

# A STATISTICAL LEO MODEL TO INVESTIGATE ADAPTABLE DEBRIS CONTROL STRATEGIES

Somma, G.L.<sup>(1)</sup>, Colombo C.<sup>(2)</sup>, Lewis H.G.<sup>(3)</sup>

<sup>(1)</sup> Faculty of Engineering and the Environment, University of Southampton, United Kingdom, Email: GL.Somma@soton.ac.uk

<sup>(2)</sup> Department of Aerospace Science and Technology, Politecnico di Milano, Italia, Email: Camilla.Colombo@polimi.it

<sup>(3)</sup> Faculty of Engineering and the Environment, University of Southampton, United Kingdom, Email: H.G.Lewis@soton.ac.uk

## ABSTRACT

Several strategies have been implemented or proposed to tackle the space debris problem. However, there is still debate on the feasibility, cost and effectiveness of these mitigation measures, especially in light of the increasing use of small satellites in low Earth orbit (LEO) and the drive towards space debris remediation.

This work presents a statistical source-sink debris evolutionary model of the Low Earth Orbit (LEO) with an innovative feedback proportional controller on Active Debris Removal (ADR).

The analysis presented here demonstrates that a proportional adaptive strategy that locally optimises the removal rate performs always better than a globally-optimised removal rate strategy in terms of total number of collisions, number of removal and end populations, lowering the end population and collisions respectively up to 14.09% and 13.24%.

## 1 INTRODUCTION

Since the beginning of the space age, the number of orbital debris has steadily increased, accounting now for more than 90% of the current Low Earth Orbit (LEO) catalogued population [1,2]. Moreover, even without ongoing launch activities, new explosions and collisions are likely to result in a continuing degradation of the environment, posing a growing hazard to future space activities [3]. To confront this threat, the Inter-Agency Space Debris Coordination Committee (IADC) was established in 1993; in 2002, its members reached an agreement on common guidelines for the reduction of space debris, later revised in 2007 [4]. Satellite manufacturers and operators are gradually implementing these measures, but the lack of a legally binding framework limits their widespread adoption. As highlighted in [5], the orbital debris population above 700 km should be intrinsically instable from both a physical

and a mathematical point of view, due to an increasing number of objects in space, the physical nature of fragmentation events and the ineffectiveness of drag above 700 km.

Orbital collisions are the main threat to the space environment due to the high number of fragments generated during each collision. Preventing some of these collisions, together with the widespread adoption of other mitigation measures, could be the key to preventing the increase of the space debris population. In the near future, it may be possible to decrease the risk of collisions by removing the most dangerous objects from space with Active Debris Removal (ADR). However, even significant investment and the removal of hundreds (or thousands) of objects will not give any assurance about the prevention of collisions. In contrast, collision avoidance manoeuvres can prevent (or at least decrease the risk of) a specific collision. They are nowadays routinely planned and performed by space operators whenever the risk of a collision exceeds a set threshold. For example, the ISS has performed 25 manoeuvres so far since its launch (in 1998) [6] and a collision avoidance manoeuvre might have also prevented the Iridium-Cosmos collision since the Iridium satellite was still active and manoeuvrable at the time of the collision [7].

In 2014, White and Lewis found that an adaptive strategy, based on a simple feedback control, was more efficient compared with a fixed ADR rate when the objective was to maintain the current debris population in a 200-year time span [8]. In this work, the same approach is used and expanded: an adaptive feedback controller on ADR is implemented and uses several different control laws, such as a fixed removal rate or a rate proportional to the orbital population. The latter control law can offer the option to adapt the rate of removals when and where there is need, i.e. when the population is increasing.

## 2 THE MODEL

### 2.1 Model description

A multi-shell and multi-species deterministic source-sink model for LEO has been developed [9]. It uses discrete time-steps and a system of first order linear equations to describe the population evolution of three object species (intact objects, explosion fragments, and collision fragments, see Figure 1) in a custom number of spherical concentric altitude shells in low Earth orbit (LEO), from 200 to 2000 km. In contrast with the previous model version [10], the initial population and launch profile are adapted to the spherical shells by dividing each object into multiple shells proportionally to the time spent by the object in each shell. In this way objects in eccentric orbits can be partially considered, including, for example, many rocket bodies in geosynchronous transfer orbits which spend only a fraction of their orbits (near their perigee) in LEO.

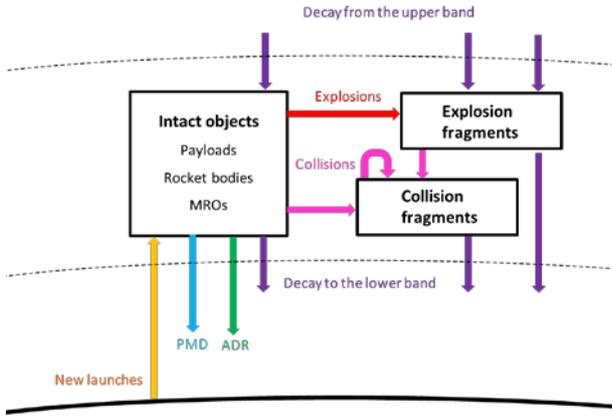


Figure 1. Schematics of object species in one of the altitude shells. Source and sink mechanism are also depicted as inbound and outbound arrows respectively.

With respect to Figure 1, explosions remove one intact object whereas collisions remove two objects from the relative species involved. The number of fragments generated during both explosions and collisions is computed a priori via the revised NASA break-up model [11,12] and stored in a lookup table for fast re-use. Drag is the only natural sink mechanism and is computed via a piecewise exponential model of the Earth's density with an average value of the solar activity [13,14]. Finally, if defined so in the input parameters, the model can remove intact objects in response to post-mission disposal (PMD) measures (with a custom residual lifetime and level of compliance), and ADR with either a fixed value or an automatic proportional controller with different control laws (see Section 2.3).

To simplify the problem, the model does not consider external factors such as the economy (e.g., the cost of remediation measures), politics (e.g., legal responsibility

and ownership) or possible future technology. The model uses several simplifying hypotheses. For example, collisions may only occur within the same altitude shell and objects can decay by only one shell at each time step. This could introduce some errors with a large time step, especially in the low altitude shells. However, this issue is minimised as, by default, the model uses a time step of 0.1 years and altitude shells of 100 km.

