Long-term collision probability computation through high-order Taylor polynomials evaluation

Andrea Zollo[(1)], Cristina Parigini[(2)], Roberto Armellin[(2)], Juan Félix San Juan Díaz[(3)], Saika Aida[(1)] and Ralph Kahle[(1)]

[(1)] *German Space Operation Centre (GSOC), DLR, Münchener Str. 20, 82234, Weßling, Germany, email: {andrea.zollo, saika.aida, ralph.kahle}@dlr.de*
[(2)] *Te Pūnaha Ātea - Space Institute, University of Auckland, Private Bag 92019, 1142, Auckland, New Zealand, email: {cristina.parigini, roberto.armellin}@auckland.ac.nz*
[(3)] *Scientific Computation & Technological Innovation Center, University of La Rioja, Madre de Dios 53, 26006, Logrono, La Rioja, Spain, email: juanfelix.sanjuan@unirioja.es*

***Abstract* – This paper introduces a versatile approach for computing the risk of collision specifically tailored for scenarios featuring low relative encounter velocities, but with potential applicability across a wide range of situations. The technique employs Differential Algebra (DA) to express the non-linear dynamical flow of the initial distribution in the primary-secondary objects relative motion through high-order Taylor polynomials. The entire initial uncertainty set is subdivided into subsets through Automatic Domain Splitting (ADS) techniques to control the accuracy of the Taylor expansions. The methodology samples the initial conditions of the relative state and evaluates the polynomial expansions for each sample while retaining their temporal dependency. The classical numerical integration of the initial statistics over the set of conditions for which a collision occurs is thus reduced to an evaluation of mono-dimensional time polynomials. Specifically, the samples reaching a relative distance below a critical value are identified together with the time at which this occurs. The approach is tested against an in-house Monte Carlo simulation for different literature test cases, showing accurate results and a consistent gain in computational time.**

I. INTRODUCTION

The escalating number of objects within the Earth's orbital environment presents a paramount hazard to spacecraft operations. The increasingly crowded orbital population leads to numerous close conjunctions over the course of a satellite's mission. Assessing the criticality of these events relies on evaluating the probability of collision (Pc) between two objects. The methods for the computation of collision risk have been tailored for different conjunction types. Close approaches between satellites are in fact typically classified as either short-term or long-term encounters.
Short-term encounters involve objects with significantly different orbits, resulting in encounter velocities reaching several kilometers per second near the point of closest approach. These encounters last only a few seconds at most. Throughout the encounter, the relative velocity vector remains constant in both intensity and direction, leading to a straight-line relative trajectory. Moreover, the relative velocity uncertainty is deemed negligible in comparison with its pronounced mean. Consequently, the position error combined ellipsoid remains stable throughout the encounter since the positional uncertainties of the objects can be defined by two uncorrelated constant covariance matrices. Different methods are available in the literature to compute the collision probability for the short-term case. The problem has been in fact extensively studied by Foster [1], Patera [2, 3], Alfriend et al. [4], Afano [5], Chan [6], and more recently by Serra et al. [7]. Typically, the risk of collision is computed by integrating the probability density function (PDF) of the combined positional uncertainty over the volume defined by the combined hard-body sphere [2]. Since the trajectory is rectilinear, the volume swept out can be visualized as an infinite cylinder along the direction of the relative velocity. Consequently, a 3D integral can be simplified to 2D, as the integral along the direction of the relative velocity equals unity.
The other type of encounter arises between two satellites traveling along neighboring orbits, such as between two GEO satellites at adjacent longitude positions or, more generally, during close satellite operations like rendezvous, formation, and cluster flights. It's worth noting that similar geometries can also occur naturally, albeit less frequently. These close approaches are characterized by a low relative velocity, typically on the order of a few meters per second. In such scenarios, the two objects remain in close proximity for an extended duration, approximately on the order of the orbital period. Unlike short-term encounters, the relative velocity vector is not constant but evolves over time, resulting in a non-linear relative trajectory. The uncertainty of the relative state cannot be assumed to be fixed, and the uncertainty associated with the velocity cannot be neglected. This leads to an evolving combined covariance matrix that depends on time.
The collision risk for long-term encounters has not been as thoroughly investigated as it has been for short-term encounters. A first category of methods, as in [8, 9] and [10], tries to solve the bending tendency of the collision tube. The key concept is to discretize the collision tube into small subsections and to consider that for each segment the assumptions of a short-term encounter still hold A better characterization of the collision volume is outlined in the works of Chan [11, 12], where the swept-

volume of the hard-body is described as an envelope of ellipsoids. However, it's important to note that these methods do not consider velocity uncertainties in the formulation of the problem. Reference [13], on the other hand, presents a comprehensive mathematical framework that, for the first time, incorporates velocity uncertainties into the formulation. This marks one of the most thorough efforts to fully address the collision probability problem in a general way. In this case, the intricate integration volume is continually mapped over time through the dynamic evolution of the initial conditions on the 3D surface of the hard-body sphere. However, the assumptions made do not accommodate for multiple encounters within the analysis timeframe, making the method not suitable for complex intersections of the integration volume.

