A high order method for orbital conjunctions analysis: sensitivity to initial uncertainties

Alessandro Morselli[1,*], Roberto Armellin[2], Pierluigi Di Lizia[2], Franco Bernelli Zazzera[3]

Dipartimento di Scienze e Tecnologie Aerospaziali, Politecnico di Milano, Via La Masa 34, 20156 Milano, Italy

Abstract

A high order method to quickly assess the effect that uncertainties produce on orbital conjunctions through a numerical high-fidelity propagator is presented. In particular, the dependency of time and distance of closest approach to initial uncertainties on position and velocity of both objects involved in a conjunction is studied. The approach relies on a numerical integration based on differential algebraic techniques and a high-order algorithm that expands the time and distance of closest approach in Taylor series with respect to relevant uncertainties. The modeled perturbations are atmospheric drag, using NRLMSISE-00 air density model, solar radiation pressure with shadow, third body perturbation using JPL's DE405 ephemeris, and EGM2008 gravity model. The polynomial approximation of the final position is used as an input to compute analytically the expansion of time and distance of closest approach. As a result, the analysis of a close encounter can be performed through fast, multiple evaluations of Taylor polynomials. Test cases with objects ranging from LEO to GEO regimes are considered to assess the performances and the accuracy of the proposed method.

Keywords: Space Debris; Conjunction Analysis; Differential Algebra

---

[*]Corresponding author

Email addresses: `alessandro.morselli@polimi.it` (Alessandro Morselli), `roberto.armellin@polimi.it` (Roberto Armellin), `pierluigi.dilizia@polimi.it` (Pierluigi Di Lizia), `franco.bernelli@polimi.it` (Franco Bernelli Zazzera)

[1]PhD candidate
[2]Postdoc fellow
[3]Full Professor

1. Introduction

The detection of orbital conjunctions between spacecraft and space debris is of fundamental importance in space situational awareness (SSA) programs. Once a potentially dangerous closest approach is identified, for instance by looking at the minimum distance between the objects, all information required to analyse the conjunction is provided to satellite operators that will compute the collision probability and evaluate the collision risk.

The collision risk depends on the geometry of the encounter and is considerably affected by the uncertainties of the orbital states at the time of closest approach (TCA). These uncertainties, in turn, depend on the uncertainties on initial state and their evolution along the orbit. When the TCA is far away from the reference epoch of the initial state, nonlinearities can play an important role in the computation of collision probability, since the initial covariance ellipsoid stretches and deforms after each revolution.

In the past, the conjunction assessment procedures relied on the catalog of unclassified objects orbiting the Earth, maintained by the United States Strategic Command (USSTRATCOM). The catalog is still available nowadays and contains all up-to-date two-line elements (TLE), that are intended for the use with the SGP4/SDP4 orbital model (Hoots et al., 2004). The Center for Space Standards and Innovation (CSSI) produces daily reports of closest conjunctions for the upcoming week using the program Satellite Orbital Conjunction Reports Assessing Threatening Encounters in Space (SOCRATES) (Kelso and Alfano, 2005). The information is publicly posted at www.celestrak.com/SOCRATES/. The conjunctions are identified using TLEs and SGP4/SDP4 and the tool also computes the maximum conjunction probability (Alfano, 2005), which represents the upper bound of collision probability and is obtained assuming the worst-case orientation and size of the covariance matrices.

When using TLEs and the SGP4/SGP4 analytical propagator, it has to be taken into account that the time of closest approach (TCA) can vary up to tens of seconds and the distance of closest approach (DCA) up to a few kilometers if more recent TLEs are considered for conjunction detection. Propagation accuracy of SGP4/SDP4 is indeed object dependent, and after a few days of propagation the errors can easily exceed tens of kilometres. In addition, potentially significant biases exist in TLE data (Kelso, 2007).

Nevertheless, when precise ephemeris of the chasers are not available, TLEs can represent a significant source of information. It is possible to estimate covariance information for TLEs by comparing states derived directly from the TLE data with states resulting from an orbit determination using pseudo-observations derived from TLE data (Alarcon-Rodriguez et al., 2004). Alternatively, it is possible to derive a covariance matrix differencing a set of TLEs propagated up to a common time (La Porte and Sasot, 2008; Vallado and Cefola, 2012).

Besides the publicly available TLE catalog, the Joint Space Operation Center (JSpOC) maintains an High-Accuracy Special Perturbation Catalog (Coffey et al., 1998) and releases Conjunction Summary Messages (CSM) to warn satellite operators of incoming close conjunctions (Aida and Kirschner, 2012). The CSM also provides the chaser covariance matrices at the TCA which can be used, together with satellite ephemeris, for accurate collision risk assessment.

The operational service for the assessment of collision risks of ESA satellites is based on the collision risk assessment software CRASS and the orbit determination software ODIN (Flohrer et al., 2009a). A daily automated screening is performed to identify close approached between covered missions and TLE from USSTRATCOM. When the estimated collision probability for an encounter exceeds a given threshold further data are acquired by the operator and processed by ODIN to improve orbit and covariance information. Methods were developed to estimate uncertainties associated with TLEs (Flohrer et al., 2009b). The process was adapted to take into account the CSM and analysis were performed to verify CSM against conjunction event analysis based on radar measurements (Flohrer et al., 2013).

In this work, a method for the computation of the DCA and TCA for all the objects compatible with the initial orbital uncertainties (referred to as virtual debris in the remainder of the paper) with a single numerical integration is presented. The method is based on the high order Taylor expansion of the flow of the dynamics enabled by differential algebraic (DA) techniques. In particular, a DA-based integrator and a polynomial inversion algorithm are used to express the dependence of TCA and DCA on orbital uncertainties in terms of high order Taylor polynomials (Armellin et al., 2010). As a result, the multiple integrations required by a Monte Carlo based approach for the computation of TCA and DCA for all the virtual debris are substituted by fast polynomial evaluations. The computation of these polynomials and the study of their accuracy is the main focus of the paper.

Their use for the fast and accurate computation of collision probability will be addressed in future work.

The paper is organized as follows. Firstly, some notes on DA techniques and the method for the high order expansion of the flow are introduced. A description of the dynamical model used follows. Then, the algorithm developed for the Taylor expansion of the TCA and DCA is presented. The performances of the numerical propagator in terms of computational time and expansion accuracy are first illustrated. Subsequently, the attention is focused on the accuracy analysis of TCA and DCA expansions.

2. Notes on Differential Algebra

DA techniques, exploited here to obtain k-th order Taylor expansions of the flow of a set of ODE's with respect to the initial conditions, were devised to attempt solving analytical problems through an algebraic approach (Berz, 1999b). Historically, the treatment of functions in numerics has been based on the treatment of numbers, and the classical numerical algorithms are based on the mere evaluation of functions at specific points. DA techniques rely on the observation that it is possible to extract more information on a function rather than its mere values. The basic idea is to bring the treatment of functions and the operations on them to a computer environment in a similar manner as the treatment of real numbers. Referring to Figure 1, consider two real numbers a and b. Their transformation into the floating point representation, $\overline{a}$ and $\overline{b}$ respectively, is performed to operate on them in a computer environment. Then, given any operation $*$ in the set of real numbers, an adjoint operation $\circledast$ is defined in the set of floating point (FP) numbers so that the diagram in Figure 1 commutes. (The diagram commutes only approximately in practice due to truncation errors.) This means , transforming the real numbers a and b into their FP representation and operating on them in the set of FP numbers returns the same result as carrying out the operation in the set of real numbers and then transforming the achieved result in its FP representation.

In a similar way, let us suppose two k differentiable functions f and g in n variables are given. In the DA framework, the computer operates on them using their k-th order Taylor expansions, F and G respectively. Therefore, the transformation of real numbers in their FP representation is now substituted by the extraction of the k-th order Taylor expansions of f and g. For each operation in the space of k times differentiable functions, an

4

$$a, b \in R \xrightarrow{\mathcal{T}} \bar{a}, \bar{b} \in FP \qquad f, g \xrightarrow{\mathcal{P}} F, G$$
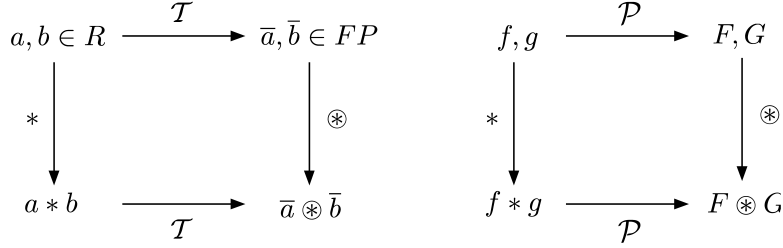
Figure 1: Analogy between the floating point representation of real numbers in a computer environment (left figure) and the introduction of the algebra of Taylor polynomials in the differential algebraic framework (right figure).

adjoint operation in the space of Taylor polynomials is defined so that the corresponding diagram commutes; i.e., extracting the Taylor expansions of f and g and operating on them in the space of Taylor polynomials (labeled as $_kD_n$) returns the same result as operating on f and g in the original space and then extracting the Taylor expansion of the resulting function.

The straightforward implementation of differential algebra in a computer enables the computation of the Taylor coefficients of a function up to a specified order k, along with the function evaluation, with a fixed amount of effort. The Taylor coefficients of order n for sums and products of functions, as well as scalar products with reals, can be computed from those of summands and factors; therefore, the set of equivalence classes of functions can be endowed with well-defined operations, leading to the so-called truncated power series algebra (Berz, 1986, 1987). Similarly to the algorithms for floating point arithmetic, the algorithms for functions followed, including methods to perform composition of functions, to invert them, to solve nonlinear systems explicitly, and to treat common elementary functions (Berz, 1999a,b). In addition to these algebraic operations, the DA framework is endowed with differentiation and integration operators, therefore finalizing the definition of the DA structure.

Some useful notes to get familiar with DA techniques are given hereafter, taking advantage of its detailed introduction by Berz (1999b). In particular, the minimal differential algebra for 1-dimensional functions and their first order expansion is illustrated in Sect. 2.1. Section 2.2 provides a clue of its extension to functions of n variables and k-th order expansions. Finally, the use of DA techniques to compute high order expansions of the flow of a general ODE is reported in Sect. 2.3, based on the description by Valli et al.

(2013).

2.1. The Minimal Differential Algebra

Consider all ordered pairs (q_0, q_1), with q_0 and q_1 real numbers. Define addition, scalar multiplication, and vector multiplication as follows:

$$
\begin{aligned}
(q_0, q_1) + (r_0, r_1) &= (q_0 + r_0, q_1 + r_1) \\
t \cdot (q_0, q_1) &= (t \cdot q_0, t \cdot q_1) \\
(q_0, q_1) \cdot (r_0, r_1) &= (q_0 \cdot r_0, q_0 \cdot r_1 + q_1 \cdot r_0),
\end{aligned} \tag{1}
$$

where $t \in \mathcal{R}$. The ordered pairs with the above arithmetic are called $_1D_1$. The multiplication of vectors is seen to have $(1, 0)$ as the unity element. The multiplication is commutative, associative, and distributive with respect to addition. Together, the three operations defined in Eq. (1) form an algebra. Furthermore, they form an extension of real numbers, as $(r, 0) + (s, 0) = (r + s, 0)$ and $(r, 0) \cdot (s, 0) = (r \cdot s, 0)$, so that the reals are included.

The multiplicative inverse of the pair (q_0, q_1) in $_1D_1$ is

$$
(q_0, q_1)^{-1} = \left(\frac{1}{q_0}, -\frac{q_1}{q_0^2} \right), \tag{2}
$$

which is defined for any $q_0 \neq 0$.