Drag is also the only perturbation considered; the model does not include solar radiation pressure, solar cycle, Earth's oblateness or other third-body perturbations.

### 2.2 System governing equations

The model uses a system of nonlinear first-order differential equations to handle the population derivatives. At each time  $t$ , the model evaluates the total number of objects  $N_T$  as

$$N_T(h_\eta, t) = \sum_{\eta} N_I(h_\eta, t) + N_C(h_\eta, t) + N_E(h_\eta, t), \quad (1)$$

where the subscripts  $I$ ,  $C$  and  $E$  respectively refer to the total number of intact objects, collision and explosion fragment, and the summation index  $\eta$  relates to the (custom) number of even-spaced altitude shells  $h_\eta$  in which the LEO region is divided. Eq. (1) expresses therefore the global summation over species and altitude shells at each time interval.

The choice of object species in Eq. (1) is dictated by the goal to simulate the behaviour and the interactions (e.g. the addition, removal, collisions) of objects within and among each species based on their characteristics. At the current stage, the model handles three different object species: intact, explosion fragments and collision fragments. Indeed, three is the minimum number of species and the most general grouping possible, while still maintaining meaningful physical differences among the species. However, to better characterise the species, future improvement will divide intact objects into active and inactive payloads, rocket bodies and mission-related objects (MROs).

In the same way as Eq. (1), the derivative of the total population in each altitude shell is expressed as a summation of the three terms::

$$\dot{N}_T(h, y, t) = \dot{N}_I(h, y, t) + \dot{N}_C(h, y, t) + \dot{N}_E(h, y, t), \quad (2)$$

where  $y$  indicates the cross-dependency of the term with other object species, and the subscript of the discrete altitude shells  $\eta$  has been dropped for clarity. These derivatives are computed in each altitude shell (and time  $t$ ) as the sum of several derivative terms:

$$\dot{N}(h, y, t) = \dot{C}(h, y, t) + \dot{D}(h, y, t) + \dot{E}(h, y, t) + \dot{L}(h, y, t) + \dot{M}(h, y, t) + \dot{U}(h, y, t), \quad (3)$$

where  $\dot{C}$  relates to collisions,  $\dot{D}$  to drag,  $\dot{E}$  to explosions,  $\dot{L}$  to launches,  $\dot{M}$  to mitigation measures, and  $\dot{U}$  to the control.

The three future states are computed from the current states with an explicit Euler method, [8,15,16]:

$$\begin{cases} N_I(t + \Delta t) = N_I(t) + \dot{N}_I(t + \Delta t, N_I(t + \Delta t))\Delta t, \\ N_E(t + \Delta t) = N_E(t) + \dot{N}_E(t + \Delta t, N_E(t + \Delta t))\Delta t, \\ N_C(t + \Delta t) = N_C(t) + \dot{N}_C(t + \Delta t, N_C(t + \Delta t))\Delta t, \end{cases} \quad (4)$$

where only the time dependency has been reported.

Rewriting Eq. (3) for the three species and applying some simplifications, a system of three equations is obtained:

$$\begin{cases} \dot{N}_I(y) = \dot{C}_I(y) + \dot{D}_I(h_\eta, h_{\eta+1}) + \dot{E}_I + \dot{L}_I + \dot{M}_I + \dot{U}_I(h), \\ \dot{N}_C(y) = \dot{C}_C(y) + \dot{D}_C(h_\eta, h_{\eta+1}), \\ \dot{N}_E(y) = \dot{C}_E(y) + \dot{D}_E(h_\eta, h_{\eta+1}) + \dot{E}_E(y), \end{cases} \quad (5)$$

where the dependence on altitude shell and time has been removed from each term for clarity, except for the drag term, which depends on the current and the upper altitude shell, and the control term, which can be a function of the population in multiple altitude shell.

The system of Eqs. (5) represents the core of the model. Collision and drag terms are common in all equations, while explosions remove intact objects and generate explosion fragments. The launch term adds new objects in the intact population, while mitigation and control remove objects from this latter species.

In a scenario without any explosions, the system of Eqs. (5) can also be used to investigate the evolution of the intact objects, existing objects, and new collision fragments. Indeed, removing the explosion term from the third equation of the system of Eqs. (5), the model would account only for atmospheric drag and collisions (for this third species)

### 2.2.1 Collisions

Two parts comprise the collision term in Eq. (3),

$$\dot{C}(h, y, t) = C_R(h, y, t)n_f(y), \quad (6)$$

where  $C_R$  is the collision rate, and  $n_f$  is the number of fragments involved in each collision. In the first two Eqs.

in (5),  $n_f$  refers to the objects removed in each collision,

leading therefore to a negative value for  $\dot{C}_I$  and  $\dot{C}_C$ . For example, a collision among two intact objects removes two of them, while collisions among an intact object and an explosion or collision fragment remove one of each. In the third equation of Eq. (5),  $n_f$  is equal to the number of fragments generated during each collision. It is computed a priori using the NASA standard breakup model [11,12], assuming that all collisions among intact objects are catastrophic and all the others are damaging.

Concerning the collision rate, the model uses the analytical laws derived from the kinetic theory of gases [9,17,18]. The collision rate among species  $i$  and  $j$  is computed (at each time step) as [7,19–21,15].

$$C_{R_{i,j}}(h, y, t) = p(h)\sigma(y_i, y_j) \frac{N(h, y_i, t)[N(h, y_j, t) - \delta_{i,j}]}{1 + \delta_{i,j}}, \quad (7)$$

where  $\delta_{k,l}$  is a Kronecker's delta (equal to one if both indexes are equal),  $\sigma(y_k, y_l)$  is the squared sum of the two object radii  $r_k$  and  $r_l$ ,

$$\sigma(m_k, m_l) = (r_k + r_l)^2 \quad (8)$$

(sometimes referred also as the square of the impact parameter), and  $p(h)$  is the intrinsic collision probability per unit of time as originally defined by Wetherill [9]. It is expressed by

$$p(h) = \pi \frac{v_r(h)}{V(h)}, \quad (9)$$

where  $V(h)$  and  $v_r(h)$  are respectively the volume of the altitude shell and the average relative velocity in the same shell. This latter is equal to about 10 km/s in LEO and therefore it was assumed to be independent of the altitude shell and takes on this value, as previously assumed by many other authors [19,22,23].