As highlighted in [14], there is a notable absence in the literature of a general method capable of simultaneously characterizing the swept volume, especially when its shape is intricate, and computing the subsequent integral of the full state vector gaussian PDF over such a volume. In their work, it is suggested to approximate the swept volume using a Polynomial Superlevel Set (PSS) followed by a Monte Carlo integration to calculate the Pc. Similarly, we propose to characterize the entire dynamical flow of the initial conditions with Differential Algebra (DA) [15] and Automatic Domain Splitting (ADS) [16]. To elaborate further, the initial conditions' dynamical evolution is computed as a patched 7D continuum, where to each patch corresponds a high order Taylor expansion, function of time and of the initial conditions. Subsequently, the initial PDF is then sampled and each patch is evaluated in a specific state-vector realization to definitively determine the time evolution of a given sample. Instead of employing classical numerical integration for the collision volume, the process involves identifying the real roots of the DA approximation of the miss distance. The Pc is then computed as the frequency of threshold violations relative to the total number of samples.

II. PROBLEM DESCRIPTION

Building upon the derivation in [13], we can define the statistical event for which a collision occurs as follows: given the initial distribution of the state for two space resident objects at time t_0, the hard-body radius HBR, and a maximum period of interest T, a collision between two objects is deemed to occur if there exists a time t, within the interval $I = [t_0, t_0 + T]$, such that the norm of the relative distance vector $\boldsymbol{d}(t)$, is less than or equal to HBR. To assess the likelihood of this event occurring, and consequently characterize the Pc in a comprehensive and general manner, we introduce the relative state vector of the two objects engaged in the encounter, $\boldsymbol{x}(t)$, which is a function of the time. This vector is defined as

$$\boldsymbol{x}(t) = \begin{pmatrix} \boldsymbol{d}(t) \\ \boldsymbol{v_{rel}}(t) \end{pmatrix}, \tag{1}$$

in which $\boldsymbol{v_{rel}}$ is the relative velocity of the secondary object with respect to the primary. At time $t_0 = 0$, this is $\boldsymbol{x}(t_0) = \boldsymbol{x_0}$.

Defining the PDF of the relative state vector at time t_0 as $\boldsymbol{\rho_0}(\boldsymbol{x_0}, t_0)$, we can, without loss of generality, define the Pc as:

$$\text{Pc} = \Pr(\boldsymbol{x_0} \in \boldsymbol{V}) = \int_{\boldsymbol{V}} \boldsymbol{\rho_0}(\boldsymbol{x_0}, t_0)\, d\boldsymbol{x_0} \tag{2}$$

where $\boldsymbol{V} \subseteq \mathbb{R}^6$ represents the initial set for which a collision occurs at some future time t. For long-term encounters, the time evolution of the set $\boldsymbol{V}$ represents exactly the complex integration volume explored in [11, 12, 14]. At time t_0, the set $\boldsymbol{V}$ can be interpreted as sub-region of the multi-dimensional space $\mathbb{R}^6$ wherein every realization of the random vector $\boldsymbol{x_0}$ inevitably leads to the violation of the condition $\|\boldsymbol{d}(t)\| \leq HBR$ at a future time t. In mathematics:

$$\boldsymbol{V} = \{\boldsymbol{x_0} \in \mathbb{R}^6 : \exists\, t \in I : \|\boldsymbol{d}(t)\| - HBR \leq 0\} \tag{3}$$

By definition of PDF, the 6D integral of the initial statistic over the set $\boldsymbol{V}$ gives in fact the likelihood that $\boldsymbol{x_0} \in \boldsymbol{V}$. Since the uncertainty distribution of $\boldsymbol{x_0}$ usually depends on the outcome of the orbit determination problem, the integrand of (2) is most of the time assumed Gaussian:

$$\boldsymbol{\rho_0}(\boldsymbol{x_0}, t_0) = \frac{e^{\left(-\frac{1}{2}(\boldsymbol{x_0}-\boldsymbol{\mu_0})^T \boldsymbol{P_0}^{-1}(\boldsymbol{x_0}-\boldsymbol{\mu_0})\right)}}{\sqrt{(2\pi)^6}\sqrt{\|\boldsymbol{P_0}\|}} \tag{4}$$

where $\boldsymbol{\mu_0}$ and $\boldsymbol{P_0}$ represent respectively the mean and the covariance matrix of the relative state at t_0.One way of computing (2) is via a Monte Carlo-based method. In fact, in such cases, the initial conditions $\boldsymbol{x_0}$ are sampled and trajectories over the time interval $[t_0, t_0 + T]$ are computed according to some dynamical model that propagates the relative state from time t_0 to t. The dynamics are usually expressed as an Ordinary Differential Equation (ODE) of the form:

$$\begin{cases} \dot{\boldsymbol{x}}(t) = f(\boldsymbol{x}(t_0), \boldsymbol{x}(t), \boldsymbol{u}(t), t) \\ \boldsymbol{x}(t_0) = \boldsymbol{x_0} \end{cases}, \tag{5}$$

where the vector $\boldsymbol{u}(t)$ represents an eventually modelled manoeuvre in the relative dynamics. Each sample trajectory is analysed to verify if, at some future time, a collision occurs. If one hit is recorded for a specific sample, it means that it was originally belonging to the set $\boldsymbol{V}$. The probability is then computed evaluating the ratio between number of samples that produced a hit over the total number of samples.

