Collision risk prediction for constellation operators

R. Lucken\textsuperscript{a,b,*}, D. Giolito\textsuperscript{a}

\textsuperscript{a}Share My Space, 17 rue Burnouf 75019 Paris, France
\textsuperscript{b}Laboratoire de Physique des Plasmas (LPP), UMR CNRS 764, Ecole Polytechnique, Route de Saclay 91128 Palaiseau, France

Abstract

INDEMN is an object-oriented Python program dedicated to the modeling of the evolution of the densities of space objects. Following the work achieved by G. L. Somma (IAC 2016), the dynamical model is based on a source and sink approach for various altitudes. The source terms represent the future launches, the explosion of intact spacecrafts, and the collision between objects. Different collision cross sections are used for the various types of objects and the number of debris generated is based on the NASA break-up model. The sink terms are the drag and the end-of-life de-orbitation for the satellites launched after 2009, with a controllable success rate. The code was validated using the benchmark released by the Inter-Agency Space Debris Coordination Committee on the Stability of the Future LEO environment (IADC-12-08, Rev. 1, 2013). In addition to the classical object types featured in several statistical codes, which are intact objects, explosion debris, and collision debris, a new type representing the satellites of a specific constellation is included. These satellites orbit with altitudes close to 1 200 km and they can perform collision avoidance maneuvers as long as they are fully operational. It is shown that if only one primary collision occurs, the risk of a collision involving a constellation satellite becomes larger than 5% by 2035, which highly jeopardizes the satellite constellation as a whole.

Keywords: LEO environment modeling, Source-sink debris model, Constellation management

1. Introduction

The Low Earth Orbit (LEO) economy is driven by the growing market of global and real-time connectivity, and new space activities represent the major share of the growth of the overall space sector today \cite{1}. The emerging needs are addressed by the deployment of large constellations of small satellites, as announced by several telecommunication companies. This involves new platform designs that could also be used for Earth surveillance and business intelligence applications. On the other hand, Cubesat technologies, that allow in-situ data collection, for instance for science and climate change monitoring, and Earth observation services, contribute significantly to the increase of the number of LEO objects. Nevertheless, they are not considered a major threat for the long-term stability of the space environment due to their short orbital lifetime.

Multiple numerical studies have investigated the risk caused by constellations both on the stability of the environment, and the long-term profitability of large constellations. In particular, some of them addressed the relevance of the 25-year rule \cite{2,3}. The tools developed by national and international agencies were primarily made to assess how much a given satellite was in danger on a given orbital path, and only recent improvements accurately account for small satellite launches \cite{4,5,6}. Due to the large number of satellite constellations that have been unveiled in the last couple of years, knowing the effect of the other satellites of the same constellation or the...
consequences of the deployment of another constellation by a competitor on the orbital environment becomes crucial. Constellation-level parameters such as the reliability of the deorbitation system, altitude spread, or deorbitation time can be set only if all the constellation satellites are included together in the model evolution as a specific population. Finally, the number of LEO objects is several orders of magnitude higher than above 2 000 km, with many different intersecting trajectories, which makes a statistical description more relevant.

The INDEMN code was developed internally at Share My Space in Paris and uses a statistical model that derives the evolution of various populations at multiple altitudes. The model was initially inspired by the work of G.-L. Somma [7, 8, 9], improved by a elaborated object-oriented code architecture (the reader can refer to Appendix B for some details about the code architecture), that allows for an arbitrary number of different object populations and a more accurate description of the size distribution function of each population. The code is essentially one-dimensional (1D) and computes the time evolution of the object densities as a function of the altitude, for each population.

Section 2 provides a general overview of the model. Section 3 shows how this approach can be used to manage risk for a satellite constellation. Finally, some practical analysis are presented in Section 4.

2. A statistical model for the low Earth orbit environment

2.1. Overview

INDEMN is an object-oriented Python 3.X code developed by Share My Space. It features a statistical model deriving the evolution of various populations in various altitude shells and it can be used both for scientific and commercial purposes. The model is based on source and sink terms that include future launches, explosions, collisions, drag, and de-orbitation policies and compliance. Many different populations can be implemented and their interactions are computed at each time step. The collision frequency between two given populations depends on the local density of each population, at a given time. After being initialized using object databases for the larger objects (> 10 cm) and statistical models for smaller ones (< 10 cm) according to the NASA breakup model [10], the density of objects belonging to each population is computed dynamically with its radial dependence. The overall operating principle of INDEMN is illustrated in Fig. 1.