### 2.2.2 Natural Decay

The atmospheric drag is the only natural sink mechanism modelled, and therefore it is important to choose an appropriate expression for computing it. A known limitation of the current model is that all object species are subject to drag, including the active satellites that are part of the intact species.

Two terms constitute the decay rate  $\dot{D}$ . The first one refers to the number of objects that decay from the upper altitude shell into the current one (i.e. from  $h_{\eta+1}$  to  $h_\eta$ ), while the second term indicates the objects decaying from the current into the lower altitude shell (i.e. from  $h_\eta$  to  $h_{\eta-1}$ ):

$$\dot{D}(h_\eta, y, t) = + \frac{B(h_{\eta+1}, y)}{\tau(h_{\eta+1})} N(h_{\eta+1}, y, t) - \frac{B(h_\eta, y)}{\tau(h_\eta)} N(h_\eta, y, t), \quad (10)$$

where  $N(h_\eta)$  and  $N(h_{\eta+1})$  are the number of objects in altitude shell  $h_\eta$  and  $h_{\eta+1}$  at time  $t$ ,  $\tau$  is the characteristic residence time computed with a unitary area to mass ratio and unitary drag coefficient  $C_D$ , and  $B$  is a ballistic coefficient. This latter is a scaling factor of the residence time and is defined as

$$B(h, y) = c_D \left( \frac{A}{m} \right) (h, y), \quad (11)$$

where the average area-to-mass is applied to the initial population of each species  $y$  in each altitude shell  $h$ , and a flat plate model that gives  $c_D = 2.2$  is assumed [14]. The resulting matrix is then stored in a look-up table

The residence time, i.e. the time required for an object to decay from the upper to the lower boundary of each altitude shell, is computed a priori for each altitude shell and is stored in a lookup table to ease the computational cost of the model. Using the simplifying hypothesis that the change of semi-major axis over time (due to the atmospheric drag) is small over one orbit and can be approximated to the first order [24], the residence time is

$$\tau(h) = \frac{1}{\tilde{B}} \int_{h}^{h_{\eta+1}} \frac{dz}{\rho(z) \sqrt{\mu_E(z + R_E)}}, \quad (12)$$

where  $\rho(z)$  is the atmospheric density at altitude  $z$ ,  $\mu_E$  and  $R_E$  are respectively the Earth's gravitational parameter and radius. The integration occurs between the upper and lower boundary value of each altitude shell and  $\tilde{B}$  is a unitary normalised ballistic coefficient in order to have coherent physical dimensions in Eq. (10). With this latter simplification, the residence time results are equal for all the species. The computation of the decay rate in the Eq. (10) is then reduced only to the multiplication of values from the pre-stored look-up tables ( $B(h, y)$  and  $\tau(h)$ ) with the input number of objects  $N(h, y, t)$ .

Each shell specific residence time was obtained via a numerical integration of Eq. (12) on the initial population with a piecewise atmospheric density profile from 200 to 2000 km. The density profile derives from the CIRA-72 (Committee on Space Research International Reference Atmosphere) model with an adjustment in the atmospheric density  $\rho$  so to have a piecewise-continuous formulation [14]. It follows:

$$\rho(h) = \rho_0 \exp\left(-\frac{h-h_0}{H}\right), \quad (13)$$

where  $\rho_0$  is the atmospheric density at reference altitude  $h_0$ ,  $h$  the object altitude and  $H$  the scale height. Above 1000 km the density follows a single exponential law with a reference height of 1000 km.

### 2.2.3 Explosions

In the same way as the collision term, two parts comprise the explosion derivatives in (3):

$$\dot{E}(h, y, t) = E_R(h, y, t) n_E(y), \quad (14)$$

where  $E_R$  is the explosion rate and  $n_E$  is the number of fragments involved in each explosion.

As a simplifying hypothesis, only intact objects can explode and generate fragments. Indeed, referring to the system of Eqs. (5), in the first equation the  $n_E$  term is equal to minus one, while in the third equation the value of  $n_E$  is computed a priori using the NASA standard break-up model.

### 2.2.4 Launch profile

New objects are injected into different altitude shells via the launch term in Eq. (3). New payloads, rocket bodies and MROs increase only the intact population (see first Eq. of (5)) based on the selected launch profile as function both of altitude shell and time.

The yearly average from ESA's MASTER 2009 database (with a reference epoch of 1 May 2009) is taken as default launch profile.

### 2.2.5 Mitigation term

The mitigation removal rate  $\dot{M}$  is computed as

$$\dot{M}(h, y, t_{PMD}) = p_C \dot{L}(h, y, t), \quad (15)$$

where  $p_C$  is the percentage level of compliance with the post-mission disposal guidelines [4] and  $t_{PMD}$  is a future time.  $\dot{M}$  is not function on the object species, since launches only occur for intact objects. Therefore, the dependency on object species can be dropped, and (referring to Eq.(5)) it yields

$$\dot{M}_I(h, t_{PMD}) = p_C \dot{L}_I(h, t). \quad (16)$$

The term  $t_{PMD}$  corresponds to the future time when the objects (launched at the time  $t$ ) will be removed from the simulation. It is equal to the sum of the current time  $t$ , the satellite operative life  $t_{SOL}$ , and the residual lifetime  $t_{SRL}$  established by the mitigation guideline (and equal to 25 years by default):

$$t_{PMD} = t + t_{SRL} + t_{SOL}, \quad (17)$$

where all the terms are expressed in years.

The current model has the known limitation of not being able to perform end-of-life manoeuvres that re-orbit objects into other shells. The formulation presented (15) indeed does not move objects into a lower shell (or re-orbit above LEO) and does not let them decay due to drag, but instead simply keeps the objects in the same shell and then removes them after a certain time based on the Eq. (17).