III. METHODOLOGY

A. *DA to express the dynamical evolution of* $\boldsymbol{x_0}$

As outlined in the preceding section, our methodology employs DA techniques to express the dynamical evolution of the initial relative conditions. DA provides a computational framework enabling the treatment of functions as n-th order Taylor polynomial expansions within a computer environment, rather than handling them solely as floating-point values. This framework holds considerable potency as it allows to extract more information on a function rather than its mere raw values [15]. Within this context, the time t can be expressed as DA variable and scaled with respect to the maximum time of interest T, such that $\tau \in [-1,1]$:

$$\tau = \frac{2(t-t_0)}{T} - 1 \,. \tag{6}$$

Consistently, we introduce a vector of DA variables, $\delta \boldsymbol{x_0}$, that corresponds to the variation of the initial relative state vector from its mean at t_0. Each component is normalized by the maximum expected variation, $\Delta \boldsymbol{x_0}$, such that it is defined within the interval $[-1,1]$. This is

$$\delta \boldsymbol{x_0} = \frac{\boldsymbol{x_0} - \boldsymbol{\mu_0}}{\Delta \boldsymbol{x_0}} . \tag{7}$$

Within the context of this work the integration of (5) is performed employing Clohessy-Wiltshire relative dynamics [17] and thus under the assumption of almost-circular orbit for the primary object. The relative state at t, can be expressed in the DA framework as:

$$\boldsymbol{x} = \mathcal{T}_{\boldsymbol{x}}(\tau, \delta \boldsymbol{x_0}) \,. \tag{8}$$

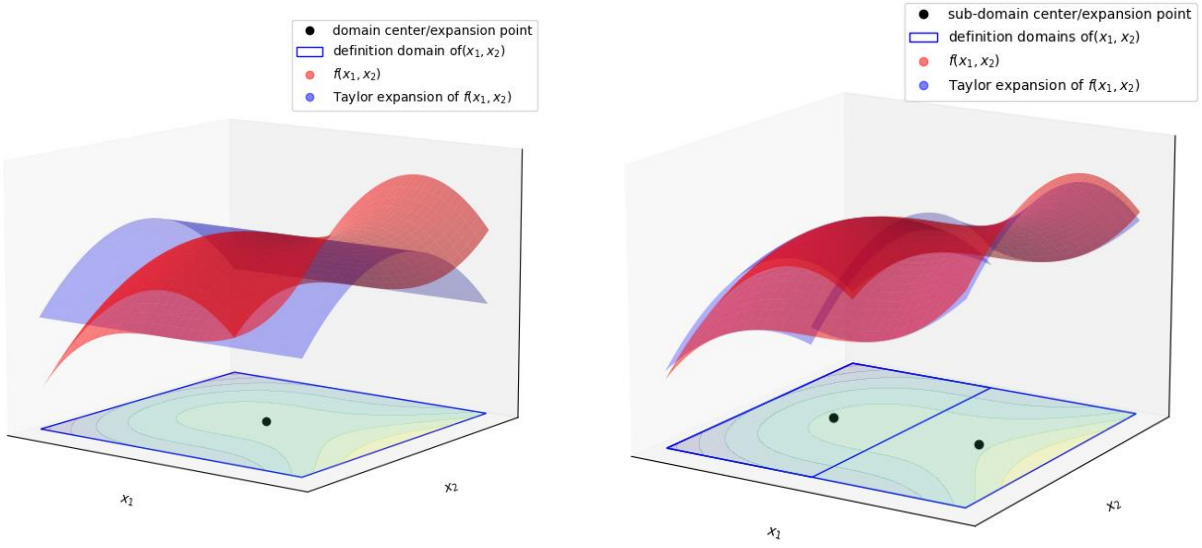
That is a vector of high order polynomials that are functions of the deviations of the scaled time τ and the initial normalized relative statistics, $\delta \boldsymbol{x_0}$. The Taylor map $\mathcal{T}_{\boldsymbol{x}}$ establishes a relationship between the perturbed initial state vector and the corresponding state vector at a specified time within T, utilizing the dynamical model defined in (5). This mapping from the initial set to the final one bears conceptual similarity to the mathematical notion of a manifold.

Utilizing DA in this context offers several advantages. Firstly, it enables the representation of an infinite set solely through the coefficients of the Taylor expansion. This preserves a specific analytical structure in contrast to a mere point-wise sets representation. Secondly, and perhaps most significantly, it allows the propagation of entire sets through a function using straightforward DA arithmetic operations. Unlike a Monte Carlo simulation, where the ODE flow of equation (5) must be integrated for each sample, in this scenario, only a single integration is required. The resulting DA expansion represents the outcome of propagating all points from the initial domain through the ODE in (5).

B. *ADS to control integration accuracy*

The challenging part of this approach arises from the highly non-linear dynamics involved usually in long-term encounters. When the Taylor map needs to approximate a strongly non-linear function, the convergence of the ODE expansion across the domain becomes cumbersome. Consequently, the DA map, which is a local representation of the function, poorly represents the actual evolution of the whole domain, even though the description is accurate in the vicinity of the centre of the expansion.

To address this issue, we employ the technique implemented in [16]. The ADS algorithm identifies instances where the ODE flow expansion over the initial conditions no longer accurately describes the dynamics.



(a) **(b)**

Fig. 1– ADS algorithm illustration. (a) Taylor expansion of $f(x_1, x_2)$ around initial domain's midpoint. (b) Taylor expansions of $f(x_1, x_2)$ recalculated around the new domain centers.