The main part of the code consists in the computation of the evolution of the density of various object populations depending on their altitude only. Choice was made to limit the boundaries of the LEO environment from 200 to 2 000 km. Below 200 km, the lifetime of an object is typically smaller than the time scale for which the models used in INDEMN are relevant (6 months minimum). Above 2 000 km, the drag compensation model does not apply anymore and the very low object density limits the relevance of any statistical approach. The whole model relies on the somewhat strong assumption that the orbital plane angles follow a chaotic law on a long time scale. This is justified by the fact that the drifts of the orbital planes depend on external perturbation terms that are very difficult to predict accurately enough for long-term orbit propagation, and on object parameters such as the attitude or the magnetic moment, which are impossible to predict or control in most cases for passive objects. For active satellites that feature a propulsion system, it is assumed that all collisions are avoided through successful maneuvers.
INDEMN cannot be used for just-in-time collision avoidance. It aims at providing a mission risk profile, for missions lasting more than six months. It is particularly relevant when the user aims at assessing the correlated risk related to the launch of multiple satellites (at various dates) on orbital trajectories that may intersect. It can be used to predict how many collision avoidance maneuvers will be required every year for a given spacecraft (SC), which has a significant impact on the operations [11]. Finally, a simplified interface was made available online since February 2018 at https://indemn.herokuapp.com. The access to the INDEMN online calculator is free and can be provided on demand by the authors.

2.2. Inputs of the model
2.2.1. Space objects catalogs
The main input of the code consists of the data of the space objects orbiting around Earth in the LEO environment. The main catalog is maintained by the US STRATCOM and made available on the space-track.org platform [12]. This data is mainly stored in the two-line-element (TLE) format and can be retrieved automatically by an API request. The information for each object is stored in a python data structure whose primary characteristics are the orbital elements directly initialized from the TLE data, and that includes various methods for secondary orbital properties, and for orbit propagation. However, orbit propagation of the objects is not used in the core of the code. A flag is added to the object definition that specifies the type of object (collision/explosion debris, rocket body, intact object, mission related object). All these objects are listed together and will be used in the initialization of the low earth (LEO) environment.

The collision and explosion object populations initialized with the object databases represent exclusively objects larger than 10 cm, due to the performance of the RADARs currently used to maintain the public catalogue [13]. However, these populations can be statistically extrapolated to smaller objects, using the NASA breakup model [10].

2.2.2. Atmosphere description
The residual atmospheric density is extrapolated from the NRLMSISE-00 Atmosphere model [14] that provides angle and altitude resolved density data since 1960. The data was averaged over the orbital angles and fitted to a sinusoidal function. This fit tends to minimize the amplitude of the temporal variations of density compared to other models such as the ones implemented in ESAs Master or NASAs Ordem.

The effect of the drag varies considerably depending on the altitude. The residual atmosphere mass density was fitted to:

\[
\rho(t, h) = \rho_m(h) + \rho_A(h) \cos \left( \frac{2\pi}{T_s} (t - t_0) \right)
\]

where \(T_s\) is the period of solar activity (11.2 years), and \(t_0\) is a date of maximum solar activity (January 1st, 2002). The parameters \(\rho_m(h)\) and \(\rho_A(h)\) where fitted using the NRLMSISE-00 atmosphere model. \(\rho_m\) is simply the time-averaged density and \(\rho_A\) is the standard deviation divided by \(\sqrt{2}\). A comparison between the historical atmosphere density data and the fitted approximation used here is presented in Fig. 2.
2.2.3. General parameters

The input parameters are sorted in Table 1 according to the level of expertise of the user. The mission time, the ballistic coefficient, and the explosion rate can be defined for each population independently. The date of the catalogue $t_{cat}$ corresponds to the date at which the data has been downloaded from the NORAD database. The relative velocity between two colliding objects $v_{rel}$ is assumed constant, which is a quite strong assumption that was however historically accepted [15]. The relative velocity primarily depends on the orbital plane angles. However, since the model depends only on the altitude, this influence is averaged out. The relative velocity should also depend on the altitude but the difference is not critical when only the LEO environment is treated. The reader can refer to a paper by Dolado-Perez et al. [16] for further details on this matter. All the expert parameters except the fraction of catastrophic collisions come from the NASA break-up model. These parameters are experimental fits and should not be changed, except if new experiments are carried out, or an uncertainty quantification study is needed.

The parameters listed in Table 1 deal exclusively with the background environment. More parameters describing the behavior of the constellation satellites will be introduced in Subsection 3.3.