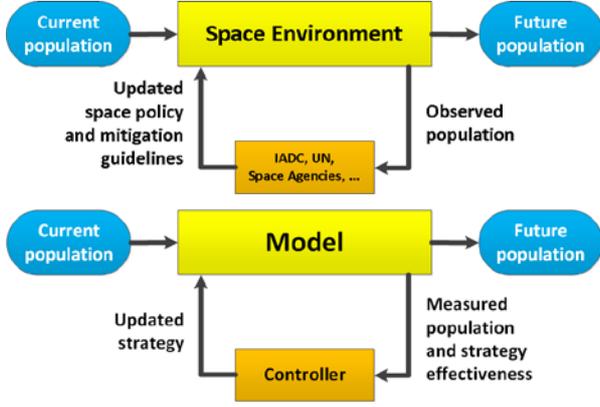


Figure 2. Schematic of the space environment (upper image) and the model architecture (lower image).

### 2.3 The controller

The main innovation of this work is the introduction of a controller that, similarly to the reality (see Figure 2), mimics the human-driven corrective actions of observing and reacting to the space environment evolution.

In the real world, the orbital population is observed with e.g. radars, telescopes, lasers. Based on these observations (and projections using evolutionary codes), international organisation such as the IADC and United Nations, space agencies and private companies, can enforce space policies and discuss mitigation guidelines to be implemented in future spacecraft. In the same way, the system of Eqs. (5) models the real space environment. Then, the model produces at every time step some outputs, including the orbital population, and passes them to the controller, which updates a debris-management strategy.

The basic idea, on which the controller is based, is to evaluate the current population at a fixed time interval (for example every year) and adapt a user-selected strategy to reach a selected objective.

The dependent variable checked is the total number of objects. However, many other choices are possible, such as the number of intact objects, collision fragments or collisions rates. Based on the observed values, the controller defines a new strategy to be applied in the

following time interval, as depicted in the schematic in Figure 2. The current controller acts only on the number of actively removed intact objects, but, in general, it can affect several parameters (for example, imposing a residual lifetime shorter than 25 years or limiting the launch rate).

#### 2.3.1 Fixed ADR Rate

Currently, the model can use two types of control law on ADR: a proportional controller or a fixed removal rate. This latter can be written as

$$\dot{U} = k_F, \quad (18)$$

where  $k_F$  is the selected number of objects removed per year. From a mathematical point of view, this is not an active controller since the number of objects removed remains constant during the whole simulation and is not dependent on any measured variable. However, Eq. (18) express a commonly accepted method for removing objects with ADR, with the only exception of the work of [8] on which this research is partially based. Moreover, the implementation of this fixed removal rate enables the numerical comparison of the model results with similar works in the literature, and produced with this model using other control laws.

#### 2.3.2 Adaptive proportional controller

The other option available is to use a proportional controller, which is a form of feedback widely used in control systems, where the controller observes an output value  $y(t)$  from a system and compares it to a specific set point  $r(t)$ . The obtained error,

$$e(t) = y(t) - r(t), \quad (19)$$

is then used to compute a control  $u(t)$  that is passed to an actuator that interfaces with the system.

In this model, the plant is the space environment itself (as can be seen from the model schematic depicted in Figure 3), and the outputs are the populations of the three species (plus the total population), as well as other useful variables such as the collision rates. The value of the control,  $u(t)$ , is defined as

$$u(t) = k_p e(t), \quad (20)$$

where  $k_p$  is the proportional gain and  $e(t)$  is the error on the total number of objects between the current measure  $N(t)$  and a set point  $N^*(t)$ , representing a population target:

$$e(t) = N(t) - N^*(t). \quad (21)$$

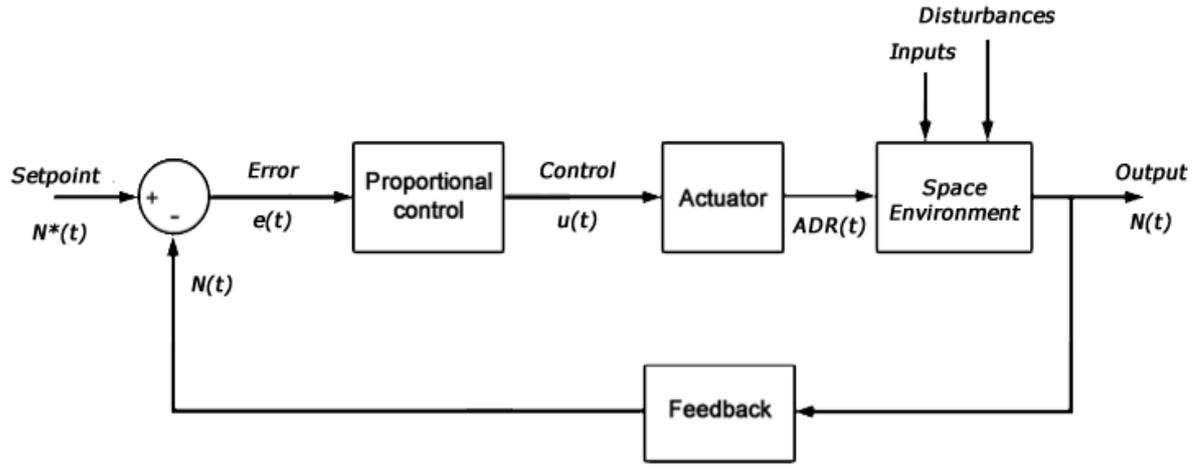


Figure 3. Schematics of a proportional controller for the space environment.

The proportional gain is defined as follows:

$$\begin{cases} k_p = 0 & \text{if } e(t) \leq 0 \\ k_p = \frac{u_{\max}}{e_{\max}} & \text{if } 0 < e(t) < e_{\max} \\ k_p = u_{\max} & \text{if } e(t) \geq e_{\max} \end{cases}, \quad (22)$$

with  $e_{\max}$  the maximum error possible above which the maximum control  $u_{\max}$  is used (see Figure 4). This simple proportional law is used to determine a removal rate from a minimum value of zero with a linear law up to the selected maximum value  $u_{\max}$ . This maximum value for the removal rate ensures that a realistic limit can be modelled and a fixed (but custom) amount of removals per year can be reached. Without this limit, the controller would have the possibility to reach unrealistic high values for yearly removals..