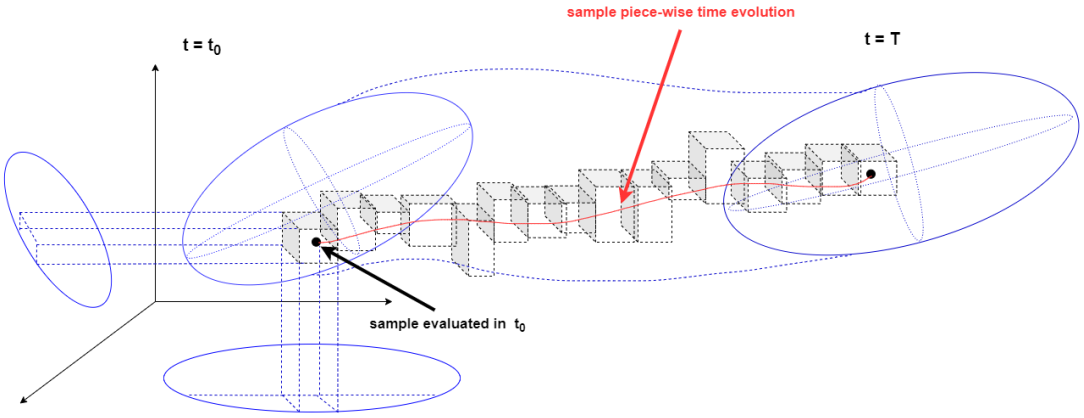


Fig. 2- Dynamical flow evolution of the initial conditions x_0

Once such a scenario is detected, the domain of the original polynomial expansion is divided along one of the expansion variables into two domains, each half the size of the original. By re-expanding the polynomials around the new centre points, two separate polynomial expansions are generated. Since the re-expansion of the polynomials preserves their order, each new polynomial remains identical to the original on its respective domain [16]. This process is visually depicted in Fig. 1 (a) and (b). In the visualization, a 2D function $f(x_1,x_2)$ is depicted alongside the defined domain of the variables (x_1,x_2). Additionally, Taylor expansion centered around the domain's midpoint is displayed to approximate $f(x_1,x_2)$. The accuracy of the approximation is high near the center but diminishes towards the domain edges. To ensure accuracy, the algorithm iteratively splits the initial domain into two segments whenever the Taylor series representation diverges from the actual function by a user-defined margin ε. Subsequently, the expansions are recalculated around the new centers, and this process continues until all expansions accurately represent the function within the specified threshold ε.

In a similar fashion, in our case the initial 7D domain, defined by the variables τ and $\delta\boldsymbol{x_0}$ is split in different sub-domains. The dynamical evolution of the initial condition assumes the shape of a patched 7D continuum, mathematically defined as a manifold object. Fig 2 attempts to give a visual representation of this last considering only the position components of the relative state vector and the time. To each patch at a given time corresponds a Taylor expansion, function of τ and of $\delta\boldsymbol{x_0}$, that approximates locally the dynamical flow. Once a single integration has been performed and the evolution of the initial condition is approximated by patched polynomials, our methodology proceeds to calculate the function f_d within the DA framework:

$$f_d(\tau,\delta\boldsymbol{x_0}) = \boldsymbol{d}^2(\tau,\delta\boldsymbol{x_0}) - HBR^2, \qquad (9)$$

f_d is a high order polynomial representation of the relative distance expressed as function of time and the variations $\delta\boldsymbol{x_0}$. It is again piece-wise defined, and its definition interval depends on the ADS accuracy control algorithm.

Subsequently, the initial set $\boldsymbol{x_0}$ is sampled via Cholesky decomposition of the covariance matrix $\boldsymbol{P_0}$. Each realization $\boldsymbol{X_{0i}}$ of the random vector is linked to its

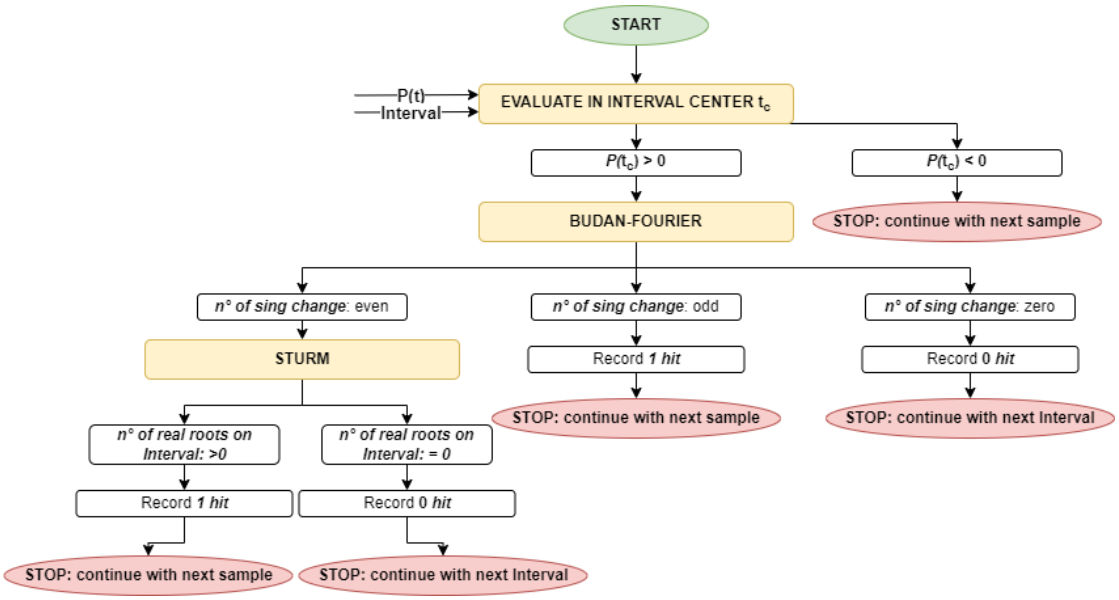


Fig. 3- Roots-finding algorithm to individualise the zeros of high order mono-dimensional time polynomials