2.3. Object populations

INDEMN was designed to run with an arbitrarily large number of populations. All the populations are stored in dictionary data structures indexed by their names. All the objects of a same population have the same ballistic coefficient $C_x$. All the populations are assumed to have the same mass density $\rho_m$ in the current version. The populations are sorted in four categories:

1. **Intact objects** populations are typically old satellites left in orbit at the end of their orbital lifetime or rocket bodies. Intact objects of the same population all have the same given size. The cross section used to compute the drag and the collision probability is calculated from the size of the object assuming a spherical geometry. They are subject to explosion with a given annual probability, that is greater than zero in general. During their de-orbitation phase, constellation satellites are treated as intact objects.

<table>
<thead>
<tr>
<th>Basic Parameter</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mission time</td>
<td>8 years</td>
</tr>
<tr>
<td>Deorbitation time</td>
<td>25 years</td>
</tr>
<tr>
<td>Compliance level to the space law, $C$</td>
<td>90%</td>
</tr>
<tr>
<td>Explosion rate, $p_e$</td>
<td>0/year</td>
</tr>
<tr>
<td>Balistic coefficient, $C_x$</td>
<td>2.2</td>
</tr>
<tr>
<td>Minimum altitude, $h_{min}$</td>
<td>400 km</td>
</tr>
<tr>
<td>Maximum altitude, $h_{max}$</td>
<td>2 000 km</td>
</tr>
<tr>
<td>Simulation start, $t_0$</td>
<td>08/09/2018</td>
</tr>
<tr>
<td>Physical duration of the run, $\Delta t$</td>
<td>80 years</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Advanced Parameter</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date of the catalogue, $t_{cat}$</td>
<td>08/09/2018</td>
</tr>
<tr>
<td>Typical radius of intact objects, $r_i$</td>
<td>2.1 m</td>
</tr>
<tr>
<td>Typical radius of explosion debris, $r_e$</td>
<td>0.3 m</td>
</tr>
<tr>
<td>Typical radius of collision debris, $r_c$</td>
<td>0.3 m</td>
</tr>
<tr>
<td>Object mass density, $\rho_m$</td>
<td>1.49 t/m$^3$</td>
</tr>
<tr>
<td>Mean number of objects created per launch, $n_L$</td>
<td>4.0</td>
</tr>
<tr>
<td>Reference velocity between colliding objects, $v_{ref}$</td>
<td>10 km/s</td>
</tr>
<tr>
<td>Number of meshes, $N_{shell}$</td>
<td>160</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Expert Parameter</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size exponent of the distribution function for collision debris</td>
<td>1.71</td>
</tr>
<tr>
<td>Mass exponent of the distribution function for collision debris</td>
<td>0.75</td>
</tr>
<tr>
<td>Pre-factor of the collision debris distribution bientt, function</td>
<td>0.03</td>
</tr>
<tr>
<td>Size exponent of the distribution function for explosion debris</td>
<td>1.6</td>
</tr>
<tr>
<td>Pre-factor of the explosion debris distribution function</td>
<td>6.0</td>
</tr>
<tr>
<td>Fraction of catastrophic collisions</td>
<td>0.3187</td>
</tr>
</tbody>
</table>

Table 1: Table of the input parameters of INDEMN
2. **Constellation objects** are object populations that are under control. They have the same properties as the populations of intact objects but are provided with a de-orbitation scenario and a success rate of de-orbitation.

3. **Explosion debris** have a distribution function in size $l$ that follows the NASA breakup model:

$$f_e \propto l^{-2.6}$$

Each population is truncated with minimum and maximum characteristic sizes $(l_{\min}, l_{\max})$.

4. **Collision debris** follow the same rule but with the exponent corresponding to the collision debris predicted also by the NASA breakup model

$$f_c \propto l^{-2.71}$$

The density of objects of a population at altitude $h$, described with a continuous size distribution function $f$, is

$$n(h) = \int_{l_{\min}}^{l_{\max}} f(h,l)dl$$

For a circular orbit, each object at altitude $r$ affects the local density according to a linear weighting algorithm:

$$\delta n_i = \left[ i + 1 - \frac{r - h_{\min}}{h_{\max} - h_{\min}} \cdot N_{\text{shell}} \right] \frac{1}{V_i}$$

$$\delta n_{i+1} = \left[ \frac{r - h_{\min}}{h_{\max} - h_{\min}} \cdot N_{\text{shell}} - i \right] \frac{1}{V_i}$$

where

$$V_i = \frac{4}{3} \pi (r_{i+1}^3 - r_i^3)$$

is the volume of the $i$-th spherical shell, the $i$-th gridpoint altitude being

$$r_i = \frac{1}{N_{\text{shell}}} \left[ i h_{\max} + (N_{\text{shell}} - i) h_{\min} \right].$$