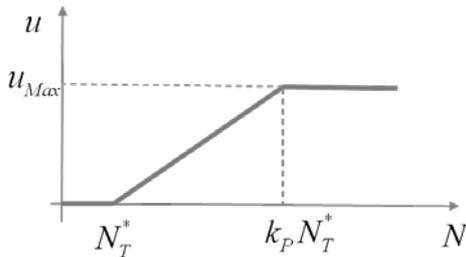


Figure 4. The proportional controller.

### 3 VALIDATION

The model was validated against the IADC comparison study of 2013 [3,25], which used very optimistic hypotheses: it assumed no new explosions (passivation effectiveness equal to 100%, and existing debris objects did not explode), and 90% of satellites decayed in 25

years after an operational lifetime of 8 years. The initial population and the launch profile of the study were from ESA's MASTER 2009 database (with a reference epoch of 1 May 2009). The population was projected forwards for 200 years, until 2209.

As the original IADC work was performed before the publication of [12], several models in [25] used the uncorrected formula in the standard NASA breakup model for generating collision fragments [11]. The same uncorrected formula was then utilised to ensure consistency in the numerical result comparison.

The validation analysis uses the same initial population and launch profile as the original IADC study, using 18 altitude shells and a time step of 0.1 years. To take into account the effect of orbit eccentricity, each object of the initial population and of the launch profile was proportionally divided into the altitude shells according to the time spent in each of them.

Collision-related values were computed using the same group of object species used in the original work: rocket bodies, payloads and MROs are grouped as intact objects, while debris represented existing fragments. Since there were no explosions, the third equation in (5) could be used to test existing intact objects that were only affected by drag. The new fragments were all the objects generated during the simulation (discounting newly launched objects); but, represented only the new collisions fragments. For this reason, the second equation in (5) could be effectively used to compute them.

Among the several models used in [25], the UK Space Agency's model DAMAGE [26] was selected for this validation. Table 1 lists the results of both DAMAGE results and the model, while Figure 5 shows a visual comparison.

The model achieved a similar trend and behaviour compared to the DAMAGE results for all object species. In both models, existing fragments tend to stabilise to a similar value, while new collision fragments became the dominant population after 2100. The number of existing fragments have almost a perfect match, with only 0.17% difference, proving the efficacy of the atmospheric and decay model. The total number of intact objects in the validation scenario was 17.06% smaller than in DAMAGE. This is caused by the inclusion of active payloads into the intact species. In this way, they are affected by the atmospheric drag and therefore there is a slighter higher number of intact objects decaying at each time step. This issue is known and will be fixed in the next model version, where intact objects will be divided into four sub-species (active and inactive payload, rocket bodies and MROs). The object grouping into three species affects also the number of new fragments generated, as spacecraft, rocket bodies and MROs have (on average) very different mass and dimensions (with

MROs about an order of magnitude lighter). As of matter of fact, even if the collision fragments are more numerous in the validation model (refer to Table 1), the total number of collisions was in good agreement: 61.03 compare to 63.37 for DAMAGE (with a 3.69% difference). The total population increased up to 20,579 in DAMAGE and 21,420 in the validation scenario, however in the validation scenario the extra collision fragments are partially balanced by the extra-decayed intact objects.

The DAMAGE results show some periodic ripples in all the population trends, as shown in Figure 5. These ripples are up to about 10% compared to the mean value and are caused by the periodic effect of the solar activity. Approaching a solar maximum Earth's atmosphere expands and so more objects decay; conversely after about 5.5 years, corresponding to low solar activity, the atmosphere shrinks and fewer objects decay in the same time interval. Currently, the model

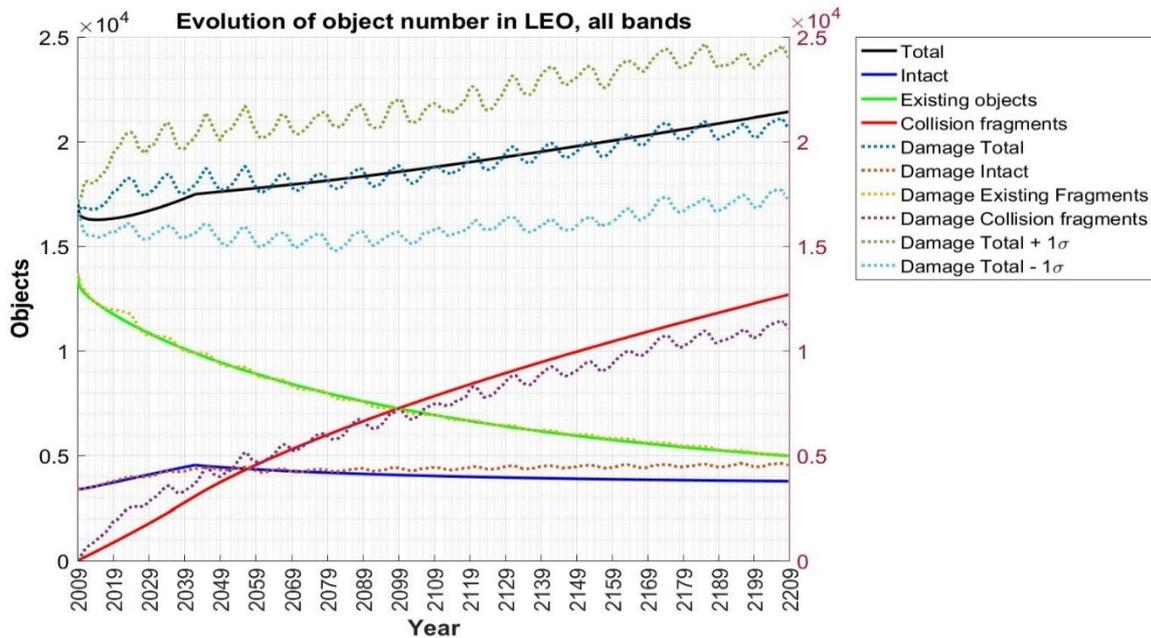


Figure 5. Comparison of the LEO population projection for UKSA's DAMAGE model (dotted lines) [25] and the validation model (solid lines).

Table 1. Results comparison of DAMAGE and the validation scenario.