One important property of this algebra is that it has an order compatible with its algebraic operations. Given two elements (q_0, q_1) and (r_0, r_1) in $_1D_1$, the following is defined

$$
\begin{aligned}
(q_0, q_1) &< (r_0, r_1) \quad \text{if} \quad q_0 < r_0 \quad \text{or} \quad (q_0 = r_0 \text{ and } q_1 < r_1) \\
(q_0, q_1) &> (r_0, r_1) \quad \text{if} \quad (r_0, r_1) < (q_0, q_1) \\
(q_0, q_1) &= (r_0, r_1) \quad \text{if} \quad q_0 = r_0 \text{ and } q_1 = r_1.
\end{aligned} \tag{3}
$$

As for any two elements (q_0, q_1) and (r_0, r_1) only one of the three relation holds, $_1D_1$ is said totally ordered. The order is compatible with the addition and multiplication; for all $(q_0, q_1), (r_0, r_1), (s_0, s_1) \in {}_1D_1$, it follows $(q_0, q_1) < (r_0, r_1) \Rightarrow (q_0, q_1) + (s_0, s_1) < (r_0, r_1) + (s_0, s_1)$; and $(s_0, s_1) > (0, 0) = 0 \Rightarrow (q_0, q_1) \cdot (s_0, s_1) < (r_0, r_1) \cdot (s_0, s_1)$.

The number $d = (0, 1)$ has the interesting property of being positive but smaller than any positive real number; indeed $(0, 0) < (0, 1) < (r, 0) = r$.

For this reason d is called an infinitesimal or a differential. In fact, d is so small that its square vanishes. Since for any $(q_0, q_1) \in {}_1D_1$

$$(q_0, q_1) = (q_0, 0) + (0, q_1) = q_0 + d \cdot q_1, \tag{4}$$

the first component is called the real part and the second component the differential part.

The algebra in ${}_1D_1$ becomes a differential algebra by introducing a map ∂ from ${}_1D_1$ to itself, and proving that the map is a derivation. Define $\partial : {}_1D_1 \to {}_1D_1$ by

$$\partial(q_0, q_1) = (0, q_1). \tag{5}$$

Note that

$$
\begin{aligned}
\partial\{(q_0, q_1) + (r_0, r_1)\} &= \partial(q_0 + r_0, q_1 + r_1) = (0, q_1 + r_1) \\
&= (0, q_1) + (0, r_1) = \partial(q_0, q_1) + \partial(r_0, r_1),
\end{aligned} \tag{6}
$$

$$
\begin{aligned}
\partial\{t \cdot (q_0, q_1)\} &= \partial(t \cdot q_0, t \cdot q_1) = (0, t \cdot q_1) \\
&= t \cdot (0, q_1) = t \cdot \partial\{(q_0, q_1)\},
\end{aligned} \tag{7}
$$

and

$$
\begin{aligned}
\partial\{(q_0, q_1) \cdot (r_0, r_1)\} &= \partial(q_0 \cdot r_0, q_0 \cdot r_1 + r_0 \cdot q_1) = (0, q_0 \cdot r_1 + r_0 \cdot q_1) \\
&= (0, q_1) \cdot (r_0, r_1) + (0, r_1) \cdot (q_0, q_1) \\
&= \partial\{(q_0, q_1)\} \cdot (r_0, r_1) + (q_0, q_1) \cdot \partial\{(r_0, r_1)\}
\end{aligned} \tag{8}
$$

This holds for all $(q_0, q_1), (r_0, r_1) \in {}_1D_1$. Therefore ∂ is a derivation and $({}_1D_1, \partial)$ is a differential algebra.

The most important aspect of ${}_1D_1$ is that it allows the automatic computation of derivatives. Assume to have two functions $f, g \in \mathcal{C}^1(D)$, with $D \subseteq \mathcal{R}$. Put their values and their derivatives at $x \in D$ in the form $(f(x), f'(x))$ and $(g(x), g'(x))$ as two vectors in ${}_1D_1$. If the derivative of the product $f \cdot g$ is of interest, it has just to be looked at the second component of the product $(f(x), f'(x)) \cdot (g(x), g'(x))$ as defined in (1); whereas the first component gives the value of the product of the functions. Therefore, if two vectors contain the values and the derivatives of two functions, their product contains the values and the derivatives of the product function. Defining the operator $[\,]$ from the space of differentiable functions to ${}_1D_1$ via

$$[f] = (f(x), f'(x)), \tag{9}$$

7

and using (2), it holds

$$
\begin{aligned}
[f + g] &= [f] + [g] \\
[t \cdot f] &= t \cdot [f] \\
[f \cdot g] &= [f] \cdot [g],
\end{aligned}
\tag{10}
$$

where $t \in \mathcal{R}$, and

$$
[1/g] = [1]/[g] = 1/[g]
\tag{11}
$$

for all g with $g(x) \neq 0$. This observation can be used to compute derivatives of many kinds of functions algebraically by merely applying arithmetic rules on ${}_1D_1$, starting from the value and the derivative of the identity function $f(x) = x$, i.e., $[f] = [x] = (x, 1)$. Consider the example

$$
f(x) = \frac{1}{x + (1/x)}
\tag{12}
$$

and its derivative

$$
f'(x) = \frac{(1/x^2) - 1}{(x + (1/x))^2}.
\tag{13}
$$

The function value and its derivative at the point $x = 3$ are

$$
f(3) = \frac{3}{10}, \quad f'(3) = -\frac{2}{25}.
\tag{14}
$$

The same result can be obtained by replacing x with $[x]$ evaluated at $x = 3$, i.e., $(3, 1)$. Performing all the operations of (12) in the algebraic framework defined in (1) and (2) yields

$$
\begin{aligned}
f([x]) = \frac{1}{[x] + 1/[x]} &= \frac{1}{(3,1) + 1/(3,1)} = \frac{1}{(3,1) + (1/3, -1/9)} \\
&= \frac{1}{(10/3, 8/9)} = \left(\frac{3}{10}, -\frac{8}{9} \bigg/ \frac{100}{9}\right) = \left(\frac{3}{10}, -\frac{2}{25}\right).
\end{aligned}
\tag{15}
$$

Thus, the real part of the result is the value of the function at $x = 3$, whereas the differential part is the value of the derivative of the function at $x = 3$. This is expected as, by applying the relations (10) and (11) to compute the

$_1D_1$ representative of f,

$$
\begin{aligned}
[f] &= \left[\frac{1}{x+1/x}\right] = \frac{1}{[x+1/x]} \\
&= \frac{1}{[x]+[1/x]} = \frac{1}{[x]+1/[x]} \\
&= f([x]).
\end{aligned}
\tag{16}
$$

It is worth highlighting that DA circumvents the analytical derivation of $f'(x)$ as performed in (13) to compute the derivative of f at $x = 3$.

2.2. The Differential Algebra $_kD_n$

This section extends the algebra $_1D_1$ to the general $_kD_n$ case, which enables the computation of the derivatives of functions in n variables up to order k. Similarly as before, it is based on taking the space $\mathcal{C}^k(D)$, the collections of k times continuously differentiable functions on D, where $D \subseteq \mathcal{R}^n$. On this space an equivalence relation is introduced. For f and $g \in \mathcal{C}^k(D)$, $f =_k g$ if and only if $f(x) = g(x)$ with $x \in D$ and all the partial derivatives of f and g agree at x up to order k. The relation $=_k$ satisfies

$$
\begin{aligned}
f &=_k f \quad \text{for all} \quad f \in \mathcal{C}^k(D), \\
f &=_k g \Rightarrow g =_k f \quad \text{for all} \quad f, g \in \mathcal{C}^k(D), \quad \text{and} \\
f &=_k g \text{ and } g =_k h \Rightarrow f =_k h \quad \text{for all} \quad f, g, h, \in \mathcal{C}^k(D).
\end{aligned}
\tag{17}
$$

Thus, $=_k$ is an equivalence relation. All the elements that are related to f can be grouped together in one set, the equivalence class $[f]$ of the function f. The resulting equivalence classes are often referred to as DA vectors or DA numbers. Intuitively, each of these classes is then specified by a particular collection of partial derivatives in all n variables up to order k. The set of these classes is called $_kD_n$.

If the values and the derivatives of two functions f and g are known, the corresponding values and derivatives of $f + g$ and $f \cdot g$ can be inferred. Therefore, the arithmetics on the classes in $_kD_n$ can be introduced via

$$
\begin{aligned}
[f + g] &= [f] + [g] \\
[t \cdot f] &= t \cdot [f] \\
[f \cdot g] &= [f] \cdot [g],
\end{aligned}
\tag{18}
$$

9

where $t \in \mathcal{R}$. Under this operations, $_kD_n$ becomes an algebra. Note that the algebra $_1D_1$ is a particular case of $_kD_n$. This is justified by the fact that the equivalence relation in (17) holds for the special case $k = n = 1$; moreover, the algebra in (18) is identical to the $_1D_1$ counterpart in (10). Thus, the definition of $_1D_1$ in Sect. 2.1 is compatible with its $_kD_n$ extension; therefore, no difference in notation is made.

For each $v \in 1, \ldots, n$, define now the map ∂_v from $_kD_n$ to $_kD_n$ for f via

$$\partial_v[f] = \left[p_v \cdot \frac{\partial f}{\partial x_v} \right], \tag{19}$$

where

$$p_v(x_1, \ldots, x_n) = x_v \tag{20}$$

projects out the v-th component of the identity function. It is easy to show that for all $v = 1, \ldots, n$ and for all $[f], [g] \in {}_kD_n$

$$
\begin{aligned}
\partial_v([f] + [g]) &= \partial_v[f] + \partial_v[g] \\
\partial_v(t \cdot [f]) &= t \cdot \partial_v[f] \\
\partial_v([f] \cdot [g]) &= [f] \cdot (\partial_v[g]) + (\partial_v[f]) \cdot [g],
\end{aligned} \tag{21}
$$

where $t \in \mathcal{R}$. Therefore, ∂_v is a derivation for all v, and hence $({}_kD_n, \partial_1, \ldots, \partial_v)$ is a differential algebra.

Observe that f lies in the same class as its Taylor polynomial T_f of order k around the origin; they have the same function values and derivatives up to order n. Therefore,

$$[f] = [T_f] \tag{22}$$

and the introduced differential algebra is referred to as Taylor polynomial algebra.

The structure $_kD_n$ can be endowed with the composition of functions and the treatment of common elementary functions such as exp, sin, and log. Consequently, the derivatives of any function $f \in \mathcal{C}^k(D)$, where $D \subseteq \mathcal{R}^n$, and composed of the elementary functions can be computed up to order k in fixed amount of effort. The DA sketched in this section was implemented by M. Berz and K. Makino in the software COSY-Infinity (Berz and Makino, 2006).

10

2.3. High Order Expansion of the Flow

Differential algebra allows the derivatives of any function f of n variables to be computed up to an arbitrary order k, along with the function evaluation. This has an important consequence when the numerical integration of an ODE is performed by means of an arbitrary integration scheme. Any integration scheme is based on algebraic operations, involving the evaluation of the ODE right hand side at several integration points. Therefore, carrying out all the evaluations in the DA framework allows differential algebra to compute the arbitrary order expansion of the flow of a general ODE with respect to the initial condition.

Without loss of generality, consider the scalar initial value problem

$$\begin{cases} \dot{x} = f(x,t) \\ x(t_0) = x_0 \end{cases} \tag{23}$$

and the associated phase flow $\varphi(t; x_0)$. We now want to show that, starting from the DA representation of the initial condition x_0, differential algebra allows us to propagate the Taylor expansion of the flow in x_0 forward in time, up to the final time t_f.

To this aim, replace the point initial condition x_0 by the DA representative of its identity function up to order k, which is a $(k+1)$-tuple of Taylor coefficients. (Note that x_0 is the flow evaluated at the initial time; i.e, $x_0 = \varphi(t_0; x_0)$.) As for the identity function only the first two coefficients, corresponding to the constant part and the first derivative respectively, are non zeros, we can write $[x_0]$ as $x_0 + \delta x_0$, where x_0 is the reference point for the expansion. If all the operations of the numerical integration scheme are carried out in the DA framework, the phase flow $\varphi(t; x_0)$ is approximated, at each fixed time step t_i, as a Taylor expansion in x_0.