Algorithm 1 DA polynomial evaluation technique

DA variable initialisation.
Perform ODE integration with ADS control. ▷ get eq. (9)
for each sample X_{0i} i=1:N **do**
  Evaluate $f_d(\tau, \delta x_0 = X_{0i})$ within the split initial set. ▷ get eq. (10)
  Get subset of polynomials that approximate sample trajectory $P(t)_i$
  for $P(t)_i$, I in sample trajectory **do**
    procedure ROOT FINDING ALGORITHM($P(t)_i, I$) ▷ Fig. 3
      if n°real roots > 0 **then**
        record 1 Hit
      end if
    end procedure
  end for
end for
Compute P_c as: $\frac{nHits}{nsamples}$

(a)

Algorithm 2 Monte Carlo

Initialize Δt and time grid.
for each sample i=1:N **do**
  Propagate sample $\forall t$
  Compute the relative distance $||d(t)||$
  procedure PARABOLIC BLENDING INTERPOLATION($t, ||d(t)||$)
    if Interpolated $||d(t)|| < HBR$ **then**
      record 1 Hit
    end if
  end procedure
end for
Compute P_c as: $\frac{nHits}{nsamples}$

(b)

Fig. 4 –High-level algorithm description for (a) DA polynomial evaluation technique and (b) standard Monte Carlo tool to compute Pc

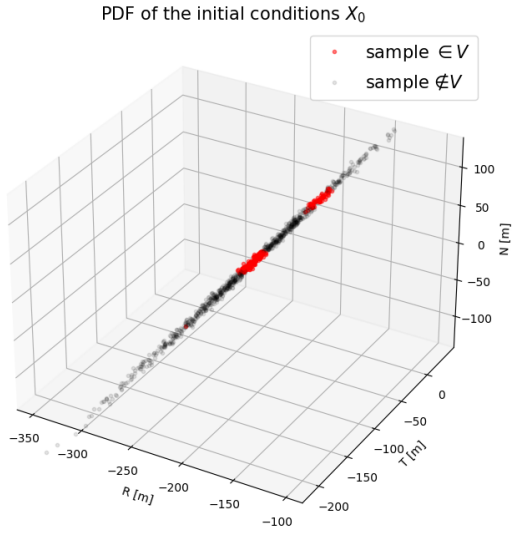
respective initial patch and evaluated only in space and velocity. This evaluation reduces the dimensions of the f_d polynomials, resulting in a set of one-dimensional Taylor expansions dependent solely on time:

$$f_d(\tau, \delta \boldsymbol{x_0} = \boldsymbol{X_{0i}}) = \mathcal{T}_{\boldsymbol{d}}(\tau), \qquad (10)$$

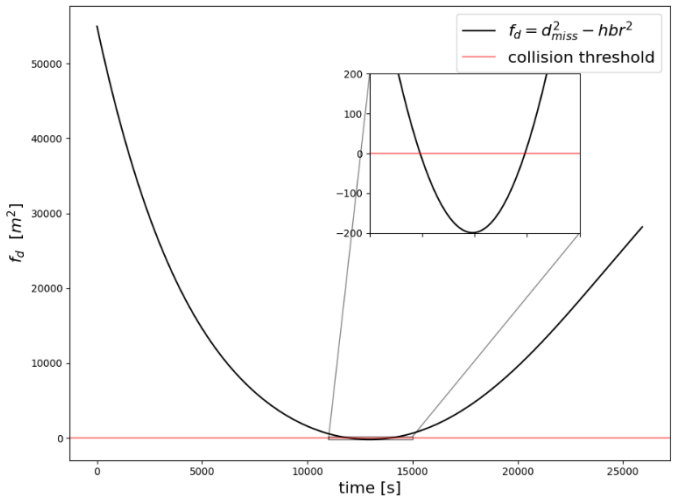
This parameter holds significant importance as it is analysed by the algorithm to determine if a collision occurs for a specific sample. In fact, the conventional numerical integration, is therefore simplified to the task of locating the roots of the function outlined in equation (10) within the defined bounds of the Taylor expansion. The process is illustrated in Fig. 2, where a sample is evaluated within the split initial set, resulting in a subset of one-dimensional polynomials that approximate the time evolution of the trajectory. For representation purposes, only spatial coordinates are shown.

C. Finding the roots of high order mono-dimensional time polynomials

The problem of finding roots for high-order mono-dimensional polynomials has been extensively researched, and various techniques exist that are considerably faster than numerical trajectory propagation. The root-finding algorithm is illustrated in. Fig. 3. After evaluating the sample and computing the DA expression in equation (9), we iterate through all the polynomials that define a sample trajectory. For each Taylor expansion and its corresponding definition interval, we initially verify if its centre falls into the negative range to rule out the possibility of finding no roots due to the segment trajectory already being below the collision threshold. Subsequently, we determine the number of sign changes of the polynomial coefficients using the Budan-Fourier theorem [18]. This theorem considers the number of roots by examining the sequence of coefficient sign variations in the polynomial. Specifically, if the number of sign changes is odd, it indicates the presence of at least one real root within the polynomial's interval. In such cases, the algorithm registers a hit and proceeds to analyse the next sample. If there are no sign changes (i.e., the count is zero), we conclude that the polynomial has no real root in the interval, and thus, the algorithm moves on to analyse the adjacent Taylor expansion as time