Object Type	Initial population	DAMAGE		Validation Scenario		Difference [%]
		Final population	Change [%]	Final population	Change [%]	
<b>Intact objects</b>	3,410	4,540.2	+ 33.14	3,766.9	+10.47	-17.03
<b>Existing fragments</b>	13,697	4,978.5	-63.65	4,969.9	-63.72	-0.17
<b>New fragments</b>	0	11,060.3	-	12,683.6	-	-14.68
<b>Total</b>	<b>17,107</b>	<b>20,579.0</b>	<b>+ 20.30</b>	<b>21,420.3</b>	<b>+25.21</b>	<b>+4.09</b>

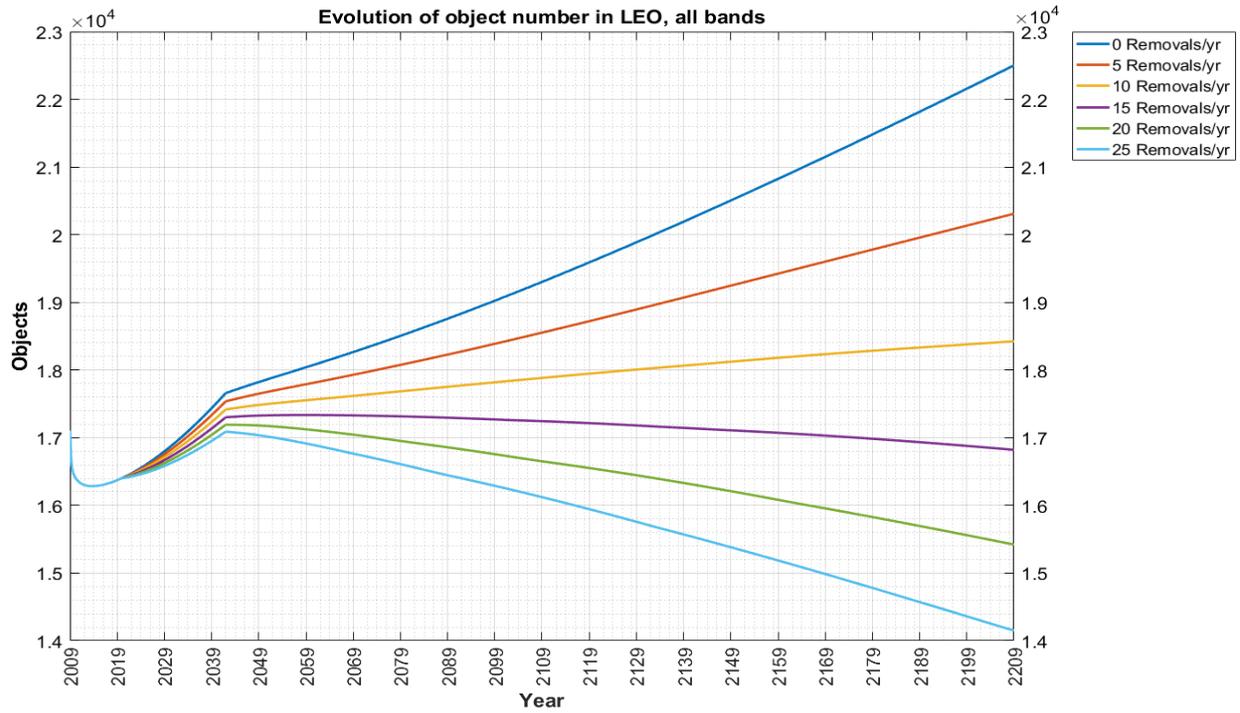


Figure 6. Comparison of total population for different removal rate.

Table 2. Control parameter, number of objects by species and total population at the end of the projected time-period performed with different removal rate.

Test name	Intact objects	Existing fragments	New fragments	Total population	Total collisions [1/yr]	Removal rate [1/yr]	Total number of removals
<b>F00</b>	3765,3	4969,8	13767,5	22502,6	63,07	0	0
<b>F05</b>	3168,9	4970,6	12170,5	20310,0	57,53	5	1000
<b>F10</b>	2677,1	4971,2	10776,7	18425,0	52,56	10	2000
<b>F15</b>	2289,5	4971,8	9559,6	16821,0	48,16	15	3000
<b>F20</b>	1960,0	4972,3	8488,4	15420,7	44,25	20	4000
<b>F25</b>	1637,6	4972,8	7540,9	14151,3	40,74	25	5000

presented in this work does not implement the solar cycle (it uses a mean solar activity instead), and therefore these ripples were not present. However, the effect is small when compared to the standard deviation produced in DAMAGE's Monte Carlo Simulations (see Figure 5). It should be noted that DAMAGE included Earth's oblateness and propagates all the orbital elements, while the model presented use a statistical source-sink approach. Nevertheless, the small error obtained demonstrates that it can be used to quickly perform simulations (in the order of seconds) later to be refined

with more complex and time-consuming models.

### 3.1 Fixed removal rate

Six simulations are conducted with a fixed removal rate equal to 0, 5, 10, 15, 20, and 25 object per year. Detailed results are listed in Table 2, while a visual comparison of the total population is also depicted in Figure 6.

As clearly shown in Figure 6, the initial trend is common to all the cases up to 2020 after which the ADR removals begin. A second features common to all the lines is the

sudden drop in slope in 2042. This behaviour is due to the application of the PMD guidelines to the intact population (with a 90% compliance). Indeed, after 8 years of operative life of the satellite and 25 years dictated by the guideline, in 2042 it starts the effective removal of object.

As expected, the case with no removal has the highest population, even compared to the validation case (in this latter case, it is due to the greater amount of fragments produced with the proper implementation of the NASA standard break-up model. Increasing the number of removals results in the reduction of the end population. However, 15 or more removals per year are needed to obtain a decreasing trend in the total population, while only in the scenario with 20 and 25 removals the total population at the end time is lower than the initial one.

### 3.2 Adaptive proportional control

Two different adaptive control strategies are tested. The first one, denoted as A, uses a controller proportional to the total number of objects at each time step, then computes a removal rate from Eq. (22) and splits the removals equally among all the shells. The second control strategy, denoted as B, is similar but the computed number of removals is split in all the shells proportionally to the number of object of each of them. Three different values were used for the proportional gain  $k_p$  (see Eq. (22) and Table 3): 1.5, 1.25, and 1.0. This last case is equivalent to an on-off controller that is active (with the maximum value allowed, equal to 25 removals per year) depending whether the threshold is exceeded or not.