For the sake of clarity, consider the forward Euler's scheme

$$x_i = x_{i-1} + f(x_{i-1})\Delta t \tag{24}$$

and substitute the initial value with the DA identity $[x_0] = x_0 + \delta x_0$. At the first time step we have

$$[x_1] = [x_0] + f([x_0]) \cdot \Delta t. \tag{25}$$

If the function f is evaluated in the DA framework, the output of the first step, $[x_1]$, is the k-th order Taylor expansion of the flow $\varphi(t; x_0)$ with respect

11

to x_0 at $t = t_1$. Note that, as a result of the DA evaluation of $f([x_0])$, the $(k + 1)$-tuple $[x_1]$ may include several non zero coefficients corresponding to high order terms in δx_0. The previous procedure can be inferred through the subsequent steps. The result of the final step is the k-th order Taylor expansion of $\varphi(t; x_0)$ in x_0 at the final time t_f. Thus, the flow of a dynamical system can be approximated, at each time step t_i, as a k-th order Taylor expansion in x_0 in a fixed amount of effort. In addition, the high order expansion of the flow with respect to the final time is obtained when the final time is initialized as DA variable. Consider the last step of the forward Euler's scheme where the initial value $[x_0]$ is a DA variable and f is evaluated in a DA environment:

$$[x_N] = [x_{N-1}] + f([x_{N-1}])\Delta t. \tag{26}$$

Since the final time is initialized as a DA variable, i.e.

$$[t_f] = t_f + \delta t_f, \tag{27}$$

the last time step Δt will be a DA variable as well, yielding

$$[x_N] = [x_{N-1}] + f([x_{N-1}])[\Delta t] = [x_{N-1}] + f([x_{N-1}])\left([t_f] - t_{n-1}\right). \tag{28}$$

As a result, the high order expansion of the flow with respect to final time variation is obtained. This plays a crucial role in the computation of the Taylor expansion of the TCA, as illustrated in Sec. 4.2.

The conversion of standard integration schemes to their DA counterparts is straightforward both for explicit and implicit solvers. This is essentially based on the substitution of the operations on real numbers with those on DA numbers. In addition, whenever the integration scheme involves iterations (e.g. iterations required in implicit and predictor-corrector methods), step size control, and order selection, a measure of the accuracy of the Taylor expansion of the flow needs to be included.

Integration schemes based on DA pave the way to the algorithms presented in this paper. A first example is given hereafter about the propagation of uncertainties on initial conditions. The Taylor polynomials resulting from the use of DA-based numerical integrators expand the solution of the initial value problem presented in Eq. (23) with respect to the initial condition. Thus, the dependence of the solution $x(t)$ on the initial condition is available, at a time t_i, in terms of a k-th order polynomial map $\mathcal{M}_{x_0}(\delta x_0)$, where

12

δx_0 is the displacement from the reference initial condition. The evaluation of the map $\mathcal{M}_{x_0}(\delta x_0)$ for a selected value of δx_0 supplies the k-th order Taylor approximation of the solution at t_i corresponding to the perturbed initial condition. The accuracy of the result depends on the function f, the expansion order k, and the value of the displacement δx_0.

The main advantage of the DA-based approach is that there is no need to write and integrate variational equations in order to obtain high order expansions of the flow. This result is basically obtained by the substitution of operations between real numbers with those on DA numbers, and therefore the method is ODE independent. Furthermore, the efficient implementation of the differential algebra in COSY-Infinity allows us to obtain high order expansions with limited computational time.

3. Numerical propagator

The numerical propagator AIDA (Accurate Integrator for Debris Analysis) is implemented to perform the expansions of TCA and DCA. The perturbations included in AIDA are the geopotential acceleration, atmospheric drag, solar radiation pressure, and third body gravity . Details on the modeling of these sources of perturbation are given in the following. For the numerical integrations presented in this paper, a DA version of the Dormand-Prince (8-th order solution for propagation, 7-th order solution for step size control) implementation of the Runge-Kutta scheme is used. A weighted norm of the resulting Taylor polynomials is computed within the step size control procedure, where the weights are selected to scale the polynomial variables by the size of their initial uncertainty.

3.1. Geopotential acceleration

The acceleration due to Earth's gravity potential can be written as

$$\boldsymbol{a}_{\text{Harm}} = \boldsymbol{\nabla} \frac{GM_E}{r} \sum_{n=0}^{\infty} \sum_{m=0}^{n} \frac{R_E^{\,n}}{r^n} \overline{P}_{nm} \left(\sin \phi \right) \left(\overline{C}_{nm} \cos \left(m\lambda \right) + \overline{S}_{nm} \sin \left(m\lambda \right) \right),$$

(29)

where GM_E is Earth's gravitational constant, R_E is Earth's radius, $\overline{C}_{nm}$ and $\overline{S}_{nm}$ are the normalized geopotential coefficients, $\overline{P}_{nm}$ are the normalized associated Legendre functions, r is the object distance from the centre of the Earth, and ϕ and λ are the geocentric latitude and longitude.

The gravitational model selected for the numerical propagator is EGM2008 (Pavlis et al., 2012). The model combines gravitational information from GRACE with surface data and is complete to spherical harmonic degree and order 2160. The field model was downloaded from the International Centre for Global Earth Models (ICGEM) website [4].

In EGM2008, Earth's gravity constant is equal to $3.986004415 \times 10^{14}$ m^3/s^2 and the Earth radius is 6.3781363×10^6 m. Earth angular velocity, ω_E, is set to 7.292115×10^{-5} rad/s and flattening parameter is $f = 1/298.257$.

The default degree n and order m in Eq. (29) are set to 10 for the simulations performed in this paper.

3.2. Atmospheric drag

The perturbing acceleration due to atmospheric drag on a satellite is

$$\boldsymbol{a}_{\text{Drag}} = -\frac{1}{2} C_D \frac{A}{M} \rho \, v_r \boldsymbol{v}_r, \tag{30}$$

where C_D is the drag coefficient, A is the satellite's cross-sectional area and M its mass, $\boldsymbol{v}_r$ is the satellite velocity relative to the atmosphere , and ρ is the atmospheric density at satellite's position.

The drag coefficient and area to mass ratio are often grouped in a parameter known as ballistic coefficient, B, defined as

$$B = C_D \frac{A}{M}. \tag{31}$$

The Naval Research Laboratory's Mass Spectrometer and Incoherent Scatter Radar of year 2000 (NRLMSISE-00) model (Picone et al., 2002) was selected to compute the atmospheric density. This model includes the anomalous oxygen component together with Helium, atomic and molecular Oxygen, atomic and molecular Nitrogen, Argon, and Hydrogen. The model requires as inputs the solar and geomagnetic activity, geodetic altitude and latitude, longitude, year, day, and time of day in UT. Solar and geomagnetic data are read from up-to-date space weather files that are automatically downloaded from CelesTrack [5].

---

[4]http://icgem.gfz-potsdam.de/ICGEM/
[5]http://celestrak.com/SpaceData/SpaceWx-format.asp

3.3. Third body perturbations

The gravitational attraction of the Sun and the Moon is the main source of perturbation among all third bodies. The perturbing acceleration acting on the orbiting object is given by

$$\boldsymbol{a}_{\text{3rdB}} = GM_{\text{3rdB}} \left(\frac{\boldsymbol{s}}{s^3} - \frac{\boldsymbol{r}_{\text{3rdB}}}{r_{\text{3rdB}}^3} \right), \tag{32}$$

where the relative position of the satellite, $\boldsymbol{s}$, is defined as

$$\boldsymbol{s} = \boldsymbol{r}_{\text{3rdB}} - \boldsymbol{r}_{\text{obj}}, \tag{33}$$

where $\boldsymbol{r}_{\text{3rdB}}$ and $\boldsymbol{r}_{\text{obj}}$ are the position vectors of the third body and the object, respectively. The position of the third bodies, i.e. the Sun and the Moon, are computed using NASA JPL's DE405 ephemeris (Standish, 1998).

3.4. Solar radiation pressure

The absorption or reflection of photons exerts a small force on the satellite. This perturbing acceleration depends on object area and mass as well as on the intensity of the solar flux Φ, that is approximately 1367 W/m^2 at 1 AU. The solar radiation pressure, P_S, is given, assuming full absorption, by

$$P_S = \frac{\Phi}{c} = \frac{L_S}{4\pi r^2 c}, \tag{34}$$

where the speed of light is indicated as c, the distance from the Sun as r, and L_S is the solar luminosity.

The information regarding shape and attitude of operative satellites are usually available, at least for satellite owners and operators. Unfortunately, in general, this is not the case for space debris. Therefore, the debris is modeled as a sphere in this work, removing the need for attitude information. The resulting acceleration, which is in the direction of the Sun-satellite vector, is equal to

$$\boldsymbol{a}_{\text{SRP}} = \frac{L_S}{4\pi c} \frac{A}{M} \left(1 + \varepsilon \right) \frac{\boldsymbol{r}_{\text{obj}} - \boldsymbol{r}_S}{||\boldsymbol{r}_{\text{obj}} - \boldsymbol{r}_S||^3} \nu, \tag{35}$$

where A is the debris area and M its mass, ε is body reflectivity, and ν is the shadow function. The position of the Sun, $\boldsymbol{r}_S$, is in ECI J2000 reference frame and is obtained from JPL's DE405 ephemeris. The acceleration $\boldsymbol{a}_{\text{SRP}}$ decreases as Sun's distance increases.

Area-to-mass ratio can be estimated from the object ballistic coefficient, assuming a drag coefficient $C_D = 2.2$. The reflectivity ε, if not otherwise specified, is set to 0.31. This value lies in the range typical of materials used for satellite construction, i.e. $[0.2; 0.9]$ (Montenbruck and Gill, 2000).

The shadow function ν is zero when the object is in shadow, one when it is in light, and varies within these two values when the object is in penumbra. The value of this function is computed by comparing the satellite position with the one of the occulting body and the Sun (Montenbruck and Gill, 2000). Three options to model the shadow of an occulting body are implemented in AIDA: no shadow, cylindrical shadow, and dual-cone shadow. Dual-cone shadow model is the default model.

4. DCA and TCA expansion

Uncertainties in orbital measurements and orbital determination process result in uncertainties in the initial orbital state of the space objects. Consequently, each pair of virtual debris is characterized by a different value of TCA. In this section the procedure to obtain the Taylor expansion of TCA and DCA is described, provided that a first guess of TCA is available.

4.1. First guess of TCA

A method to compute first guess values of TCA and DCA was developed in (Armellin et al., 2012). This method is based on the propagation of TLE with SGP4/SDP4 and the identification of the conjunction by the rigorous solution of a global optimization problem with the algorithm COSY-GO (Makino and Berz, 2005).

The approach enables the computation of both the global minimum and all local minima of the squared distance function in the time window of interest. In the first case, the selected objective function is the square of the relative distance, i.e.

$$J_{GM}(t) = d(t)^2 = (\boldsymbol{r}_2 - \boldsymbol{r}_1) \cdot (\boldsymbol{r}_2 - \boldsymbol{r}_1), \tag{36}$$

where $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ are the position vectors of the first and the second object, and the subscript GM stands for global minimum.

When all the minima are searched for, the time derivative of the squared distance is computed first

$$\frac{\mathrm{d}d(t)^2}{\mathrm{d}t} = 2d\dot{d} = 2(\boldsymbol{v}_2 - \boldsymbol{v}_1) \cdot (\boldsymbol{r}_2 - \boldsymbol{r}_1). \tag{37}$$

16

Then the objective function is set to

$$J_{SP}(t) = [(\boldsymbol{v}_2 - \boldsymbol{v}_1) \cdot (\boldsymbol{r}_2 - \boldsymbol{r}_1)]^2, \tag{38}$$

where SP stands for stationary points.

Note that Eq. (38) is positive semidefinite, and its zeros (i.e., its global minima) are stationary points of the distance function. In both global minimum and local minima research, the time window is set to one week starting from the epoch of the most recent TLE, retrieved from Space-Track[6]. This method is here used to obtain an estimate of TCA and DCA, but any other method can be used for this purpose. In the remainder of the paper, the first guess values of TCA and DCA identified by this method are referred to as t^*_{FG} and d^*_{FG}, respectively.