For an object that has an elliptical orbit, the density increment $\delta n_i$ needs to be weighted according to the residence time of the object in each spherical shell. The simplest way is to use an elliptic function defined by [19]

$$f(\theta, e) = \int_0^\theta \frac{dx}{(1 + e \cos x)^2}$$

$$= 2 \arctan \left[ \tan \left( \frac{\theta}{2} \left( \frac{1 - e^2}{1 + e} \right)^{1/2} \right) \right] - e(1 - e^2)^{1/2} \frac{\sin \theta}{1 + e \cos \theta}$$

where $e$ is the orbit eccentricity. The density increment writes

$$\delta n_i = [f(\theta_{i+1}, e) - f(\theta_i, e)]/\pi$$

where $\theta_i$ is the phase angle corresponding to where the orbit cross the edge of the shell

$$\frac{p}{1 + e \cos \theta_i} = r_i,$$

$p$ being the ellipse parameter.

2.4. **Source terms**

Constellation objects and intact objects populations can be increased by taking into account predictions of **future launches**. Default background launch profile is calculated from a fit of the number of launches performed in year 2016 as a function of the altitude. Specific constellations can have customized launch profiles and launch scenarios implemented, as a function of time and altitude.

**Explosion** objects are generated from populations of intact objects and constellation satellites. The number of objects generated by one explosion for a population of explosion objects bounded by $(l_{\min}, l_{\max})$ is:

$$N_e = 6(l_{\max}^{-1.6} - l_{\min}^{-1.6})$$

according to the NASA breakup model again. At each altitude and each time step the explosion source term is defined by:

$$S_e = N_e \sum p_e n_e$$

where $p_e$ is the explosion annual probability and $n_e$ is the source object density. The summation is performed over all the intact and constellation populations.

The source term corresponding to the **collisions** is the most difficult to calculate because it requires to integrate
the collision cross section over the two distribution functions of the colliding populations. The cross section is defined by:

\[ \sigma(l_i, l_j) = \frac{\pi}{4} (l_i + l_j)^2 \]  

(14)

The number of debris generated by a collision between two populations \(i\) and \(j\) that have single masses \(m_i\) and \(m_j\) is according to the NASA breakup model:

\[ N_{c,ij} = 0.03 (m_i + m_j)^{0.75} \left( \frac{l_{\max}}{l_{\min}}^{1.71} - \frac{l_{\min}}{l_{\max}}^{1.71} \right) \]  

(15)

But the primary object mass depends on the size according to – again for a spherical geometry

\[ m = \rho_{\text{mat}} \frac{\pi l_i^3}{6}. \]  

(16)

The production rate due to collisions between populations \(i\) and \(j\) is hence:

\[ S_{c,ij} = v_{\text{ref}} \tilde{n}_j (1 - s_{\text{cam},ij}) \]

\[ \times \int_{l_{\min}}^{l_{\max}} \int_{l_{\min}}^{l_{\max}} f_i(l_i) f_j(l_j) \sigma(l_i, l_j) N_{c,ij}(l_i, l_j) \, dl_i \, dl_j \]  

(17)

where \(\tilde{n}_j = \frac{n_j - \delta_{ij} V_i}{1 + \delta_{ij}}\), \(V_i\) being the volume of the shell at the altitude considered and \(\delta_{ij}\) is a Kronecker operator \((\delta_{ij} = 1\) if \(i = j\), 0 otherwise), introduced here to avoid to count twice the same collision between objects of the same population. \(v_{\text{ref}}\) is the typical impact velocity at the altitude considered, which is close 10 km/s in most cases. In Eq.\[17\], \(s_{\text{cam},ij}\) is the probability of success of a collision avoidance manoeuvre when a collision between objects of populations \(i\) and \(j\) occurs. Of course, \(s_{\text{cam},ij} = 0\) when both populations are passive. The production generated by collisions between all population is then the sum over all the population pairs:

\[ S_c = \sum_{i,j} S_{c,ij} \]  

(18)

This population based approach allows for refining the model as much as needed for the simulation purposes. Finer description of fragmentation events is available through recent studies [20].

2.5. **Loss terms**

The loss terms modeling collisions and explosions come directly from the production terms but without taking the number of secondary objects generated by explosion or collision into account.

\[ L_c = p_e n_c \]  

(19)

\[ L_c = \sum_i \int_{l_{\min}}^{l_{\max}} \int_{l_{\min}}^{l_{\max}} f_i(l_i) f_i(l_j) \sigma(l_i, l_j) \, dl_i \, \int_{l_{\min}}^{l_{\max}} f_i(l_j) \, dl_j \]  

(20)

The model also includes end-of-life manoeuvres for populations of active satellites. If \(S_n(t, h)\) is the source term due to new launches, the loss term corresponding to end-of-life manoeuvres is:

\[ L_{\text{eol}}(t, h) = S_n(t - T_{\text{mission}} - T_{\text{eol}}, h) C_{\text{eol}} \]  

(21)

where \(T_{\text{eol}} = 25\) years is the deorbitation time after the end of operations recommended by the IADC, \(T_{\text{mission}}\) is the typical mission time of non-failing satellites, and \(C_{\text{eol}}\) is the level of compliance (between 0 and 1) to the 25-year rule.