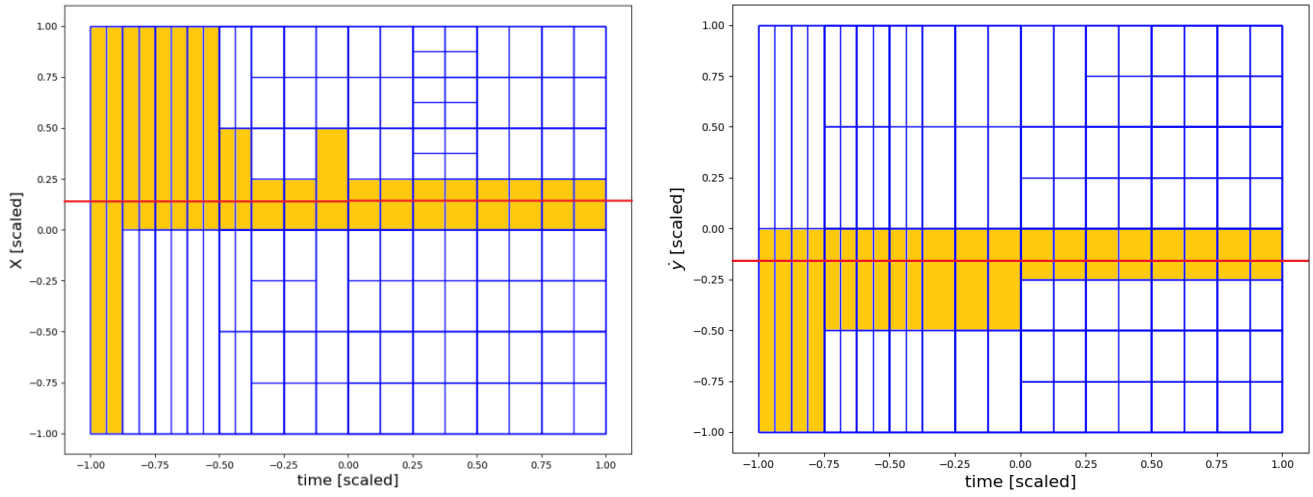


(a)



(b)

Fig. 5 – (a) PDF of the initial positional uncertainty for test case #1. (b) Time evolution of the initial condition mean expressed in term of f_d



(a) **(b)**

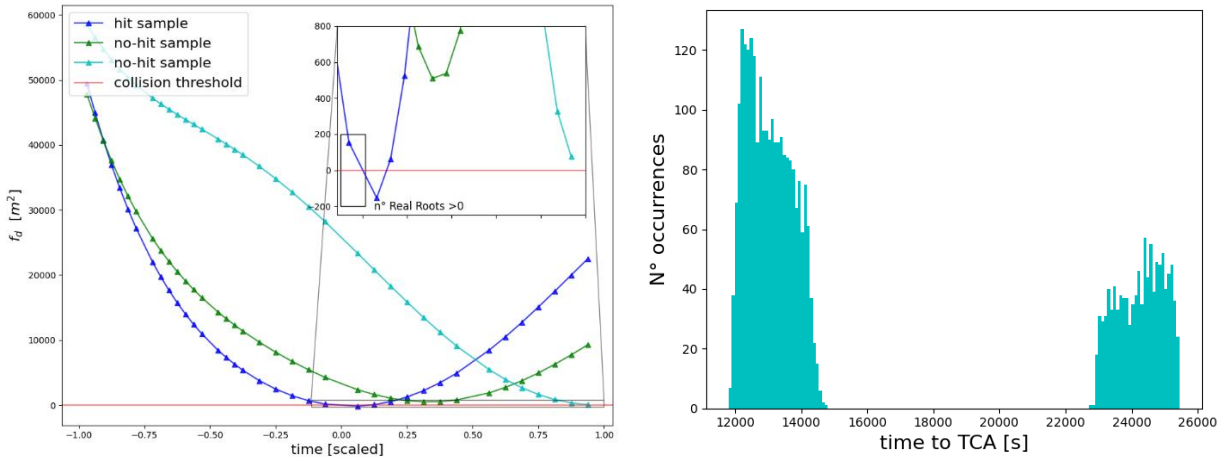
Fig. 6 –Subdomains time evolution of relative state (a) first- component and (b) fifth - component. In orange all the subdomains associated to a given sample trajectory.

progresses along the trajectory.

The situation differs when the number of sign changes is even. In such instances, the theorem does not provide conclusive results because the number of roots can be a multiple of an even number, potentially including zero. Therefore, the workflow further investigates using the Sturm algorithm [19]. This algorithm is a robust root isolation method that precisely determines the number of roots of a high-order polynomial within an interval by recursively performing Euclidean divisions to construct a sequence of polynomials. The sign variations in this sequence are analysed to ascertain the number of roots. As before, if there are no roots, the algorithm proceeds to analyse the neighbouring polynomials. However, if at least one root is found, a hit is recorded, and the polynomial approximation of the subsequent trajectory is studied.

D. Monte Carlo tool for validation

To verify our methodology and measure the efficiency of our approach, we compare the results obtained with those generated by a standard, in-house Monte Carlo simulation. To this aim, each sample is propagated from t_0 to T, symmetrically bracketing the TCA, using the state transition matrix once again. A grid of equally time-spaced points in miss distance is created, followed by interpolation where the real roots of a localized cubic polynomial are extracted. To elaborate, curve fitting is executed using a technique called parabolic blending [21], where a set of four equally spaced points is utilized to construct a third-order polynomial by merging two quadratic polynomials generated from the initial three points and the last three points. The minimum of the fitted curve is then determined by extracting the roots of the polynomial's first derivative. The Monte Carlo process then assesses whether a collision occurs for a given sample by checking if the relative distance at any point within the timeframe is equal to or less than the HBR. Figure 4 outlines the primary distinctions between our DA approach and a conventional Monte Carlo-based method for computing the Pc. Essentially, our technique requires only a single integration to generate the 7D manifold and a series of polynomial evaluation for each sample. Instead, in a standard Monte Carlo method, one



(a) **(b)**

Fig 7 – (a) Taylor expansions of f_d evaluated at the centre of each time sub-domain for two "hit" samples and one "no-hit" sample. (b) TCA distribution for test case #1.

must initially perform numerical propagation of each sample and then interpolate the grid of discrete points in relative distance.