4.2. TCA and DCA expansion algorithm

The aims of the algorithm presented in this section are:

1. Starting from t^*_{FG} and d^*_{FG}, compute the nominal value of TCA and DCA when SGP4/SDP4 propagation is substituted by AIDA propagation. These values will be referred to as t^* and d^*.
2. Compute the Taylor expansion of t^* and d^* with respect to the initial conditions of the two objects:

$$[t^*] = t^* + \left(\delta\boldsymbol{x}_0^1, \delta\boldsymbol{x}_0^2\right)$$
$$[d^*] = d^* + \left(\delta\boldsymbol{x}_0^1, \delta\boldsymbol{x}_0^2\right) \tag{39}$$

where $\boldsymbol{x}_0^1$ and $\boldsymbol{x}_0^2$ are vectors of six elements (any set of orbital elements, e.g. Keplerian elements, Delunay variables, position and velocity in any arbitrary reference frame, can be used).

The algorithm starts with the initialization of the initial orbital states and the final time as DA variables

$$\left[\boldsymbol{x}_0^1\right] = \boldsymbol{x}_0^1 + \delta\boldsymbol{x}_0^1$$
$$\left[\boldsymbol{x}_0^2\right] = \boldsymbol{x}_0^2 + \delta\boldsymbol{x}_0^2, \tag{40}$$

$$[t_f] = t^*_{FG} + \delta t. \tag{41}$$

---

[6]https://www.space-track.org

Note that the nominal value of the final time is set to the first guess TCA delivered by the global optimizer.

The resulting objects initial conditions are propagated forward with AIDA and, according to Section 2.3, the Taylor polynomials of the final state are obtained as function of time and initial uncertain states

$$\left[\boldsymbol{x}_f^1\right] = \boldsymbol{x}_f^1 + \mathcal{M}_{\boldsymbol{x}_f^1}\left(\delta t,\ \delta\boldsymbol{x}_0^1\right)$$

$$\left[\boldsymbol{x}_f^2\right] = \boldsymbol{x}_f^2 + \mathcal{M}_{\boldsymbol{x}_f^2}\left(\delta t,\ \delta\boldsymbol{x}_0^2\right). \tag{42}$$

In Eq. (42), $\boldsymbol{x}_f^1$ and $\boldsymbol{x}_f^2$ are the nominal final positions of the two objects at time t_{FG}^*, whereas $\mathcal{M}_{\boldsymbol{x}_f^1}\left(\delta t,\ \delta\boldsymbol{x}_0^1\right)$ and $\mathcal{M}_{\boldsymbol{x}_f^2}\left(\delta t,\ \delta\boldsymbol{x}_0^2\right)$ are the higher order terms of the Taylor polynomials that describe how changes in both final time and initial conditions affect the final states.

After the integration, the squared relative distance can be computed in the DA framework, resulting in

$$\left[d^2\right] = d^2 + \mathcal{M}_{d^2}\left(\delta t,\ \delta\boldsymbol{x}_0^1,\ \delta\boldsymbol{x}_0^2\right), \tag{43}$$

which is a Taylor polynomial of 13 variables. By using the operator of differentiation implemented in COSY-Infinity, the Taylor polynomial

$$\left[\frac{\partial d^2}{\partial t}\right] = c_0 + \mathcal{M}_{\frac{\partial d^2}{\partial t}}\left(\delta t,\ \delta\boldsymbol{x}_0^1,\ \delta\boldsymbol{x}_0^2\right) \tag{44}$$

is computed. Note that the constant part c_0 is not zero as the numerical integration in AIDA is stopped at t_{FG}^*, which is the nominal TCA when SGP4/SDP4 is used for the propagation. By subtracting c_0 to (44), and defining $\delta\frac{\partial d^2}{\partial t} = \left[\frac{\partial d^2}{\partial t}\right] - c_0$, the following origin preserving augmented map can be built

$$\begin{bmatrix} \delta\frac{\partial d^2}{\partial t} \\ \delta\boldsymbol{x}_0^1 \\ \delta\boldsymbol{x}_0^2 \end{bmatrix} = \begin{bmatrix} \mathcal{M}_{\frac{\partial d^2}{\partial t}} \\ \mathcal{I}_{\delta\boldsymbol{x}_0^1} \\ \mathcal{I}_{\delta\boldsymbol{x}_0^2} \end{bmatrix} \begin{bmatrix} \delta t \\ \delta\boldsymbol{x}_0^1 \\ \delta\boldsymbol{x}_0^2 \end{bmatrix}, \tag{45}$$

where identities in the variation of the initial states are added. This polynomial map can then be inverted using suitable polynomials inversion techniques (Berz, 1999a) obtaining

$$\begin{bmatrix} \delta t \\ \delta\boldsymbol{x}_0^1 \\ \delta\boldsymbol{x}_0^2 \end{bmatrix} = \begin{bmatrix} \mathcal{M}_{\frac{\partial d^2}{\partial t}} \\ \mathcal{I}_{\delta\boldsymbol{x}_0^1} \\ \mathcal{I}_{\delta\boldsymbol{x}_0^2} \end{bmatrix}^{-1} \begin{bmatrix} \delta\frac{\partial d^2}{\partial t} \\ \delta\boldsymbol{x}_0^1 \\ \delta\boldsymbol{x}_0^2 \end{bmatrix}. \tag{46}$$

18

The goal is to compute δt such that $\left[\frac{\partial d^2}{\partial t}\right] = 0$. This is obtained by evaluating (46) in $\delta \frac{\partial d^2}{\partial t} = -c_0$

$$
\begin{bmatrix} \delta t \\ \delta \boldsymbol{x}_0^1 \\ \delta \boldsymbol{x}_0^2 \end{bmatrix} = \begin{bmatrix} \mathcal{M}_{\frac{\partial d^2}{\partial t}} \\ \mathcal{I}_{\delta \boldsymbol{x}_0^1} \\ \mathcal{I}_{\delta \boldsymbol{x}_0^2} \end{bmatrix}^{-1} \begin{bmatrix} -c_0 \\ \delta \boldsymbol{x}_0^1 \\ \delta \boldsymbol{x}_0^2 \end{bmatrix}. \tag{47}
$$

Note that the first row of the map in Eq. (47) is a Taylor polynomial with 1) a constant part that is the correction to t_{FG}^* necessary to impose the stationarity of d^2 in the AIDA propagator 2) higher order terms that approximate how the TCA changes depending on the uncertainties in the initial conditions. By plugging the first row of Eq. (47) into Eq. (41) the following result is achieved

$$
[t_f] = [t^*] = t^* + \mathcal{M}_{t^*}(\delta \boldsymbol{x}_0^1, \delta \boldsymbol{x}_0^2), \tag{48}
$$

which is the high order Taylor expansion of the TCA evaluated with the numerical integrator AIDA. Similarly, the high order expansion of the squared distance is obtained by inserting the first row of Eq. (47) into Eq. (43), yielding

$$
[d^{*2}] = d^{*2} + \mathcal{M}_{d^{*2}}\left(\delta \boldsymbol{x}_0^1, \delta \boldsymbol{x}_0^2\right). \tag{49}
$$

For any perturbed initial condition of the two objects (i.e., for any pair of virtual debris) the evaluation of the Taylor polynomials in Eq. (48) and (49) delivers the associated values of TCA and DCA. Of course, these values are not exact: their accuracy depends on the selected expansion order, the propagation window, and the range of the uncertainties on the initial conditions.

5. Numerical Examples

In this section the performance of the proposed method for the computation of the expansion of TCA and DCA is assessed. This analysis is split into three phases with the goal of investigating:

1. The computational cost and accuracy of the expansion of the flow, aimed at the selection of the expansion order;
2. The accuracy of the computation of the nominal TCA and of the expansion in time;

3. The accuracy of the TCA and DCA Taylor expansions.

The six objects listed in Table 1 (either active spacecraft or debris) are considered for these analyses. The first three objects are classified as Low Earth Orbit (LEO) objects, since their altitude is below 2,000 km. In particular the altitude of the first two objects is comprised between 750 and 800 km, whereas for object 3 it is approximately 1400 km. Object 4 in an Highly Elliptical Orbit (HEO) since it has an altitude of 1340 km at its perigee and at its apogee is at 5700 km above the Earth's surface. Objects 5 and 6 instead are on a Geostationary Earth Orbit (GEO) since their orbit is almost circular, with a low inclination, and an orbital period of nearly 24 hours.

Table 1: Selected objects

ID	Sat. No.	Sat. Name	a [km]	e [-]	I [deg]	Orbit
1	11510	COSMOS 1125	7161	1.15E-3	74.04	LEO
2	21574	ERS 1	7149	3.49E-3	98.25	LEO
3	20237	COSMOS 2043	7787	4.19E-4	82.57	LEO
4	23820	OPS 0856 DEB	9895	2.20E-1	87.62	HEO
5	36744	COMS 1	42165	7.88E-5	0.01	GEO
6	28194	RADUGA 1-7	42166	1.21E-4	5.28	GEO

The initial conditions for the numerical propagation are computed from TLEs, using the procedure described in Vallado and Cefola (2012). The method consists of performing an Orbit Determination (OD) process on a set of pseudo-observations generated over a certain time span. In our case we generated one measure, i.e. position and velocity of the satellite, every hour over a time span of 24 h, centered at the reference epoch of the selected TLE. The resulting state vectors were then transformed from TEME of epoch to ECI J2000 (Vallado, 2001), that is the reference frame of AIDA. A nonlinear least square fit of the pseudo-observations was then performed using AIDA, to compute the initial position and velocity of the object, and its ballistic coefficient and area-to-mass ratio for SRP. The (6×6) covariance matrix of initial position and velocity was also obtained after this process.

The values of initial conditions, covariance matrixes, and ballistic coefficients of the objects used for the following analyses are reported in Appendix A. All computations are performed on an Intel Core i5 2500 @3.30GHz, 8Gb RAM processor running Sabayon Linux 10 (kernel version 3.5.0).

5.1. Performances of the differential algebraic numerical propagation

The selection of the expansion order requires a trade-off between accuracy and computational time. This analysis is performed by propagating the objects for a time window of 7 days (which is the time window for conjunction identification) with uncertainties on initial states compatible with the estimated covariance matrixes.

The computational time required for each run of AIDA is reported in Table 2 for zeroth (i.e., pointwise integration) to fourth order expansions. Only one test case for each orbital regime is given, since similar results are obtained for objects in similar orbits. The computational time required to propagate each object for one orbit is also given in Table 2. It can be observed that the computational time per revolution is nearly the same for each of the three orbits for a given order of truncation. The total computational time for 7 days propagation clearly depends on the orbital period of the orbits, i.e. on the number of orbital revolutions completed in the considered time span.

