APPLICATION OF A CONTROLLED EVOLUTIONARY MODEL OF THE ORBITAL POPULATION TO TARGET SUSTAINABLE SPACE UTILISATION

Martina Rusconi and Camilla Colombo

Department of Aerospace Science and Technology. Politecnico di Milano, Via La Masa 34, 20156, Milan, Italy, Email: martina.rusconi@polimi.it, camilla.colombo@polimi.it

ABSTRACT

The space sector is rapidly growing due to advances in technology and more affordable space access, which has increased the risk of in-orbit collisions. This highlights the need for effective space debris management. Recently, sustainability in space has gained attention, with efforts focused on creating a sustainable orbital environment through efficient space activity management. This work presents a framework that combines environmental models and a control system to allocate mitigation resources for sustainability. Using a statistical one-dimensional model, the system simulates space environment evolution, considering factors like atmospheric drag, collisions, launches, and active debris removal. A state-dependent differential Riccati approach is applied for feedback control, handling non-linear dynamics. The novelty lies in automatically identifying actions needed to meet space debris targets. Results from this ERC-funded GREEN SPECIES project provide insights into optimal strategies for minimising risks and the effort required for mitigation.

Keywords: space debris; feedback control; sustainable space; mitigation strategy..

1. INTRODUCTION

After less than a century of activity the space sector has dramatically grown and launch numbers have been propelled by phenomena such as technology miniaturisation and a more affordable access to space. The consequence of this exponential growth in the population of orbital objects, without adequate management, is to increase the future risk of in-orbit collisions for current and future space activities. Recently, there has been growing attention to sustainability in space. The scientific community has been investigating ways in which a sustainable scenario for the orbital population could be obtained through efficient management of space activities (e.g. [1], [2]). This translates into understanding how different launch traffic cases, and combined mitigation and remediation measures, such as Post-Mission Disposal (PMD) activities and Active Debris Removals (ADR), affect the future space debris population. In parallel, past work aimed at discussing what could be a sustainable scenario and what metric could be used to quantify it (e.g. number and distribution of objects and debris, probability of collision and risk for satellites, and capacity indices [3]). Discussions around sustainability are ongoing, with the urgency of implementing mitigation measures shaping the future of space utilisation. The development of tools capable of analysing various mitigation approaches across different future scenarios or definitions of sustainable space would be invaluable. Such tools could simultaneously advance discussions on both defining sustainable space targets and determining the most effective strategies to achieve them. Previous research has primarily focused on investigating different strategies to reach a target scenario, often relying on trial-and-error approaches to approach that target [4]. In contrast, some studies have explored the formulation of automatic strategies, typically targeting a specific scenario and shaping control actions around it [5]. In [5] work, the control logic was explicitly dependent on the number of objects, aiming to reduce it.

The objective of this paper is to establish a unified framework that addresses both of these problems. This framework takes as input a definition of sustainable future and the methods to achieve it. Then it provides insight into how the environment can be influenced to reach the target scenario, the most efficient ways to do so, and the robustness of the target definition considering available methods. This approach also accounts for constraints and limitations associated with the implementation of mitigation measures and the relative impact of the use of different spatial regions. This paper presents an initial step towards defining this general framework. The proposed approach integrates the definition of sustainability metrics and figures of merit to evaluate the effectiveness of debris mitigation measures, while also defining a strategy to achieve the desired outcome based on these metrics. To do this, we apply an active controller that optimises a value function of an evolutionary model of space activities and debris growth. The model uses a statistical propagation of objects in space, considering environmental effects and space activities-related effects such as in-orbit collisions and launches, along with mitigation actions like PMD and ADR. These mitigation actions are managed by a state-dependent differential Riccati con-

