 2339–2358, 2013.

41. V. Vittaldev, R. P. Russell, and R. Linares. Spacecraft uncertainty propagation using gaussian mixture models and polynomial chaos expansions. *Journal of Guidance, Control, and Dynamics*, pages 2615–2626, 2016.

42. M. P. Wand and M. C. Jones. *Kernel Smoothing*. Chapman-Hall/CRC, 1995.

43. A. Wittig, P. Di Lizia, R. Armellin, K. Makino, F. Bernelli-Zazzera, and M. Berz. Propagation of large uncertainty sets in orbital dynamics by automatic domain splitting. *Celestial Mechanics and Dynamical Astronomy*, 122(3):239–261, 2015.