Table 2: Computational costs for 7 days propagation and for each revolution with AIDA for different expansion orders

Order	Computational time [s]			Comp. time per rev. [s]		
	LEO	HEO	GEO	LEO	HEO	GEO
0	63.53	37.65	6.45	0.63	0.61	0.91
1	98.81	55.87	9.38	0.98	0.90	1.33
2	176.31	98.19	15.85	1.75	1.59	2.26
3	443.77	234.87	41.38	4.41	3.80	5.90
4	1391.02	690.77	110.80	13.84	11.19	15.79

It is worth highlighting that the computational time of pointwise integrations is not the main focus of the following analysis, since it depends on the details of the implementation of the dynamical model, programming language, compiler, and machine. The significant figure to be investigated is the time of a k-th order computation compared to a pointwise one. For this reason, the computational times can be normalized using the pointwise integration as reference value. The ratios are plotted, for each orbital regime, in Figure 2. It is clear that the computational time increases exponentially with the order, but it is remarkable that a fourth order expansion is roughly only ten times slower than a pointwise integration.
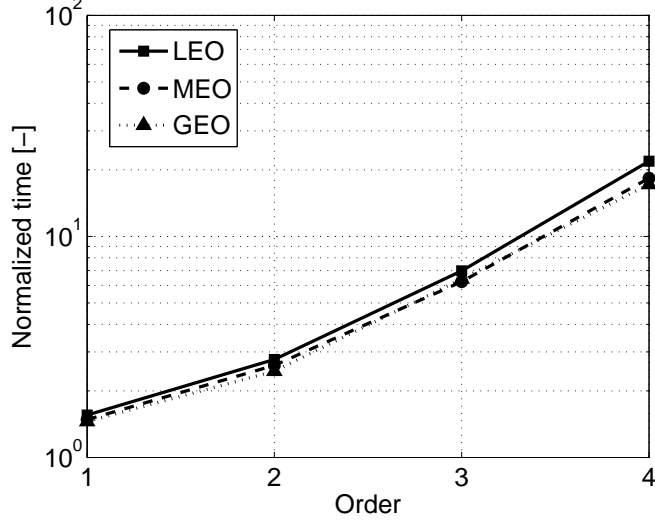
Figure 2: Normalized computational time for three classes of orbits

The accuracy of maps (42) with respect to uncertainties in the initial states (thus neglecting variation in the final time) is reported in Figure 3. The final positions are computed by evaluating the polynomial

$$[\boldsymbol{r}_f] = \boldsymbol{r}_f + \mathcal{M}_{\boldsymbol{r}_f}\left(\delta\boldsymbol{r}_0, \delta\boldsymbol{v}_0\right), \tag{50}$$

which is obtained by the DA-based integration in AIDA for a given time window up to 7 days. Then, they are compared with pointwise propagations of the same perturbed initial states. For each of the three orbital regimes 10^9 samples were generated considering the normalized full state covariance matrix obtained with the OD process described in Sect. 5. Since the accuracy of the map decreases with the distance of the sample from the reference point, the 100 samples with the largest displacement from the reference position and velocity were selected. This represents an accurate estimation of the worst cases that can be encountered if collision probability is computed using Monte Carlo based on the polynomial approximation of the final position. Realistically, at least one billion samples would be required to estimate a collision probability of 10^{-6} with a reasonable confidence level (Dagum et al., 2000).

The maximum differences between Taylor polynomials evaluations and pointwise integrations over a time window of 7 days are plotted in Figure 3(a) for different expansion orders. The error on the final position is

22

lower for the GEO orbit, due to the lower number of revolutions, and higher for the HEO. For the latter, the reason is mainly the higher values of the states covariances and the particular environment encountered by the objects during their motion. The influence of perturbations changes along the orbit, with an increasing relative importance of atmospheric drag near the pericenter and of third body gravity and solar radiation pressure at the apogee. In addition, the area to mass ratio is the highest among the considered orbits. Note that in all cases the maximum error on position after 7 days is approximately 1 meter for the HEO, less than 0.1 m for the LEO, and 1 mm for the GEO.

To provide more insight on the expansion error, Figure 3(b) illustrates the maximum differences between the polynomial evaluations and pointwise integration of the final position for different propagation time using an expansion order $k = 3$.
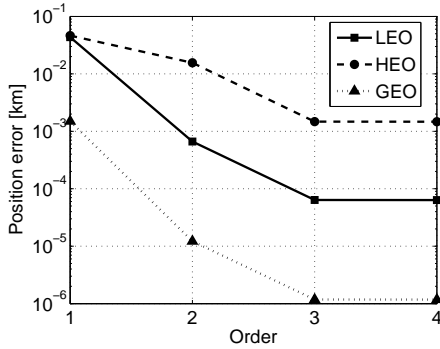
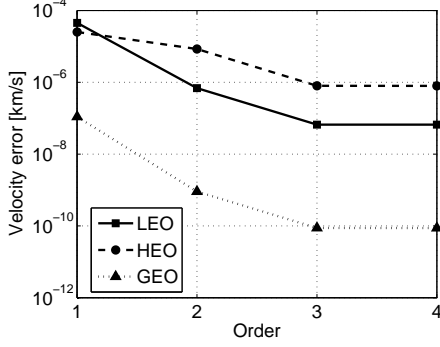(a) Final position error versus DA expansion order
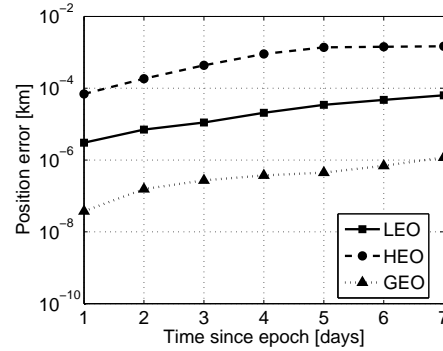
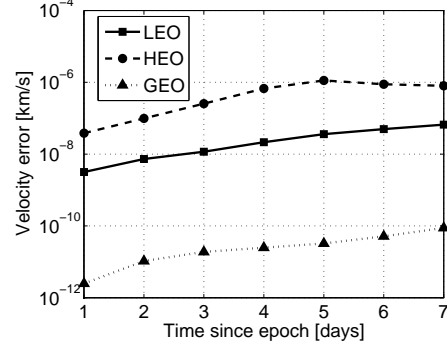(b) Final position error versus propagation time using order $k = 3$

Figure 3: Accuracy of the DA map of final position $\boldsymbol{r}_f(\boldsymbol{r}_0, \boldsymbol{v}_0)$ obtained with AIDA against DA expansion order and propagation time

The results of the same analysis focused on the final velocity are plotted in Figure 4(a) and Figure 4(b). The figures confirm the behaviour found for the final position. The maximum error is found again for HEO and is approximately 1 mm/s.

According to these results, a third order expansion is a good compromise between computational time and expansion accuracy for the selected test cases.

(a) Final velocity error versus DA expansion order

(b) Final velocity error versus propagation time

Figure 4: Accuracy of the DA map of final velocity $\boldsymbol{v}_f(\boldsymbol{r}_0, \boldsymbol{v}_0)$ obtained with AIDA against DA expansion order and propagation time

In the following analysis, the effects of orbital perturbations on the final position are studied to assess

1. the effect of the perturbations in the considered cases;
2. how well these effects are captured by the Taylor expansion.

For point 1) the final position obtained by considering the gravitational model of degree and order 10, atmospheric drag, solar radiation pressure with dual-cone shadow model, and third body perturbation from Sun and Moon is used as reference. Starting from a Keplerian orbital model (labeled as K in the figure), perturbations are gradually added and the displacement between the resulting final position and the reference value are computed. The results for a 7-day propagation are represented by the black curves in Figures 5(a), 5(b), 5(c). As before, the error curves are obtained considering the 100 samples with the largest displacement from the reference initial state selected from a set of 10^9 samples.

For the LEO regime, the largest effect is due to the gravitational model. The reason is that the orbital perigee is high enough to limit the effect of drag, whereas the high inclination result in a large effect of Earth oblateness. For the HEO regime no significant improvements are obtained by adding solar radiation pressure or atmospheric drag to Keplerian model. Nevertheless, the combined effects of gravitational harmonics, solar radiation pressure, and drag reduce considerably the displacement with the reference position.
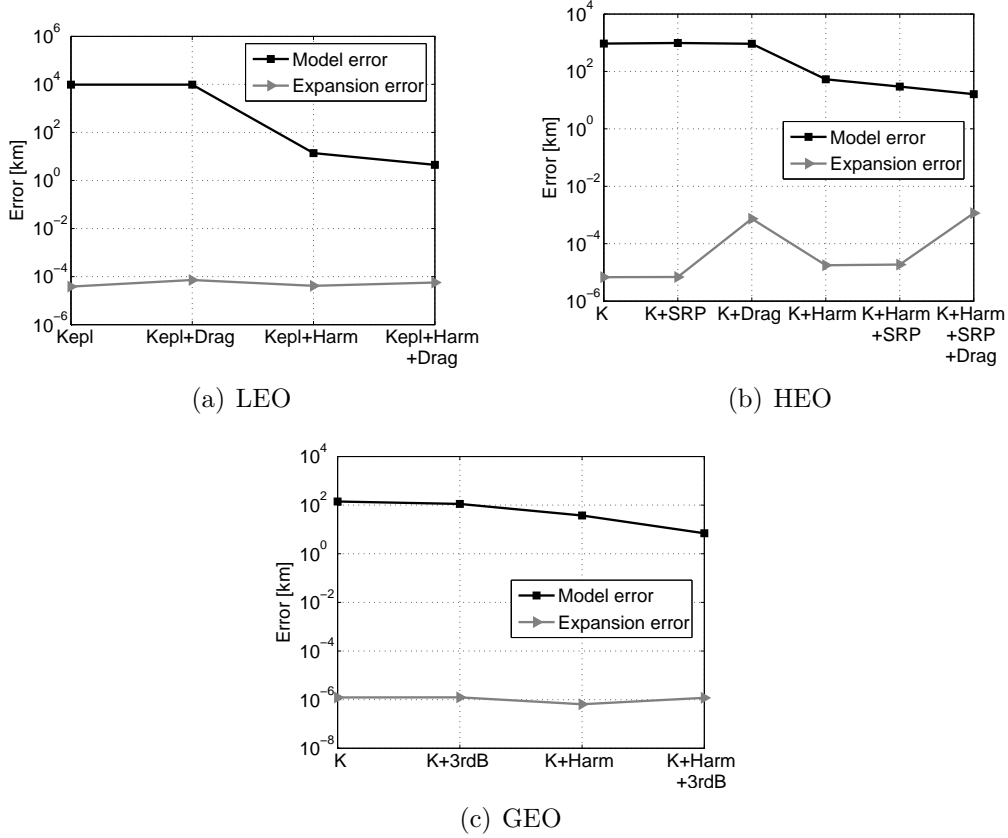
(a) LEO

(b) HEO

(c) GEO

Figure 5: Analysis of model and DA expansion error with different perturbations for the three orbital regimes for a 7-day propagation.

Even in the GEO regime, the combined effect of third body and gravitational harmonics is necessary to obtain a displacement on the final position of less than 10 km.

The accuracy of the DA expansions for the different orbital regimes and dynamical models is also analyzed in Figure 5 to highlight the contribution of the different sources of perturbation in the expansion error budget. The analysis is again obtained by comparing pointwise integrations with map evaluations for 100 virtual debris selected with the procedure used throughout this section. The curves are obtained considering 7-days propagations with an expansion order of the flow $k = 3$. For the LEO and GEO regimes the error remains almost constant with the introduction of the different sources

of perturbation. In particular, the expansion error remains below 0.1 m for the LEO and is approximately 1 mm for the GEO, confirming the figures obtained previously in the position accuracy analysis with respect to order of the expansion and propagation time. For the HEO test case the largest contribution to final position error comes from atmospheric drag. The orbit is highly eccentric, with a perigee of 1340 km and an apogee of 5700 km. In addition, the area to mass ratio estimated by the OD process is 2.94 m^2/kg. After each revolution, the virtual debris will distribute along the trajectory. Due to the eccentricity of the orbit, debris can experience different air densities since they are found at different altitudes. This behaviour has to be captured by the DA expansion of the density, which is a function of altitude that in turn is a function of the current satellite position. As a result, the accuracy of the air drag perturbation is lower due to the large range of altitude that is captured by the Taylor expansion of the satellite position at each integration step within the atmosphere bound of 2,000 km. Nevertheless, the expansion error of the HEO remains always below 1 m after 7 days of propagation.

5.2. Nominal TCA identification and time expansion accuracy

Three close approaches can be identified for the test case orbits listed in Table 1 using TLEs and SGP4/SDP4 propagations. The obtained TCA and DCA are reported in Table 3, together with the relative velocity Δv at the close approach. It can be observed that the relative velocity at the close encounter varies from more than 12 km/s for LEO to a few hundred m/s for the geosynchronous case. As a result, the close approaches can last from few seconds up to tens of seconds or even minutes.