Proc. 9th European Conference on Space Debris, Bonn, Germany, 1–4 April 2025, published by the ESA Space Debris Office
Editors: S. Lemmens, T. Flohrer & F. Schmitz, (http://conference.sdo.esoc.esa.int, April 2025)

troller, which efficiently achieves the desired future scenario based on an input value function. This function translates the environmental metrics into a performance function that is minimised via optimal control.
The work is part of the GREEN SPECIES project, funded by the European Research Council through a consolidator grant. The control and model in this study are closely integrated, meaning that the control action is informed by the dynamics of the model. This approach enables high control performance, as the expected behaviour of the model is known during a simulation, provided the problem is well-defined. In contrast, a "black-box" model would require extensive tuning through trial and error to customise the control capabilities and parameters on each simulation, which might lack clear physical interpretation, while the control in our study has clearer physical meaning, facilitating easier analysis.
Fig. 1 shows the general framework of the method. In principle it is independent of the model used as long as the required inputs can be given in the adequate shape. The environmental model and space activities are translated in state-space fashion, as will be better described in Section 2, while the environmental metric or target scenario definitions are given as input to the controller as a quadratic performance function. Any non-linear model and value functions can be inputted to the controlled framework through factorisation in time- and state-dependent form.
The complexity lies in defining a complex system in matrix form and the challenges posed by the curse of dimensionality, which could complicate computations for large matrices on limited computational resources. However, the benefits of a non-informed control approach, such as its broad applicability to complex models, are well-established and will be explored in future work. For example, [6] developed a non-linear model-predictive control for their shell-based environmental model.
The paper is structured as follows: Section 2 provides a description of each component of the framework in Fig. 1, including the environmental model developed for the control application, the derived state-dependent Riccati equations, and the presentation of the quadratic value function. in Section 3 an application of the framework in Fig. 1 is provided, along with validation of its benefits. Finally, conclusions are in Section 4.

2. METHODOLOGY

To take advantage of the framework's in Fig. 1 efficiency we apply control logic to a simplified environmental model. It is a one-dimensional, shell-based model that uses a statistical approach to propagate objects in space. The model considered in this paper is based on the one from a previous work of the authors in [7], and its key features are summarised below. The objects in Earth's orbit are treated as a continuous flow, moving in one-dimensional space under the influence of environmental effects, sources, and sinks. Propagation is performed by enforcing the conservation of the number of objects and fragments over time using the system of continuity equations for each of the N_s shells of the domain in Eq. 1. The objects' density (n) in space evolves over time, driven by atmospheric drag (relevant for the low-Earth orbit region) and in-orbit collisions. Atmospheric drag effect exploits King-Hele averaged formulation [8] and it's effect on the shell volume (V) of the continuity equation is included through the surface integral $\int_{S_{shell}} (v_r n)_{obj} dS$, where S is the boundary surface of the shell and v_r the radial velocity associated to the central radius r in that shell. Two object families are considered, each characterised by average physical properties. The first family consists of intact objects ($_{obj}$ in Eq. 1), including payloads, rocket bodies, and large debris with a cross-sectional area greater than 1 m^2. The second family accounts for smaller fragments ($_{frag}$ in Eq. 1). This distinction allows separate consideration of catastrophic collisions among intact objects, described as the product of collision rate $\dot{\eta}$ and number of generated fragments N_c or removed objects, and non-catastrophic collisions (generating N_{nc} fragments) between small fragments and intact objects, as assumed by the model in Eq. 1. It is important to note that the intact object family does not account for the active status of satellites. All items are assumed to passively de-orbit under the influence of drag during the simulation. This is a strong assumption made to simplify the model and reduce the number of species. While both families of objects are influenced by environmental dynamics and in-orbit breakups, space activities only affect the evolution of intact objects. The model considers a historical repetition of launch traffic from the five years preceding the initial simulation epoch that is modelled as a fixed yearly density deposition rate $\frac{N_{i_L}}{1year}$ in Eq. 1. PMD actions relocate satellites at the end of their operational life, based on launch traffic, to below the re-entry limit radius. Intact objects are placed below 630 km to re-enter within 25 years. The model assumes a fixed operational lifetime for intact objects, and at each time step, a percentage β of objects at the end of life in each shell (which in terms of removal density rate is $\int_{V_{t_L}} \frac{\dot{n}_{i_L}(t_L)}{1year}$) is removed through PMD and placed in the first shell below the re-entry limit, following the approach in [7]. Another method to reduce the number of objects in orbit is through ADR. In the model, a fixed number of removals per year is considered ($\frac{N_{i_{ADR}}}{1year}$), with a percentage α of these satellites being removed from a shell and considering complete re-entry through active de-orbiting.