For the constellations, an other model was included to account for the residence time of deorbited objects between their initial altitude and the altitude for which the effect of the drag becomes really significant on a short time scale. In this description, after the de-orbitation has started, the constellation objects start descending with a uniform velocity

\[ v_{\text{deorbit}} = (h_0 - h_{\min})/T_{\text{eol}} \]  

(22)

\(h_0\) being the mean altitude of the constellation. Let \(v\) be the probability of failure per unit of time. If what happens at each time-step is statistically uncorrelated, if \(N_0\) is the number of objects starting their deorbitation, the number of objects whose deorbitation system is still active after \(T_{\text{eol}}\) is \(N_{\text{deorbit}} = N_0 \exp(-v T_{\text{eol}})\). Moreover, the ratio \(N_{\text{deorbit}}/N_0\) is defined as the compliance \(C\) of the deorbitation system to the space regulations, which sets:

\[ v = -\ln(C)/T_{\text{eol}} \]  

(23)
Finally, a downward deorbitation velocity due to the drag is estimated by conservation of energy: during a revolution of period $T$ at a distance $r$ from the center of the Earth, an object undergoes an energy dissipation

$$W_{friction} = \frac{C_x}{2} \rho v^2 S \times 2\pi r$$

(24)

where $C_x$, $S$, and $v$ are the ballistic coefficient, the cross section and the orbital velocity. This must be equated to the loss in the total mechanical energy, which can be written as an effective potential energy:

$$E_m = -\frac{\mu m}{2r}$$

(25)

where $\mu = GM_T$. To the first order in $\Delta r$,

$$\Delta E_m = \frac{\mu m}{2r^2} \Delta r$$

(26)

Defining the deorbitation velocity by

$$v_d \approx \frac{\Delta r}{T}$$

(27)

leads to

$$v_d = \frac{C_x \rho S (\mu r)^{1/2}}{m}. \quad (28)$$

The number of objects lost due to the drag is just the divergence of the flux:

$$L_{drag} = -\frac{1}{r^2} \frac{\partial (r^2 n v_d)}{\partial r}$$

(29)

The velocity defined by Eq. 28 is analogous to a fluid velocity of a 1D fluid model. Therefore, the time and space discretization must satisfy the Courant-Friedrich-Lewy (CFL) condition for the numerical scheme to be stable. Thus, it is necessary that at the minimum altitude of the domain – where the downward velocity is the largest:

$$v_d \Delta t < \Delta x$$

(30)

Eq. 30 can be quite restrictive if low altitudes need to be included. Fig. 3 is an example of how this radial drag velocity changes with altitude. At 200 km, the typical velocity is 300 km/y, which means that for a spatial mesh of 10 km, the time step should be lower than 11 days.

$$\text{Figure 3: Radial drag velocity (pointing toward the Earth) as a function of the altitude for a spherical object of 2 m of diameter, 1 t/m}^3 \text{ of mass density, and a ballistic coefficient } C_x \text{ equal to 3, for the minimum and maximum densities corresponding to the solar activity.}$$

2.6. Numerical model

The time derivative of the densities are computed at each time step and each altitude for all populations. The differential system is solved using an explicit second order modified Euler method (midpoint method). The densities and their derivatives are totally vectorized in the altitude space, which yields faster memory access, hence a faster computation of the code.