Figure 6 illustrates the relative distance as a function of time in the neighbourhood of TCA for the three close approaches. The dashed line refers to the relative distance between the two objects computed using TLEs and SGP4/SDP4 propagations; the grey dot is the identified minimum. The relative distance obtained with AIDA propagation is plotted on the same figure with a solid black curve. The TCA of Table 3 are used as first guesses for the algorithm described in Section 4, which uses the nominal positions and velocities listed in Appendix A as inputs. The resulting minima of the relative distances are reported as grey dots on the solid black curves in Figure 6. The threshold value used to sieve potentially dangerous approaches, here set to 7.5 km, is indicated by the dashed-dotted lines.

Table 3: Close approaches: first guesses with SGP4/SDP4

Test case	Sat. No.	Orbit	TCA [days]	TCA	DCA [km]	Δv [km/s]
A	11510	LEO	3.538	2013 Jan 26 17:45:19	1.149	12.472
	21574	LEO	3.805			
B	20237	LEO	4.125	2013 Mar 07 06:34:23	6.064	8.553
	23820	HEO	2.403			
C	36744	GEO	0.240	2011 Feb 07 23:14:17	1.676	0.283
	28194	GEO	0.431			

As can be seen, the algorithm correctly identifies the minima of the relative distances computed with AIDA propagations. The differences between the two curves are due to the more accurate dynamical model implemented in AIDA and to the procedure adopted to estimate the initial conditions from TLEs. In particular, the error increases with the distance from the time windows on which the OD process is performed. The largest difference in TCA and DCA is obtained for test case B, probably due to slightly differences in the two orbital periods that caused a phase shift at the TCA with respect to SGP4/SDP4 propagation. For what concerns the GEO test cases, the TCA and DCA are the closest to the SGP4/SDP4 ones since the close conjunction occurs within the time window used for the OD process, i.e. less than 0.5 days from the reference epoch of both objects.

For the sake of completeness, the numerical values of TCA and DCA obtained with AIDA are reported in Table 4.

The analysis of the accuracy of the time expansions is performed on the square root of Map (43) evaluated in $-c_0 + \delta t$, and considering nominal initial conditions, i.e.

$$[d] = d + \mathcal{M}_d(\delta t). \tag{51}$$

The accuracy of Map (51) in a time window of 8 minutes around t^* is illustrated for the three conjunctions in Figure 7, where the relative distances computed with the DA approach are compared with pointwise propagations for samples equally spaced in the time domain. A third order expansion is used, following the considerations carried out in Section 5.1 about computational time and expansion accuracy.

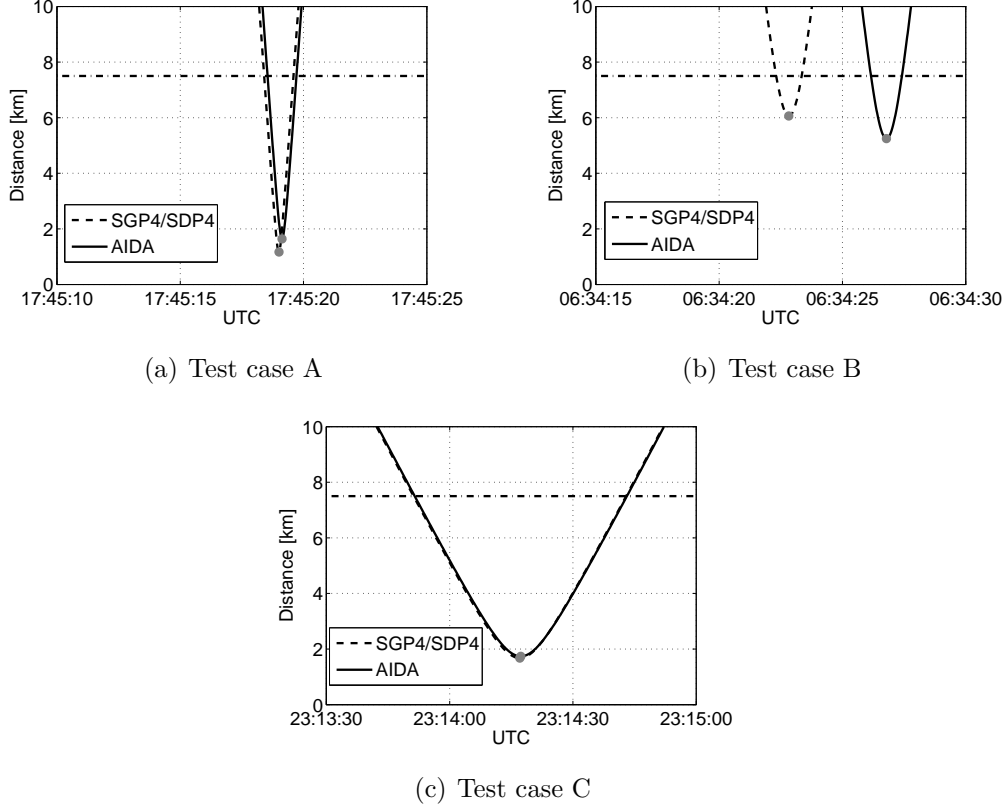(a) Test case A

(b) Test case B

(c) Test case C

Figure 6: Comparison of the relative distances obtained with AIDA and SGP4/SDP4 propagations. Grey dots are identified local minima and the dashed-dotted line is the 7.5 km conjunction threshold.

The error reaches its minimum at the TCA and increases with the displacement from TCA. The maximum errors on the DCA are at the boundary of the considered time intervals, and are 108 m for test case A, 294 m for test case B, and 4×10^{-3} m for test case C. The computational costs for each simulation are listed in Table 5. Most time is needed in the propagation of the two orbits, whereas the time required for the identification of the TCA is only a few milliseconds and is independent of the propagation period covered for each orbit. Note that the computational time for orbit propagation of the two GEO orbits is really low due to the short time interval between TCA and reference epoch.

28

Table 4: Close approaches: TCA, DCA, and relative velocities obtained with AIDA and TCA expansion algorithm

Test case	Sat. No.	Orbit	TCA	DCA [km]	Δv [km/s]
A	11510	LEO	2013 Jan 26 17:45:19	1.641	12.471
	21574	LEO			
B	20237	LEO	2013 Mar 07 06:34:27	5.248	8.553
	23820	HEO			
C	36744	GEO	2011 Feb 07 23:14:17	1.739	0.283
	28194	GEO			

Table 5: Computational time of TCA identification algorithm. In this case propagation with AIDA is performed only with one DA variable, time t_f.

Test case	1st orbit prop.	2nd orbit prop.	TCA expansion	Total time	
A	31.44	34.11	8.83E-3	65.56	s
B	33.05	12.85	8.51E-3	45.91	s
C	0.34	0.51	8.62E-3	0.85	s

5.3. TCA and DCA expansion accuracy

As already pointed out in Sect. 5.1, each virtual debris is characterized by a different trajectory evolution. Thus, each pair of virtual debris has its own value of TCA and DCA. If a pointwise method is used, the integration of two orbits and the use of a root-finding method would be required for each pair of virtual debris. The DA-based method introduced in this work allows replacing multiple propagation with faster evaluations of Taylor polynomials, after a single DA propagation of the two objects is carried out with AIDA. These maps can be used provided that they are sufficiently accurate for the aimed task: i.e., the identification of conjunctions and the computation of collision probabilities. As an example, when the computation of collision probabilities is of interest, the error on DCA expansion should be at least lower than the threshold adopted for the classification of a close conjunction as potentially dangerous. The accuracy analysis of these maps is the focus of this section.
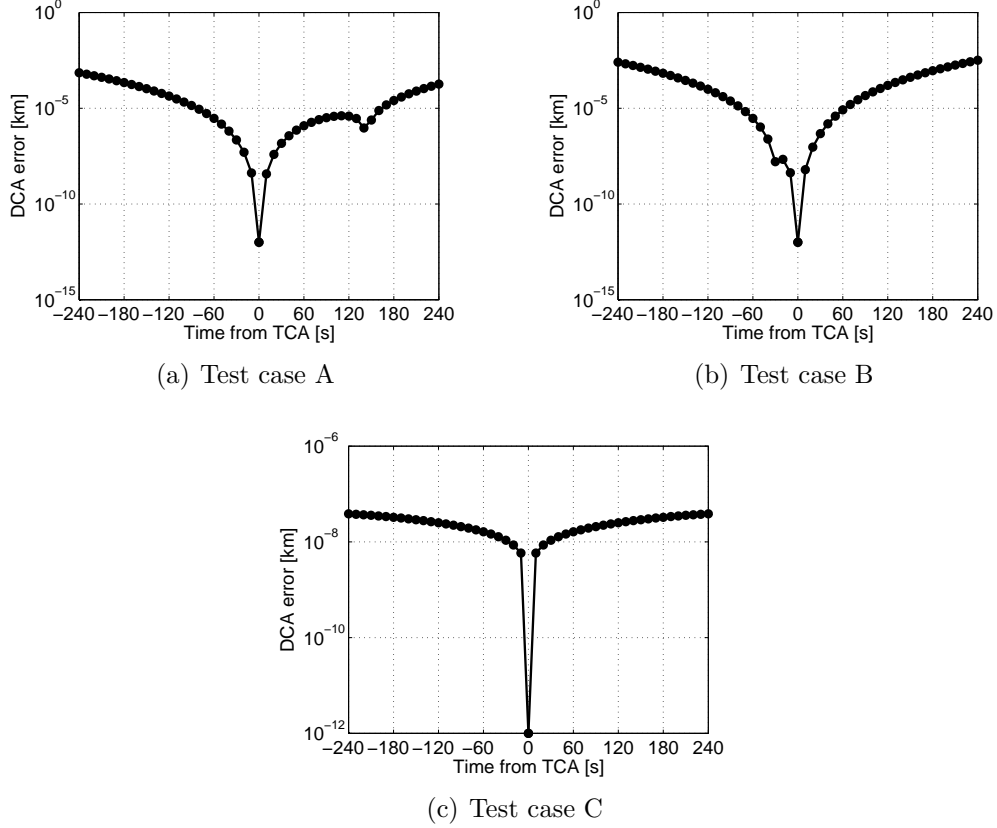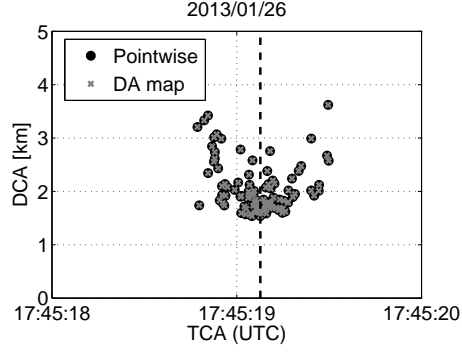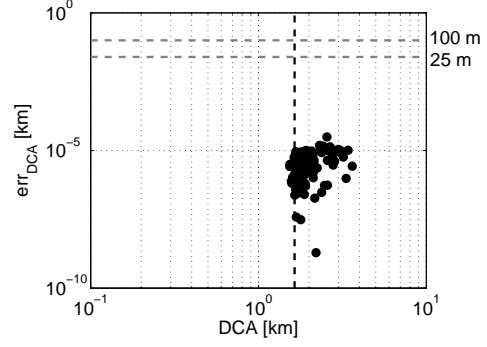
(a) Test case A

(b) Test case B

(c) Test case C

Figure 7: Comparison between pointwise numerical propagations and polynomial expansions for the relative distance d in the proximity of TCA