IV. TESTING

We examine the Pc values obtained by our approach using a set of artificial test cases from [20]. These test cases are widely used in the literature as a benchmark for Pc computation methodologies, offering diverse scenarios suitable for both short-term and long-term analyses. Each test case provides the primary and secondary distributions at TCA in an inertial reference frame assuming a Gaussian distribution for the uncertainties in position and velocities. Given our methodology's reliance on relative dynamics, we compute the relative state and its related combined covariance at TCA and employ the inverse of the State Transition Matrix (STM), which integrates eq. (5) according to [17], to backpropagate to the initial time, t_0. Once the initial conditions have been derived consistently from those at TCA, random samples for the initial state can be generated. The standard deviations from the covariance matrix $\boldsymbol{P_0}$ are utilized to introduce random perturbations via Cholesky decomposition. To determine the minimum number of samples required for statistical significance, we adopt the same statistical bounding criteria as presented in [20].

The first test case considered involves two satellites in GEO with non-linear relative motion. Fig. 5-(a) illustrates the combined positional covariance sampled at the initial time. The red points represent a subset of samples, denoted as $\boldsymbol{V}$ in the preceding section, for which a collision occurs within the timeframe of analysis. As depicted in Fig. 5-(b), the test case was deliberately designed so that even the mean of the initial distribution results in a collision, leading to a notably high final Pc reference value of 0.21747. This elevated Pc level necessitates fewer than 16,000 Monte Carlo runs for the results to attain statistical significance.

As described before, the initial uncertainties are represented as a vector of DA variables and integrated over 25,926 seconds to position the TCA precisely at the midpoint of the time span. The evolution of the initial condition generates a 7D manifold, with the initial sub-domains established by the ADS routine. Fig 6 illustrates how the domain in the first and fifth components of the initial relative state and the time is split. Each subdomain defines the range in these variables for which a single Taylor expansion is able to represent the relative state vector with the required accuracy. The red line represents a particular determination of the initial relative state component, based on which all the associated sub-domains are selected. The evaluation of all these polynomials for a given sample of the initial relative state allows computing a DA expansion outlined in (10), which is a function only of the time. This is illustrated in Figure 7-(a), where the Taylor expansions of f_d are evaluated at the centre of each time sub-domain for three different samples. Two samples do not result in a collision (cyan and green curves), and one does lead to a collision (blue).

The cumulative collision probability determined through the DA polynomial evaluation method, stabilizes at 0.219172, mirroring the result obtained from our in-house Monte Carlo simulation. In comparison to the value cited in [20], our finding represents a minor overestimation of 0.78%. This discrepancy could be attributed to disparities in the dynamic models employed between [20] and our research. Over the analysis period, the initial conditions evolve in a manner that, on average, leads to an accumulation of Pc at two distinct times. The case illustrates the method's ability to address multiple conjunctions and, consequently, multiple TCAs within the analysis timeframe. Figure 7-(b) depicts the TCA distribution in the form of a histogram, clearly indicating that among the samples resulting in a collision, a subset hits around 2000 seconds after the start of the simulation. Successively, the Pc levels off about 2000 seconds after, and then increases again approximately 9500 seconds later. The computational time is compared to that of the in-house Monte Carlo simulation with an integration time-step of 5 second. As shown in Table 1, approximately 16,000 iterations are processed in roughly 7 second on an Intel(R) Core (TM) i7-8665U CPU @ 1.90GHz, resulting in a 50%-time reduction.

Additional test cases were examined to evaluate the method's performance when more than one million simulations were required to ensure statistically significant results. Case #2 features the same collision geometry as Case #1 but with a smaller HBR. In this instance, the resulting Pc is within 0.006% of the reference value and surpasses the in-house Monte Carlo method in terms of computational efficiency. The findings remain consistent for Case #4, which also entails non-linear relative motion in GEO. In this scenario, the computational time savings exceed 70%, and the computed Pc closely aligns with the Monte Carlo simulation. Overall, the Pc error remains within a 2% margin relative to the literature reference. On average, the reduction in computational time compared to our in-house Monte Carlo method is contingent upon

Case	n° Samples	DA Polynomials		Monte Carlo			Pc err [%]	Reference [20]	
		Pc	CPU time [s]	Δt [s]	Pc	CPU time [s]		Pc	Pc err [%]
# 1	1.57E+04	0.219172	7	5	0.219172	13	0	0.217467140	0.78
# 2	6.52E+06	0.016040	1721	5	0.016039	6685	0.006	0.015736620	1.90
# 4	1.24E+06	0.073810	385	5	0.073810	1331	4.06 E -04	0.073089530	0.99

Tab. 1 Comparison with Monte Carlo results and literature reference.

the number of samples utilized. For instance, with a relatively low sample count (Case #1), we observe approximately a 50% decrease. Conversely, when the sample count exceeds one million, the efficiency of performing a single DA integration becomes evident, resulting in a 70% improvement.