3. Satellite constellation risk modelling

3.1. Motivation

Typical Monte-Carlo codes developed by space agencies [21, 22] require heavy computations to update the debris populations. The evolution of the environment is so difficult to predict by direct simulations using Monte-Carlo collisions for example, and the uncertainty on the position of each object on their orbits after several weeks of orbit propagation is so high, that a statistical models becomes relevant. With the steady rise of the total number of objects in LEO, tracking each individual objects is very challenging. Concepts such as debris densities and debris fluxes, inspired from fluid mechanics, become more and more relevant. Statistical models of the same kind as INDEMN were already used in the past to investigate
for example the criticality of the environment on the long run, i.e. several centuries [23, 24]. Large constellations of satellites such as OneWeb or Starlink will run with several thousands of satellites for several decades. Due to the limited lifetime of each individual satellite, the total number of satellites launched for the full operation of these constellations will outreach 10,000 satellites. This represents a major threat for the sustainability of the LEO environment, as emphasized in recent papers. The way these satellites are manufactured on semi-automated production lines is totally new to the industry, which makes the reliability of the whole system unpredictable. Several studies have shown that the stability of the environment critically relies on the level of compliance of the satellites to the 25-year de-orbitation rule, proposed by the IADC. About ten constellations with 100 to 4,000 satellites were announced since year 2015, many of which are quite hypothetical, but may well be implemented at full capacity by mid 2020’s. The financial scheme is still to be found, especially because of the high industrial risk uncertainty – including the risk of collision. The INDEMN software has flexible global and system-level inputs and can be updated instantly with the latest release of space-track.org. It provides an accurate estimate of the collision risk and the debris density, and helps identifying what are the critical parameters for the profitability of an industrial project or the sustainability of the environment as a whole, with respect to orbital pollution.

Before launching a constellation, the model evolution can be tested in various configurations, varying parameters such as the system reliability, the orbit spread of the various satellites, and the end-of-life scenario. It is a first step towards evaluating the necessity of implementing new end-of-life constraining measures, and active debris removal (ADR) actions.

### 3.2. Collision risk assessment

The collision frequency of one single object with population $i$ is computed the same way as the term of losses by collisions

$$v_{ci} = \sum_{i} v_{ref} \hat{n}_i \int_{l_{min}}^{l_{max}} \int_{l_{i,min}}^{l_{i,max}} f(l) f_i(l_i) \sigma(l, l_i) d l d l_i \left(31\right)$$

The individual risk of collision between two dates $t_1$ and $t_2$ is computed using Poisson’s law

$$r(t_1, t_2) = 1 - \exp \left(- \int_{t_1}^{t_2} \sum_{i} v_{ci} t \right) \cdot \left(32\right)$$

### 3.3. Simplified model of a constellation

The advantage of INDEMN over other tools is that the density can be computed accounting for the effect of the population of customer’s satellite on the environment. The correlated risk of collision can hence be estimated, including parameters such as the altitude spread or the reliability (see section [Appendix B.2](#) for a more thorough description of the population parameters).

Satellites of a same constellation are spread over various orbital planes, close to the reference value of the constellation altitude $h_{const}$. It is assumed that the satellites are spread uniformly in a spherical shell between $h_{const} - \Delta h/2$ and $h_{const} + \Delta h/2$. The parameter $\Delta h$ is called the altitude spread.

$$h = h_{const} \pm \Delta h/2 \quad \left(33\right)$$

$\Delta h$ has a significant influence on the collision risk.

The parameters chosen for the constellation correspond approximately to the OneWeb constellation, taken from publicly available information [25].

The cross section is a tumbling-averaged cross section, such that the satellite attitude does not affect the model. Since the 1990’s satellite operators have enforced passivation measures that have made in-orbit explosions very rare. Consequently, the explosion rate is set to 0 in all the simulation carried out in the next sections. The default value for the success rate of the post-mission disposal maneuver is 90%, which is slightly lower than what the operator claims, but much higher than operational standards.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>Altitude</td>
<td>1 200 km</td>
</tr>
<tr>
<td>Launch date</td>
<td>2019/01/01 a</td>
</tr>
<tr>
<td>Constellation build-up</td>
<td>2019-2021</td>
</tr>
<tr>
<td>Full operation</td>
<td>2022-2072</td>
</tr>
<tr>
<td>Satellite lifetime</td>
<td>5 years</td>
</tr>
<tr>
<td>Satellite cross section</td>
<td>2.7 m²</td>
</tr>
<tr>
<td>Altitude spread</td>
<td>40 km</td>
</tr>
<tr>
<td>Post-mission disposal success rate</td>
<td>90 %</td>
</tr>
<tr>
<td>De-orbitation time</td>
<td>2 years</td>
</tr>
<tr>
<td>De-orbitation scenario</td>
<td>Low thrust</td>
</tr>
<tr>
<td>Yearly explosion rate</td>
<td>0/year</td>
</tr>
<tr>
<td>Collision avoidance success rate</td>
<td>100 %</td>
</tr>
<tr>
<td>Launches per year</td>
<td>20 (deployment)</td>
</tr>
<tr>
<td>Satellites per launch</td>
<td>12 (replenishment)</td>
</tr>
</tbody>
</table>

Table 2: General parameters for the reference constellation used in this paper. The data is taken from Radtke et al. [25] (2017) and Bastida et al. [26] (2016) and are close to the scenario of the OneWeb constellations.

a As of August 2018, the launch date of the first satellites of OneWeb is expected to be between December 2018 and February 2019 (space-news.com).