As the expansion errors get bigger for larger uncertainties, the same method used for the selection of virtual debris in Sect. 5.1 is here used. Thus, 10^9 samples are generated and those with the largest displacement from the nominal initial position and velocity of the two objects are selected. Given a couple of virtual debris, the TCA and DCA computed with the evaluation of the Taylor polynomials (48) and (49) are compared with those obtained with AIDA pointwise integrations (i.e., two integrations and some root-finder iterations). The comparison of the times and associated distances of close approach for the three conjunctions are given in Figures 8(a), 8(c), and 8(e).
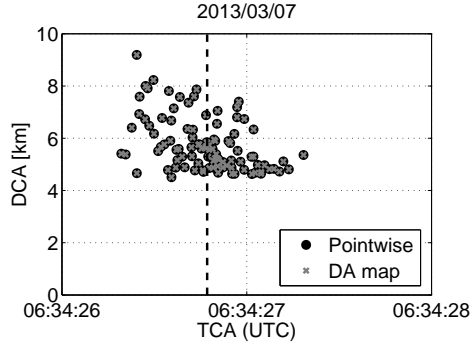
The black dots are the TCAs and DCAs obtained with the pointwise numerical propagation of the set of virtual debris, whereas grey crosses are
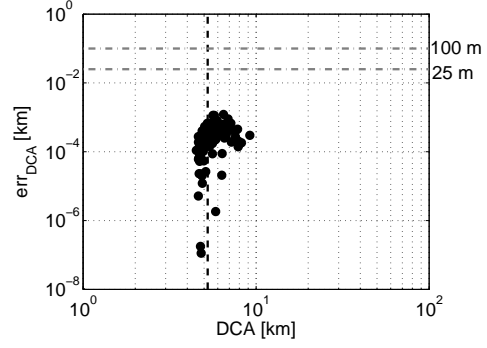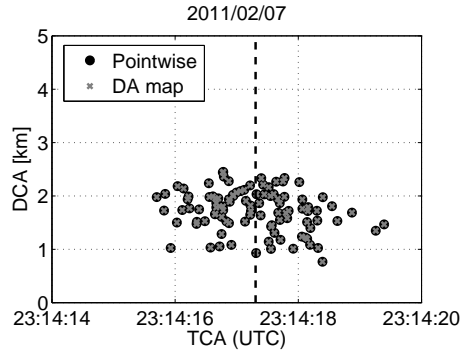
(a) Test case A: TCA vs. DCA
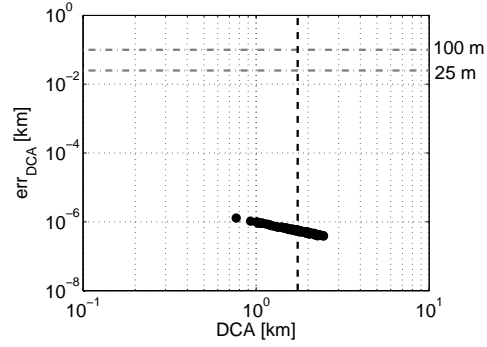
(b) Test case A: DCA error

(c) Test case B: TCA vs. DCA

(d) Test case B: DCA error

(e) Test case C: TCA vs. DCA

(f) Test case C: DCA error

Figure 8: Comparison between pointwise numerical propagations and polynomial expansions for TCA and DCA with uncertain initial states

the values obtained with DA maps evaluations. The dashed line locates the nominal TCA value. In all cases the polynomial evaluation matches the numerical one. The difference between the polynomial evaluation of the perturbed TCA, t^*_{DA}, and the pointwise one, t^*_{PW}, is always lower than 10^{-4} seconds making them indistinguishable in Figure 8.

The error on the DCAs is defined as

$$err_{d*} = |d^*_{PW} - d^*_{DA}|, \tag{52}$$

where d^*_{PW} is the numerical DCA obtained with pointwise integrations, and d^*_{DA} is the one obtained with the evaluation of the Taylor expansions. This error is plotted versus the numerical DCA in Figures 8(b), 8(d), and 8(f). The data are compared with two thresholds, 100 m and 25 m. These two values represent estimates of the combined hard body radius that can be used for collision probability computation. The nominal DCA is indicated with a vertical dashed line.

In all cases the error on the DCA is well below the 25 m threshold, so the accuracy of the methods is suitable for a collision probability computation using a Monte Carlo method. The accuracy of the DA map can be related to the accuracy of the final position map with respect to propagation time analyzed in Sect. 5.1 and summarized in Figure 3(b). The maximum error for test case A is around 10^{-5} km and the TCA is 3.5 days after the reference epoch of satellite 1 and 3.8 days after the reference epoch of satellite 2. This is the same error of the final position map of the LEO orbit after 4 days of propagations. Similar considerations can be made for test case 2, where the TCA is 4.1 days after the reference epoch of satellite 3 and 2.4 days after the reference epoch of satellite 4, that is in HEO. The maximum error for these propagation times, according to Figure 3(b), is obtained for the HEO orbit and is nearly 10^{-3} km, which is compatible with the error obtained for the DCA expansion. The lower error is found for the GEO, also due to the short propagation time.

The computational times required for computing Map (48) and (49) are listed in Table 6. The higher number of variables results in higher computational cost with respect to the single variable DA propagations of Table 5. Note that these performances could be improved by running the propagation of the two orbits in parallel, reducing in this way the computational time to the one required by the most demanding orbit. In addition, in case of multiple conjunctions, the propagation of the two objects can be performed

only once, whereas the TCA and DCA expansion algorithm is run on each conjunction.

Table 6: Computational time of TCA and DCA expansion algorithm. Each propagation with AIDA involves in this case 7 DA variables, i.e. time and initial states, since the aim is to obtain the polynomial approximation of $d^* = \mathcal{M}(\delta\boldsymbol{x}_0^1, \delta\boldsymbol{x}_0^2)$

Test case	1st orbit prop.	2nd orbit prop.	TCA&DCA expansion	Total time	
A	265.76	313.80	4.85E-2	579.60	s
B	262.44	101.53	3.99E-2	364.02	s
C	1.86	3.13	3.86E-2	5.03	s

Note that the evaluation of the d^* map takes 1.73×10^{-5} s on average (this number is obtained by evaluating the map for one million samples using the Horner scheme implemented in COSY-Infinity). If one considers that the reliable computation of collision probability requires a large set of virtual debris to be evaluated, it is apparent that the proposed approach allows large time savings with respect to pointwise propagations, with limited loss of accuracy.

6. Conclusion

A method for the analysis of close conjunctions between objects in Earth orbit has been presented. Differential algebraic (DA) techniques have been exploited to calculate the Taylor expansion of the time and distance of closest approach (TCA and DCA) with respect to initial orbital uncertainties. This reduces the problem of computing the TCA and DCA for the entire set of virtual debris to 1) two DA-based numerical propagations 2) polynomial manipulations to get maps of the time t^* and distance d^* of closest approach as function of uncertain initial states 3) multiple fast evaluations of these maps.

The numerical propagations are performed with AIDA (Accurate Integrator for Debris Analysis), a high-fidelity propagator (which includes EGM2008 gravity model, NRLMSISE-00 atmosphere, Sun and Moon perturbations based on JPL DE405 ephemerides, and solar radiation pressure) written in the language COSY-Infinity. This allows us to compute, along with the trajectory, the high order expansion of the flow with respect to initial conditions

in a limited amount of time. The resulting expansions are then used to build the map of the squared distance function. Partial inversion techniques and polynomial compositions are exploited to calculate the nominal values of TCA and DCA as well as their Taylor expansion with respect to uncertainties in the initial states. Thus, for each pair of virtual debris the computation of TCA and DCA reduces to polynomial evaluations. As a result, a drastic reduction in computational cost is achieved compared to classical pointwise methods: this paves the way to the development of efficient algorithms for the computation of collision probabilities.

The accuracy of the Taylor polynomials is suitable for the computation of a collision probability. An analysis is performed considering the 100 samples with the largest displacement from the nominal initial conditions among a set of 10^9 samples generated considering the full covariance matrix. The error between the polynomial evaluations of the DCA maps and a pointwise propagation are always less than 1 m for the considered orbits. It was also shown that the accuracy of the map of d^* is related to the accuracy of the maps of the final position of the objects. With order $k = 3$ and propagation window of 7 days, the error on the final position is less than 1 m for the orbits that are analyzed in this paper.

Further analysis will be performed to test the proposed approach on different orbital regimes and for different conjunction geometries. Future research will also take into account computationally efficient methods for the estimation of small collision probabilities. The reason is that the collision probability between Earth orbiting objects is usually of the order 10^{-4} to 10^{-6} and a high number of samples, and in turn polynomial evaluations, are required in those cases.

7. Acknowledgment

References

Aida, S., Kirschner, M., Collision risk assessment and operational experiences for LEO satellites at GSCOC. J. Aerospace Eng. 4 (2), 121, 2012.

Alarcon-Rodriguez, J., Martinez-Fadrique, F., Klinkrad, H., Development of a collision risk assessment tool. Adv. Space Res. 34 (5), 1120–1124, 2004.

Alfano, S., Relating position uncertainty to maximum conjunction probability. J. Astronaut. Sci. 53 (2), 193–205, 2005.

Armellin, R., Di Lizia, P., Bernelli-Zazzera, F., Berz, M., Asteroid close encounters characterization using differential algebra: the case of Apophis. Celest. Mech. Dyn. Astr. 107 (4), 451–470, 2010.

Armellin, R., Morselli, A., Di Lizia, P., Lavagna, M., Rigorous computation of orbital conjunctions. Adv. Space Res. 50 (5), 527 – 538, 2012.

Berz, M., The new method of TPSA algebra for the description of beam dynamics to high orders. Los Alamos National Laboratory, Technical Report AT-6:ATN-86-16, 1986.

Berz, M., The method of power series tracking for the mathematical description of beam dynamics. Nucl. Instrum. Methods A258 (3), 431–436, 1987.

Berz, M., Differential Algebraic Techniques, Entry in Handbook of Accelerator Physics and Engineering. World Scientific, New York, 1999a.

Berz, M., Modern Map Methods in Particle Beam Physics. Academic Press, New York, 1999b.

Berz, M., Makino, K., COSY INFINITY Version 9 Reference Manual. Michigan State University, East Lansing, MI 48824, MSU Report MSUHEP-060803, 2006.

Coffey, S. L., Neal, H. L., Visel, C. L., Conolly, P., Demonstration of a Special-Perturbation-Based Catalog in the Naval Space Command System. Adv. Astronaut. Sci. 99, 227–248, Paper AAS 98-113, 1998.

Dagum, P., Karp, R., Luby, M., Ross, S., An Optimal Algorithm for Monte Carlo Estimation. SIAM Journal on Computing 29 (5), 1484–1496, March 2000.

Flohrer, T., Krag, H., Klinkrad, H., ESA's process for the identification and assessment of high-risk conjunction events. Adv. Space Res. 44 (3), 355–363, 2009a.

Flohrer, T., Krag, H., Klinkrad, H., Bastida Virgili, B., Früh, C., Improving ESA's Collision Risk Estimates by an Assessment of the TLE Orbit Errors of the US SSN Catalogue. Proc. of the 5th Europ. Conf. on Space Debris, Darmstadt, Germany ESA SP-672, 2009b.

Flohrer, T., Krag, H., Lemmens, S., Bastida Virgili, B., Merz, K., Klinkrad, H., Statistical Look on ESA's Conjunction Event Predictions. Proc. of the 6th Europ. Conf. on Space Debris, Darmstadt, Germany ESA SP-723, 2013.

Hoots, F. R., Schumacher, P. W., Glover, R. A., History of Analytical Orbit Modeling in the U.S. Space Surveillance System. J. Guid. Control Dynam. 27 (2), 174–185, 2004.

Kelso, T. S., Validation of SGP4 and IS-GPS-200D against GPS precision ephemerides. Proceedings of the 17th AAS/AIAA Space Flight Mechanics Conference Paper AAS 07-127, January 2007.

Kelso, T. S., Alfano, S., Satellite Orbital Conjunction Reports Assessing Threatening Encounters in Space (SOCRATES). Adv. Astronaut. Sci. 120 (I), 317–326, Paper AAS 05-124, 2005.

La Porte, F., Sasot, E., Operational management of collision risks for LEO satellites at CNES. Proceedings of the Space-Ops 2008 Conference, Paper AIAA-2008-3409, 2008.

Makino, K., Berz, M., Verified global optimization with Taylor model-based range bounders. Trans. Comput. 4 (11), 1611–1618, 2005.