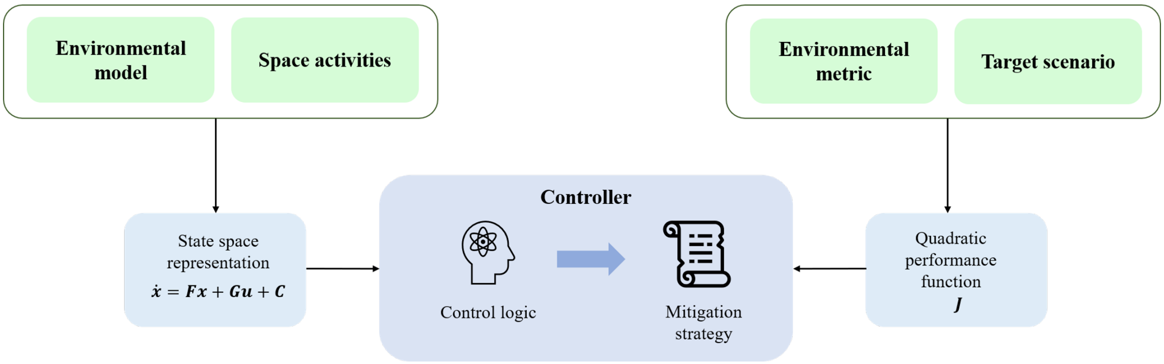


Figure 1. Framework of the method presented in the paper. On one side, an environmental model to propagate the density of objects under the effect of space activities is formulated in state-space fashion; on the other side a target function is defined to identify a desirable future scenario and it is formulated as a quadratic value function. The two interface with an active feedback controller for the automatic allocation of mitigation measures to efficiently reach the target environment.

$$\begin{cases} \frac{dn_{obj_i}}{dt} = & \frac{1}{V}\Big[- \int_{S_{shell}} (v_r n)_{obj} dS \\ & + \frac{N_{i_L}}{1year} - \beta \int_{V_{t_L}} \frac{\dot{n}_{i_L}(t_L)}{1year} \\ & -\alpha \frac{N_{i_{ADR}}}{1year} - 2\dot{\eta}_{i_{obj-obj}} \Big] \\ \vdots & \text{for} \quad i = 1, ..., N_s \\ \frac{dn_{frag_i}}{dt} = & \frac{1}{V}\Big[- \int_{S_{shell}} (v_r n)_{frag} dS \\ & -\dot{\eta}_{obj-frag} + \dot{\eta}_{obj-frag} N_{nc} \\ & +\dot{\eta}_{i_{obj-obj}} N_c \Big] \\ \vdots & \text{for} \quad i = 1, ..., N_s \end{cases} \tag{1}$$

By using a simplified model, we can apply a dynamics-informed control approach that exploits the expected environmental behaviour. The state-dependent differential Riccati control, in its most general form, has been applied here. This optimal control technique uses well-established linear control methods to define an optimal strategy that minimises a performance function. The formulation in [7] has been extended to generalise the control application for any non-linear, non-affine model and performance function. As illustrated in Fig. 1, the model interfaces with the controller by being written in matrix form, following a state-space representation, as shown in Eqs. 2 and 3, where the state $\mathbf{x}$ is the objects' and fragments' density in each shell, as in the system in Eq. 1 $\mathbf{x} = \begin{bmatrix} \mathbf{x}_{obj} \\ \mathbf{x}_{frag} \end{bmatrix}$. $\mathbf{F}$ is the state-matrix mapping the natural dynamics of the state due to drag and in-orbit collisions in Eq. 1; the control matrix $\mathbf{G}$ maps the control action $\mathbf{u}$, which is the application considered in Section 3 represents the percentage of PMDs and ADRs, in terms of density rate and the $\mathbf{C}$ vector accounts for disturbances to the model, e.g. launches in Eq. 1. The output parameter $\mathbf{y}$ in Eq. 3 represents any generic metric of the environment taken as performance parameter which is described as linear function of the state through the state-dependent target matrix $\mathbf{L}$.