During their de-orbitation maneuver, end-of-life satellites form a new population, that is not able to perform collision avoidance maneuvers anymore (0% success rate). This assumption has already been discussed in the past by other authors [27].

On-orbit collisions are critical events that may engage the third-party liability of the operator if it is shown that he is at fault. Moreover, it makes the operations of the other constellation satellites much more difficult to manage because the orbit is polluted. Nevertheless, since constellation operators planned back-up satellites in orbit, the fallouts on the service continuity should be minor. One can still reasonably assume that the profitability of the constellation as a whole would be at stake if a collision occurs. Therefore, the probability of causing at least one collision during the deorbitation phase is a key parameter, and it will be thoroughly investigated in this paper.

4. Practical cases

4.1. Influence of the constellation altitude

The altitude of the constellation characterized by Table 2 was varied with a fixed deorbitation time of two years. The consequences on the probability of causing one or more collisions during the deorbitation phase is presented in Fig. 4 in semi-logarithmic scale. The collision risk is larger for lower altitudes because the deorbitation speed increases with altitude. Consequently, the density of deorbiting satellites is larger for lower altitudes, hence the increase in collision risk. For an altitude of 1 000 km, the probability of causing at least one collision by 2050 is approximately 1.8%. This is a scenario with only one constellation, and with no fragmentations due to explosions. The post-mission disposal maneuver is performed within two years with a success rate of 90%. All these assumptions are optimistic so the results presented in Fig. 4 should be seen as a best case scenario.

4.2. Time of the de-orbitation maneuver

As discussed in former studies [26], the deorbitation time has a strong influence on the long term impact of a constellation on the LEO environment. These studies revealed that the IADC 25-year rule might not be sufficient to sustain the stability of the environment. Indeed,
Fig. 5 shows that the asymptotic slope of the long-term evolution of the collision risk is strongly affected by the time needed for constellation satellites to deorbit after their mission ended. By 2065, the risk of causing more than one collision is 3% in the 2-year deorbitation scenario chosen by the OneWeb constellation and 27% in the 25-year de-orbitation scenario recommended by the IADC, which may not be accepted neither by the operator, nor by the insurer or funder. Interestingly, the private sector has anticipated this risk and has implemented mission requirements that are more constraining than the international recommendations. It seems that the 25-year rule, that was decided at a time when announcements of large constellations were not so numerous, is not very well suited for the current paradigm.

4.3. Conditional risk

By default, INDEMN provides mean values of space debris density, from which parameters such as the collision frequency can be derived. In order to test the robustness of the outputs with respect to some rare events, external conditions can be imposed in the evolution scenario, and compared with the reference case. In Fig. 6, the red dashed line represents the average temporal risk profile for the constellation depicted by Table 2. In the scenario represented by the solid blue line, it has been assumed that a fragmentation has occurred at 800 km in year 2025 that generated 3 000 debris of more than 10 cm, independently from the constellation operation. This event is comparable with the collision between Iridium-33 and Kosmos-2251 that took place in 2009. Computation was performed for a 10-year de-orbitation scenario. Even though the collision considered here has taken place 400 km kilometers away from the constellation operational altitude, the influence on the collision risk for de-orbiting satellites is significant. In 2065, 40 years after the hypothetical fragmentation, the collision risk is as high as 25%, while it is only 15% in the reference case. By 2035, the collision risk reaches 5% in case of a catastrophic collision, as opposed to only 3% in the averaged reference scenario.

5. Conclusions

The first results obtained with the INDEMN software were presented in this paper. Among all the parameters of the constellation, the de-orbitation time was identified as a key driver of the total collision risk for end-of-life satellites. INDEMN is a very versatile tool to assess the long term collision risk at constellation level. For industrial use, it would be useful to implement best and worst case evolution scenarios, and to perform an uncertainty quantification analysis. Preliminary inspections showed that the risk profiles vary by more than one order of magnitude between the best and the worst case after three decades,
mainly because of the uncertainty on the future launches. The worst case scenario would include all the constellations that have been announced since 2015, which sums up to 15,000 more satellites in LEO. The 1D statistical model used here could be improved to 2D or 3D, so as to account for the orbit angles more precisely, as it was tested in some former studies [28, 29]. Adding a model for solar and Earth radiation fluxes has also been identified as a useful improvement [30].

In summary, INDEMN is a tool suited for mission analysis, constellation planning, policy impact assessment, and it could foster the definition of new standards of space sustainability, for example through the World Economic Forum Space Sustainability Rating (SSR) framework [31] or ISO [32]. Finally, this tool could be combined with a short-term collision prediction tool to build a Space Situational Awareness (SSA) software suite for debris-related risk forecasts, and could be used to assess the relevance of particular remediation strategies [33, 34].