As with the fixed removal rate, also with these two controllers, the total population is the same up to 2020 and in 2042 the same change in slope occurred on all curves due to PMD guidelines (refer also to Section 4.1 and Figure 7).

The adaptive strategies always perform better than those ones with a fixed removal rate, with respect to the size of the population at the end of the simulation. In the adaptive cases the end population never exceeds 20,000 (refer to Table 3) while with a fixed removal rate this happens only with 10 or more removals.

Observing Figure 7, it can be noted that the two strategies have a different behaviour. The A-strategy have increasing benefits with the reduction of the proportional gain parameter, but only with  $k_p = 1$  the end population is lower than the 2009 population. In this latter case, just after 2042 the population starts to decline due to the very high number of removals performed. After 2092, the total population is stable and the controller is mostly turned off. With strategy B the total end population decreases with  $k_p$  as well. However, the B-strategy it has always better performances than the A-strategy (with same  $k_p$ ) in terms of total collisions, total ADR, maximum yearly removals, and total end population and its derivatives at the end time. Even with  $k_p = 1.5$  the end population is only 0.09 % greater than the 2009 population, while with  $k_p = 1.25$  and  $k_p = 1.0$ , the end population is respectively 3.09% and 5.72% lower than at 2009, but with also negative derivatives in all three cases.

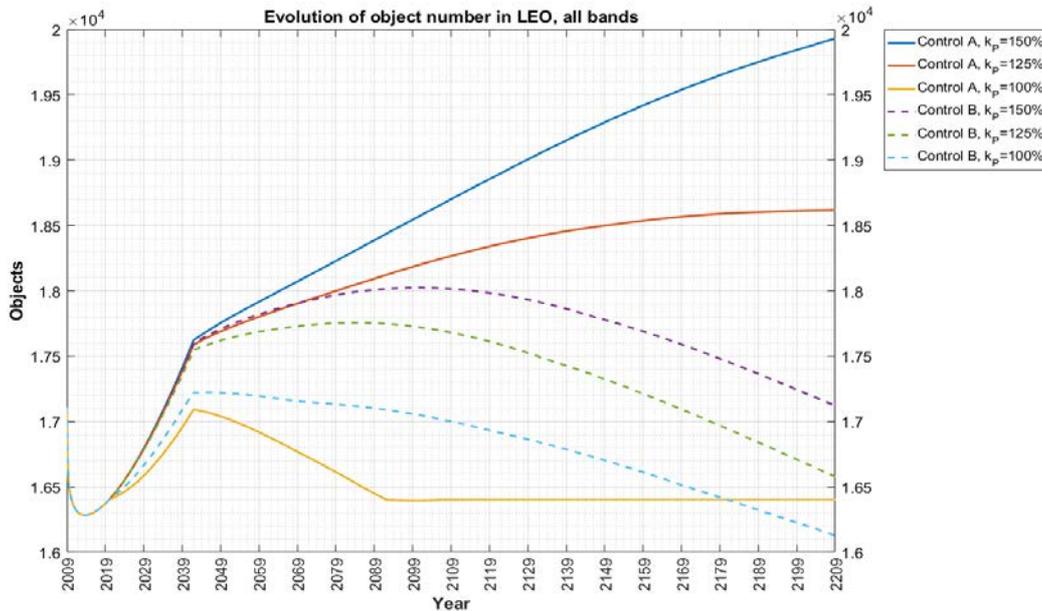


Figure 7. Comparison of total population for different control strategies.

Table 3. Control parameter, number of objects by species and total population at the end of the projected time-period performed with different control parameters. In the last column, the comparison is carried out against the scenario with no removals.

Test name	Control parameter $k_p$	Intact objects	Existing fragments	New fragments	Total population	Comparison of total end pop. [%]
<b>A1.5</b>	1,5	2950,4	4970,6	12009,0	19930,0	-11,43%
<b>A1.25</b>	1,25	2614,9	4971,0	11030,7	18616,6	-17,27%
<b>A1</b>	1	2376,6	4972,1	9054,0	16402,7	-27,11%
<b>B1.5</b>	1,5	2760,8	4971,4	9389,8	17122,0	-23,91%
<b>B1.25</b>	1,25	2670,7	4971,6	8935,9	16578,2	-26,33%
<b>B1</b>	1	2644,1	4972,0	8512,6	16128,7	-28,33%

Table 4. Detailed collisions and ADR-related results for the test performed with different control parameters. In the second column, the comparison is carried out against the scenario with no removals.

Test name	Total collisions [1/yr]	Comparison total coll. [%]	Total number of removals	Max ADR in shell [1/yr]	Max ADR [1/yr]
<b>A1.5</b>	57,39	-9,01%	1306,54	0,6	10,75
<b>A1.25</b>	53,97	-14,43%	1988,3	0,75	13,5
<b>A1</b>	45,32	-28,14%	2702,5	1,39	25
<b>B1.5</b>	49,79	-21,06%	1142,43	4,11	7,16
<b>B1.25</b>	48,24	-23,51%	1269,17	4,01	7,31
<b>B1</b>	46,05	-26,99%	1404,23	3,97	19,52

Between the two adaptive strategies with the same  $k_p$ , those using the second type always reached a lower final population with fewer removals. For  $k_p = [1.5; 1.25; 1]$  the number of collisions vary respectively by -13.24%, -10.62%, and +1.61%, while the total number of removals are lower respectively by 12.56%, 36.17%, 48.04%.

In the very short term the A strategy with  $k_p = 1.0$  could be preferred, as it reduces the total population more quickly. However, this behaviour is due only to the very high number of removals performed (the strategy uses the maximum allowed of 25 removals per year). For this reason the strategy is highly unrealistic, at least with current technology, space economy and inostructures.

#### 4 CONCLUSIONS

This paper presents a statistical multi-species source-sink LEO model capable of performing quantitative analysis and obtain results comparable to other evolutionary

models. The main innovation presented is a proportional controller that can adaptively select the ADR rate needed to achieve an objective.