V. Outlook and Future Work

In this work, we introduced a general technique that computes the probability of collision between two space objects. We employ DA to characterize the non-linear evolution of the multivariate Gaussian initial relative state vector. The ADS algorithm regulates the accuracy of the dynamical integration, which is carried out under the assumption of linearized relative dynamics. The initial conditions' dynamical evolution takes the form of a patched 7D continuum, where each patch corresponds to a high order Taylor expansion, which is a function of the time and of the initial conditions. The initial PDF is then sampled and each patch is evaluated in a specific state-vector realization to definitively determine the time evolution of a given sample. The identification of collisions is reduced to finding the real roots of the DA approximation of the miss distance. Under the assumption of Gaussian initial uncertainty and Chloessy-Wiltshire relative-dynamics propagation, the technique can handle velocity uncertainties and intricate conjunction scenarios in which multiple conjunctions are allowed in the screening interval, releasing the assumption of [13]. It is worth mentioning that the methodology is, in principle, also able to handle any arbitrary initial statistic and dynamics.

The technique is tested against benchmark literature test cases. The error in the Pc estimation is always below 0.006% with respect to the in-house Monte Carlo, and below 2% with respect to the values reported in the literature. The methodology is very similar to a Monte Carlo based technique for computing the Pc, with the exception that only a single integration is necessary rather than propagating in time each sample. This resulted in a promising reduction of the computational time with respect to the classical Monte Carlo method. It's worth noting that the Monte Carlo method is already notably fast, given its analytical propagation of the dynamics [17] through the STM. The gain is up to 70% in the case of millions of samples.

In future work, the authors plan to increase the fidelity of the dynamics. Additionally, the technique needs to be further tested in a real-world scenario to determine how it performs with low Pc levels, particularly in terms of processing time. Even though the test cases are broadly used, they are completely artificial, resulting in unrealistically high Pc values typically not encountered in spacecraft operations.

VI. References

[1] Foster, J. A., Estes, H. S. "Parametric analysis of orbital debris collision probability and maneuver rate for space vehicles." *Technical Report. NASA* JSC, 1992.

[2] Patera, R. P. "General method for calculating satellite collision probability." *Journal of Guidance, Control, and Dynamics*, 2001, 24(4): 716–722.

[3] Patera, R. P. "Calculating collision probability for arbitrary space vehicle shapes via numerical quadrature." *Journal of Guidance, Control, and Dynamics*, 2005, 28(6): 1326–1328.

[4] Alfriend, K. T., Akella, M. R., Frisbee, J., Foster, J. L., Deok-Jin, L., Wilkins, M. "Probability of collision error analysis." *Space Debris*, 1999, 1(1): 21–35.

[5] Alfano, S. "A numerical implementation of spherical object collision probability." *The Journal of the Astronautical Sciences*, 2005, 53(1): 103–109.

[6] Chan, F. K. *Spacecraft Collision Probability*. Reston, VA, USA: Aerospace Press, 2008.

[7] Serra, R., Arzelier, D., Joldes, M., Lasserre, J. B., Rondepierre, A., Salvy, B. "Fast and accurate computation of orbital collision probability for short-term encounters." *Journal of Guidance, Control, and Dynamics*, 2016, 39(5): 1009–1021

[8] Patera, R. P. "Satellite collision probability for nonlinear relative motion." *Journal of Guidance, Control, and Dynamics*, 2003, 26(5): 728–733.

[9] Patera, R. P. "Collision probability for larger bodies having nonlinear relative motion." *Journal of Guidance, Control, and Dynamics*, 2006, 29(6): 1468–1472.

[10] Alfano, S. "Addressing nonlinear relative motion for spacecraft collision probability." In: *Proceedings of the AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, 2006: 6760

[11] F.K. Chan. "Hovering collision probability." In *AAS/AIAA Space Flight Mechanics Meeting*, number AAS 15-234, Williamsburg, VA, USA, January 2015.

[12] K. Chan. "Spacecraft collision probability for long-term encounters." Number AAS 03-549, Big Sky, Montana, USA, 2003.

[13] V.T. Coppola. "Including Velocity Uncertainty in the Probability of Collision between Space Objects." *Advances in the Astronautical Sciences*, 143, 2012.

[14] Denis Arzelier, Florent Bréhard, Mioara Joldeș, Jean-Bernard Lasserre, Sohie Laurens, et al. „Poly nomial superlevel set approximation of swept-volume for computing collision probability in space encounters." 2021. hal-03158347

[15] Armellin, R., Di Lizia, P., Bernelli Zazzera, F., Berz, M. "Asteroid close encounters characterization using differential algebra: the case of Apophis." *Celest. Mech. Dyn. Astron.* 107, 451–470 (2010)

[16] 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*, Vol. 122, No. 3, 2015, pp. 239–261.

[17] Clohessy, W. H.; Wiltshire, R. S. (1960). Terminal Guidance System for Satellite Rendezvous. *Journal of the Aerospace Sciences*. 27 (9): 653–658. doi:10.2514/8.8704

[18] Fourier, Jean Baptiste Joseph (1820). "Sur l'usage du théorème de Descartes dans la recherche des limites des racines". *Bulletin des Sciences, par la Société Philomatique de Paris*: 156–165

[19] Sturm, Jacques Charles François (1829). "Mémoire sur la résolution des équations numériques". *Bulletin des Sciences de Férussac*. 11: 419–425.

[20] Alfano S., *Satellite conjunction Monte Carlo analysis*. Advances in the Astronautical Sciences, 134:2007–2024, 2009.

[21] Alfano, S. "Determining Satellite Close Approaches," *Journal of the Astronautical Sciences*, Vol. 41, No. 2, April-June 1993, pp. 217-225