6. Acknowledgment

This authors are thankful to C. Bonnal for the many helpful discussions that we had. Participation to the IAC 2018 was supported by a CNES IAC award.

Appendix A. Benchmarking INDEMN

Appendix A.1. IADC 2012 benchmark for the stability of the Future LEO environment

The IADC 12-08 report Stability of the Future LEO environment [17] is an initiative of the Inter-Agency Space Debris Coordination Committee aiming at comparing the simulation results of the simulation codes of six national or international space agencies against each other, starting with the same model assumptions and the same input data. The estimated 2012 initial population was provided by the Master 2009 for objects of size greater than 10 cm. It was assumed that there were no on-orbit explosion occurring (strong assumption) and that the compliance to the 25-year rule would be 90%. Instead of integrating the solar flux as it is done by five of the codes, INDEMN directly embeds a model for atmospheric density. The simulation is run for the LEO environment between 200 and 2000 km.

Only two populations were considered in this case, intact objects and collision debris. Following G.-L. Somma [7], all the intact objects are assumed to have a typical size of 2.1 m and the collision debris of 0.3 m (the distribution functions are Diracs for both populations), and the cross sections and masses are computed accordingly. The orbital collision velocity is fixed to 10 km/s for all altitudes. The ballistic coefficients are $C_x = 2$ also for both populations.

INDEMN shows an agreement better than 18% with all the other code at all time and better than 5% agreement with the ISRO sand ESA simulation tools. Considering the large discrepancy between the simulation codes, this is quite satisfactory.

Appendix A.2. Risk to space sustainability from large constellations of satellites

We propose a second simulation case to improve the validation of INDEMN based on a paper by Bastida et al. [26]. The assumptions for the background populations are basically the same as for the IADC-2012 benchmark and it is still assumed that there will be no spacecraft (SC) explosion. A population corresponding to a satellite constellation is added whose characteristics are summarized in Table A.3. The constellation satellites, like all other objects, are assumed to be passivated with a 100% success.

![Figure A.7: Comparison of the simulation results by various codes developed by national and international agencies with the results provided by INDEMN for the IADC 12-08 benchmark.](image-url)
rate, but are deorbited at the end of their operation time with a success rate varying between 50 and 100%.

These simulations emphasize the critical role of the reliability of the de-orbitation phase. Paradoxically, it is when the satellite is the least likely to be fully operational – at the end of its lifetime – that the most critical and challenging maneuver is required. The success of the de-orbitation maneuver is so critical for the sustainability of the orbit of operation that it should be totally independent and isolated from the rest of the system. The propulsion system used for de-orbitation, including its power supply if relevant, should be exclusively dedicated to this maneuver.

Appendix B. Code architecture

INDEMN is divided in 3 main classes:

- The TLE objects
- The population class
- The shell environment class

Appendix B.1. The TLE class

The TLE class contains the raw TLE data as attributes. A documentation of the TLE data is available at space-track.org. The launch date is converted to a python date-time format using the dtlaunch method, which makes it easier to extract directly the year of the launch. Orbital parameters such as the instantaneous right ascension, the semi-major axis, the angular momentum, the instantaneous longitude, the apogee and perigee are also calculated with methods that have been implemented into the TLE class.

Appendix B.2. The population class

The population class mainly contains a vector of densities at the various altitude shells. The type of population is also an attribute of the population class, and so are the ballistic coefficient, and the level of compliance to the 25-year rule C. The minimum and maximum sizes of the population objects, and the exponent coefficients giving the size distribution functions (1.6 for explosion debris, 1.71 for collision debris) are also attributes. Each source and loss terms (in km^{-3}s^{-1}) are computed using dedicated methods. Some of the integrals used in the computation of the collision terms are calculated and stored in dedicated dictionary data structures at code initialization.

Appendix B.3. The environment class

The environment class (shell_space) encapsulates most of the information used during a run. Its attributes are summarized below:

- A 1D dictionary of populations;
- A 2D dictionary of collision avoidance success rates between each pair of populations;
- The integrals used for the computation of the terms of source by collision (2D dictionary) and loss by collision (1D dictionary);
- The altitude shells stored in a simple vector;
- The parameters for atmosphere modeling;
- The relative date.

The TLE data are read directly from the shell_environment class through the add_TLE and add_TLElist methods. The evolution of the densities of the various populations is also computed within the shell_environment class, using a time vector (in years) as input, and embedding the modified Euler solver.

Appendix B.4. Output management

The successive environment data objects during a run are stored together in a list of states in memory. This allows to access all the information easily at the end of the run, for data storage and post-processing.
Bibliography