A proportional adaptive strategy that locally optimises the removal rate (i.e. the B strategy, see Section 4.2) always performs better than a globally-optimised removal rate strategy (i.e. the A strategy), with the end population and collisions respectively up to 14.09% and 13.24% lower. Both strategies also can achieve a smaller end population after 200 years than a fixed proportional removal rate and with fewer objects actively removed.

The results demonstrate that the use of such removal strategies can greatly improve the effectiveness of ADR while meeting external constraints on the maximum number of removals due, for example, to logistic constraints (e.g. launch availability) and economic factors.

## 5 ACKNOWLEDGEMENTS

The Doctoral Training Partnership funded part of this research through the Engineering and Physical Sciences Research Council (EPSRC) Grant EP/M50662X/1.

ESA Space Debris Office provided data on the orbital population and launch traffic.

## 6 REFERENCES

- [1] Space-Track, Satellite Catalog (SATCAT) Data, Database. (2016). <https://www.space-track.org/> (accessed September 7, 2016).
- [2] Union of Concerned Scientists, Union of Concerned Scientists Database, Database. (2011) 12–14. [http://www.ucsusa.org/nuclear\\_weapons\\_and\\_global\\_security/space\\_weapons/technical\\_issues/ucs-satellite-database.html](http://www.ucsusa.org/nuclear_weapons_and_global_security/space_weapons/technical_issues/ucs-satellite-database.html) (accessed September 7, 2016).
- [3] J.-C. Liou, A.K. Anilkumar, B. Bastida Virgili, T. Hanada, H. Krag, H.G. Lewis, M.X.J. Raj, M.M. Rao, A. Rossi, R.K. Sharma, Stability of the Future Leo Environment – an IADC Comparison Study, Proc. 6th Eur. Conf. Sp. Debris. (2013) 1–8. doi:10.13140/2.1.3595.6487.
- [4] Inter-Agency Space Debris Coordination Committee, IADC Space Debris Mitigation Guidelines, 2007.
- [5] J.C. Dolado-Perez, C. Pardini, L. Anselmo, Review of the uncertainty sources affecting the long-term predictions of space debris evolutionary models, Acta Astronaut. 113 (2015) 51–65. doi:10.1016/j.actaastro.2015.03.033.
- [6] NASA Orbital Debris Program Office, Orbital Debris Quarterly News 2015 #4, Orbital Debris Q. News. 19 (2015) 1–14.
- [7] C. Pardini, L. Anselmo, Review of past on-orbit collisions among cataloged objects and examination of the catastrophic fragmentation concept, in: 64th Int. Astronaut. Congr., Beijing, China, 2013.
- [8] A.E. White, H.G. Lewis, An adaptive strategy for active debris removal, Adv. Sp. Res. 53 (2014) 1195–1206. doi:10.1016/j.asr.2014.01.021.
- [9] G.W. Wetherill, Collisions in the Asteroid Belt, J. Geophys. Res. 72 (1967) 2429–2444.
- [10] G.L. Somma, H. Lewis, C. Colombo, Adaptive remediation of the space debris environment using feedback control, in: 67th Int. Astronaut. Congr., Guadalajara, Mexico, 2016.
- [11] N.L. Johnson, P.H. Krisko, J.-C. Liou, P.D. Anz-Meador, NASA’s new breakup model of EVOLVE 4.0, Adv. Sp. Res. 28 (2001) 1377–1384. doi:10.1016/S0273-1177(01)00423-9.
- [12] P.H. Krisko, Proper Implementation of the 1998 NASA Breakup Model, Orbital Debris Q. News. 15 (2011) 4–5.
- [13] D.G. King-Hele, Satellite Orbits in an Atmosphere: Theory and Application, GlasgowBlackie, s.a., Glasgow, 1987.
- [14] D.A. Vallado, Fundamentals of Astrodynamics and Applications, 4th ed., Microcosm Press and Springer, Hawtorne, CA, 2013.
- [15] H.G. Lewis, G.G. Swinerd, R.J. Newland, A. Saunders, The fast debris evolution model, Adv. Sp. Res. 44 (2009) 568–578. doi:10.1016/j.asr.2009.05.018.
- [16] C. Kebschull, P. Scheidemann, S. Hesselbach, J. Radtke, V. Braun, H. Krag, Simulation of the space debris environment in LEO using an analytical approach, in: 40th COSPAR Sci. Assem., 2014.
- [17] P. Houston, Kinetic Theory of Gases, in: Chem. Kinet. React. Dyn., 2000: p. 53.
- [18] D.L. Talent, Analytical Model for Orbital Debris Environment Management, in: AIAA/NASA/DoD Orbital Debris Conf. Tech. Issues Futur. Dir., 1990: pp. 90–1363.
- [19] D.L. Talent, Analytic Model for Orbital Debris Environmental Management, J. Spacecr. Rockets. 29 (1992) 508–513.
- [20] P. Farinella, A. Cordelli, The Proliferation of Orbiting Fragments: a Simple Mathematical Model, Sci. Glob. Secur. 2 (1991) 365–378. doi:10.1080/08929889108426373.
- [21] L. Anselmo, A. Rossi, C. Pardini, Updated results on the long-term evolution of the space debris environment, Adv. Sp. Res. 23 (1999) 201–211. doi:10.1016/S0273-1177(99)00005-8.
- [22] C. Pardini, L. Anselmo, Review of past on-orbit collisions among cataloged objects and examination of the catastrophic fragmentation concept, Acta Astronaut. 100 (2014) 30–39. doi:10.1016/j.actaastro.2014.03.013.
- [23] D.J. Kessler, R.C. Reynolds, P.D. Anz-Meador, Orbital debris environment for spacecraft in low Earth orbit, 1990. doi:10.2514/3.26250.
- [24] D.G. King-Hele, D.M.C. Walker, The prediction of satellite lifetimes, 1987.
- [25] Inter-Agency Space Debris Coordination Committee, Stability of the Future LEO Environment, 2013. doi:IADC-12-08, Rev. 1.

- [26] H.G. Lewis, G.G. Swinerd, N. Williams, G. Gittins, DAMAGE: a dedicated GEO Debris Model Framework, in: Proc. 3rd Eur. Conf. Sp. Debris, The European Space Agency (ESA), Noordwijk, Netherlands, 2001: pp. 373–378. doi:10.1017/CBO9781107415324.004.