Montenbruck, O., Gill, E., Satellite Orbits - Models, Methods, and Applications. Springer-Verlag, 2000.

Pavlis, N. K., Holmes, S. A., Kenyon, S. C., Factor, J. K., The development and evaluation of the Earth Gravitational Model 2008 (EGM2008). J. Geophys. Res-Sol. Ea. 117 (B4), 2012.

Picone, J. M., Hedin, A. E., Drob, D. P., Aikin, A. C., NRLMSISE-00 empirical model of the atmosphere: Statistical comparisons and scientific issues. J. Geophys. Res-Space 107 (A12), SIA 15–1–SIA 15–16, 2002.

Standish, E. M., JPL Planetary and Lunar Ephemerides, DE405/LE405. JPL IOM 312, 1998.

Vallado, D., Fundamentals of Astrodynamics and Applications, 2nd Edition. Springer-Verlag, 205–231, 2001.

Vallado, D. A., Cefola, P. J., Two-Line element sets – Practice and Use. 63rd International Astronautical Congress, Naples, Italy, 2012.

Valli, M., Armellin, R., Lizia, P. D., Lavagna, M. R., Nonlinear mapping of uncertainties in celestial mechanics. Journal of Guidance, Control, and Dynamics 36 (1), 48–63, 2013.

Appendix A. Initial states

Object 1

```
#Satellite ID
11510
# Reference UT
23/01/2013 04:50:14.3891483545
# ECI J2000 mean position (km)          # ECI J2000 mean velocity (km/s)
-3.7721713215E+03                        +2.0902060722E-01
+4.9994447579E+03                        -4.1566598389E+00
+3.4596010085E+03                        +6.2007675483E+00
#Covariance matrix (km^2, km^2/s, km^2/s^2)
+3.3085138968E-03    +8.1569556907E-04    +6.3216213375E-04    +1.8594276250E-07    -6.8263129256E-07    +5.9476230591E-07
+8.1569556907E-04    +3.5565393379E-03    -2.0265346793E-03    -1.4711117793E-06    +2.4696267856E-06    +3.9897201273E-07
+6.3216213375E-04    -2.0265346793E-03    +5.2194974395E-03    +1.8751377639E-06    -2.0999712757E-06    -2.4422592051E-06
+1.8594276250E-07    -1.4711117793E-06    +1.8751377639E-06    +4.1301523076E-09    +1.5475454258E-10    +2.0530594462E-10
-6.8263129256E-07    +2.4696267856E-06    -2.0999712757E-06    +1.5475454258E-10    +3.3307135024E-09    +9.4333446860E-10
+5.9476230591E-07    +3.9897201273E-07    -2.4422592051E-06    +2.0530594462E-10    +9.4333446860E-10    +2.1401382330E-09
# Drag Area to mass ratio (m^2/kg), Cd = 2.2
+7.9693388204E-07
# SRP Area to mass ratio (m^2/kg), eps = 0.31
+7.2433066671E-02
# TLE used for nonlinear least square fit
1 11510U 79078A   13023.20155543  .00000103  00000-0  47518-4 0  1325
2 11510 074.0358 118.1525 0011496 088.5235 301.8268 14.32681184744812
```

Object 2

```
#Satellite ID
21574
# Reference UT
22/01/2013 22:26:54.0620023012
# ECI J2000 mean position (km)          # ECI J2000 mean velocity (km/s)
+3.0799289553E+03                        -6.7335462784E+00
-1.1774790818E+03                        +9.0909039658E-02
+6.3094366569E+03                        +3.2876594254E+00
#Covariance matrix (km^2, km^2/s, km^2/s^2)
+6.5110022394E-03    +1.1558082277E-04    -2.6158096782E-03    +2.7966467039E-06    -9.1277337669E-07    +4.6393449869E-06
+1.1558082277E-04    +3.4915733193E-03    +3.9873127221E-04    -2.0268510003E-07    +9.4972338534E-09    -4.9783742550E-09
-2.6158096782E-03    +3.9873127221E-04    +2.3878201022E-03    -2.8390289006E-07    +4.3116429054E-07    -2.7993620662E-06
+2.7966467039E-06    -2.0268510003E-07    -2.8390289006E-07    +1.9513723058E-09    -7.2082566955E-11    +1.6422454906E-09
-9.1277337669E-07    +9.4972338534E-09    +4.3116429054E-07    -7.2082566955E-11    +3.9810526042E-09    -2.2788964517E-10
+4.6393449869E-06    -4.9783742550E-09    -2.7993620662E-06    +1.6422454906E-09    -2.2788964517E-10    +4.5198732031E-09
# Drag Area to mass ratio (m^2/kg), Cd = 2.2
+3.1525096930E-02
# SRP Area to mass ratio (m^2/kg), eps = 0.31
+8.6998667752E-02
# TLE used for nonlinear least square fit
1 21574U 91500A   13022.93534794  .00000203  00000-0  83024-4 0  5448
2 21574 098.2460 355.3520 0034877 082.5971 341.2307 14.36304867126904
```

Object 3

```
#Satellite ID
20237
# Reference UT
03/03/2013 03:34:55.0660854578
# ECI J2000 mean position (km)          # ECI J2000 mean velocity (km/s)
-6.9608504967E+02                       -7.1908922523E-01
+3.9447413218E+03                       -6.1667222531E+00
+6.6676677691E+03                       +3.5653521805E+00
#Covariance matrix (km^2, km^2/s, km^2/s^2)
+2.6282128459E-03    +4.9609574850E-04    -1.2472570972E-04    -4.6583233145E-08    +1.7868359512E-07    +4.6499234699E-07
+4.9609574850E-04    +5.1294297368E-03    -2.8175167448E-03    -2.6604126144E-07    +2.0213010498E-06    +3.1959036434E-06
-1.2472570972E-04    -2.8175167448E-03    +2.5231867160E-03    +2.3789204150E-07    -6.2505918713E-07    -2.4279044407E-06
-4.6583233145E-08    -2.6604126144E-07    +2.3789204150E-07    +2.1743122769E-09    -1.5519486573E-10    +3.4526380891E-11
+1.7868359512E-07    +2.0213010498E-06    -6.2505918713E-07    -1.5519486573E-10    +1.1046525934E-09    +1.0043646178E-09
+4.6499234699E-07    +3.1959036434E-06    -2.4279044407E-06    +3.4526380891E-11    +1.0043646178E-09    +2.6911031636E-09
# Drag Area to mass ratio (m^2/kg), Cd = 2.2
+1.2645425516E-03
# SRP Area to mass ratio (m^2/kg), eps = 0.31
+1.2072974147E-03
# TLE used for nonlinear least square fit
1 20237U 89074F   13062.14924845  .00000051  00000-0  22376-3 0   924
2 20237 082.5723 087.7657 0004186 082.5944 337.2822 12.63395421 82164
```

Object 4

```
#Satellite ID
23820
# Reference UT
04/03/2013 20:54:42.1015834808
# ECI J2000 mean position (km)          # ECI J2000 mean velocity (km/s)
+8.7741400840E+03                       -1.4355103802E+00
+2.5174129938E+03                       -1.2478148630E-01
-1.1037130913E+01                       +6.7080731673E+00
#Covariance matrix (km^2, km^2/s, km^2/s^2)
+3.7365649111E-03    -1.8774896159E-03    -1.7756879090E-03    +1.1737797328E-06    +7.2965655117E-07    -1.9125787150E-06
-1.8774896159E-03    +8.4701814389E-03    +1.0368790267E-03    -3.6471225605E-07    -1.6589500283E-06    -4.7980730691E-07
-1.7756879090E-03    +1.0368790267E-03    +1.6075521088E-02    -7.5785218204E-06    -1.8780849958E-06    -6.5081630239E-07
+1.1737797328E-06    -3.6471225605E-07    -7.5785218204E-06    +4.4433848385E-09    -2.2730797064E-10    +2.1850803760E-10
+7.2965655117E-07    -1.6589500283E-06    -1.8780849958E-06    -2.2730797064E-10    +4.4757376092E-09    -1.2691331517E-10
-1.9125787150E-06    -4.7980730691E-07    -6.5081630239E-07    +2.1850803760E-10    -1.2691331517E-10    +1.4485851866E-09
# Drag Area to mass ratio (m^2/kg), Cd = 2.2
+2.9406164525E+00
# SRP Area to mass ratio (m^2/kg), eps = 0.31
+3.0085686833E+00
# TLE used for nonlinear least square fit
1 23820U 66077D   13063.87132062  .00020734  00000-0  60931+0 0  3081
2 23820 087.6151 016.1776 2204224 081.8477 302.4728 08.81977559657175
```

Object 5

```
#Satellite ID
36744
# Reference UT
07/02/2011 17:28:54.3233361840
# ECI J2000 mean position (km)          # ECI J2000 mean velocity (km/s)
-4.1218586022E+04                       -6.4868868413E-01
+8.8973993544E+03                       -3.0052824298E+00
+6.5767406271E+01                       +1.1524550572E-03
#Covariance matrix (km^2, km^2/s, km^2/s^2)
+4.8684260805E-03    +2.8144019728E-03    +2.5855660862E-04    -2.9284545423E-07    -2.6416134160E-07    +4.7498191816E-10
+2.8144019728E-03    +2.1151760568E-02    -3.7731926146E-04    -8.9572939141E-07    +3.3550231926E-07    +1.4314764844E-09
+2.5855660862E-04    -3.7731926146E-04    +1.2383334290E-02    -1.5150684923E-08    -2.4339739820E-08    +1.8043084888E-11
-2.9284545423E-07    -8.9572939141E-07    -1.5150684923E-08    +5.3865847597E-11    -3.4214779531E-12    +1.8558767562E-14
-2.6416134160E-07    +3.3550231926E-07    -2.4339739820E-08    -3.4214779531E-12    +2.8061830734E-11    +7.0815634448E-15
+4.7498191816E-10    +1.4314764844E-09    +1.8043084888E-11    +1.8558767562E-14    +7.0815634448E-15    +6.5166338686E-11
# SRP Area to mass ratio (m^2/kg), eps = 0.31
+4.6465256622E-12
# TLE used for nonlinear least square fit
1 36744U 10032A   11038.72840652 -.00000340  00000-0  10000-3 0   956
2 36744 000.0086 088.0631 0000788 256.3015 183.5741 01.00269939  2416
```

Object 6

```
#Satellite ID
28194
# Reference UT
07/02/2011 12:53:3.5177153349
# ECI J2000 mean position (km)          # ECI J2000 mean velocity (km/s)
-6.2997743088E+03                        -3.0273029590E+00
+4.1664952743E+04                        -4.6758259629E-01
+1.6017472390E+03                        +2.6224219851E-01
#Covariance matrix (km^2, km^2/s, km^2/s^2)
+2.4642428245E-02    +2.2712137544E-03    -6.4227590670E-04    -2.7598724396E-07    +1.0326836589E-06    +5.0508709424E-08
+2.2712137544E-03    +4.7804014798E-03    -5.3528868674E-04    +2.9818134924E-07    +2.3812269421E-07    -2.1062388599E-08
-6.4227590670E-04    -5.3528868674E-04    +1.3920101199E-02    +2.1358765258E-08    -9.7456008690E-08    -4.2449078518E-09
-2.7598724396E-07    +2.9818134924E-07    +2.1358765258E-08    +2.8385263365E-11    -3.0436486056E-12    +4.0211735020E-12
+1.0326836589E-06    +2.3812269421E-07    -9.7456008690E-08    -3.0436486056E-12    +6.1556025538E-11    -2.2819832001E-14
+5.0508709424E-08    -2.1062388599E-08    -4.2449078518E-09    +4.0211735020E-12    -2.2819832001E-14    +7.3295880653E-11
# SRP Area to mass ratio (m^2/kg), eps = 0.31
+4.0165085827E-02
# TLE used for nonlinear least square fit
1 28194U 04010A   11038.53684627 -.00000341 +00000-0 +10000-3 0 03215
2 28194 005.2813 074.4923 0001212 268.2646 116.0489 01.00265873025169
```