$$\dot{\mathbf{x}} = \mathbf{F}(t, \mathbf{x})\mathbf{x} + \mathbf{G}(t, \mathbf{x}, \mathbf{u})\mathbf{u} + \mathbf{C}(t, \mathbf{x}) \tag{2}$$

$$\mathbf{y} = \mathbf{L}(t, \mathbf{x})\mathbf{x} \tag{3}$$

In general, any non-linear system can be written in a linear form through factorisation. This allows not only the collision dynamics to be state-dependent but also any matrix related to the sources and sinks, and control mappings to depend on the control action, making the system non-affine. The environmental dynamics, including drag and in-orbit collisions, and launches as disturbances in vector form $\mathbf{C}$ in Eq. 2 are defined as in [7]. The percentages of removed objects through PMD and ADR (β and α in Eq. 1) are used as control inputs, with the corresponding $\mathbf{G}$ matrices in E. 2 written accordingly. Additionally, the dependencies in Eq. 2 allow to account for state- or control-dependent constraints and limitations in the control action through the $\mathbf{G}$ matrix. An example of $\mathbf{G}$ matrix will be provided in Section 3. The other input in Fig. 1 is the definition of the target scenario and the metrics used to define it, which are translated in matrix form of Eq. 3. As mentioned in Section 1, various metrics are discussed to evaluate the environmental impact of human activities in space, which help define a target scenario. To apply the state-dependent Riccati control method, this target must be expressed as a quadratic performance function, as shown in Eq. 4.

$$J = \frac{1}{2}\mathbf{e}^T \mathbf{S}_f(t, \mathbf{x})\mathbf{e} + \frac{1}{2}\int_{t_0}^{t_f} \left(\mathbf{e}^T \mathbf{A}(t, \mathbf{x})\mathbf{e} + \mathbf{u}^T \mathbf{B}(t, \mathbf{x})\mathbf{u}\right) dt \tag{4}$$

The terms in this value function account for the time-integrated error $\mathbf{e}$ in the output $\mathbf{y}$ with respect to a reference value $\mathbf{r}$ weighted on matrix $\mathbf{A}$, and the integrated control effort $\mathbf{u}$ weighted on matrix $\mathbf{B}$, with an additional penalty at the final output value weighted on matrix $\mathbf{S}_f$. This penalty increases with the controller's effort to achieve the reference scenario, serving as a soft constraint. This function has been extended to account for state dependencies other than time. As a result, $\mathbf{y}$ can be any non-linear metric, which is factorised in quadratic matrix form. The weight matrices $\mathbf{A}$ and $\mathbf{B}$ also serve as a valuable tool for translating external constraints into penalties in the cost function. These matrices allow the model to prioritise achieving goals in certain regions of space or address the complexity of control implementation.

These two inputs, the state-space model and the performance function, are connected through the control logic, forming a general framework that enables the investigation of various implementation strategies for achieving a given target. This framework considers the available control actions and their limitations in real-world applications. The control logic is derived using standard optimal control techniques, following the optimality conditions outlined in Eq. 5, as detailed in the literature (e.g. [9], [10], [11]). In Eq. 5 H is the Hamiltonian defined as $H = \frac{1}{2}(\mathbf{y}-\mathbf{r})^T\mathbf{A}(\mathbf{y}-\mathbf{r}) + \mathbf{u}^T\mathbf{B}\mathbf{u} + \lambda^T(\mathbf{F}\mathbf{x}+\mathbf{G}\mathbf{u}+\mathbf{C})$ and λ is the co-state vector.

$$\begin{aligned}\frac{\partial H}{\partial \mathbf{u}} &= 0\\ \frac{\partial H}{\partial \mathbf{x}} &= -\dot{\lambda}\\ \frac{\partial H}{\partial \lambda} &= \dot{\mathbf{x}}\end{aligned} \tag{5}$$

In a regulation case, where the controller aims to bring the performance parameter to a target value within a finite time, the control logic is derived by writing the performance function in Eq. 4 in terms of error in the state with respect to the desired final value of the same from the reference $\mathbf{r}$ previously defined. Applying the assumption that the co-state λ takes the form $\lambda = \mathbf{S}\mathbf{e}+\mathbf{D}$, where $\mathbf{S}$ is a positive definite matrix of gains to drive the error $\mathbf{e}$ to zero, while the additional $\mathbf{D}$ vector accounts for disturbance rejection of the $\mathbf{C}$ term in Eq. 2 accounting for launches, the expressions in Eqs. 6 and 8 for the $\mathbf{S}$ matrix and the $\mathbf{D}$ vector are obtained, after some mathematical operations. Similar to [7], a numerical back-propagation approach is used to obtain the control matrices over time by solving the equations backwards with the final constraints in Eqs. 7 and 9. The final solutions are then obtained by forward-propagating the system and applying the control in Eq. 10.

$$\begin{aligned}\dot{\mathbf{S}} =& -\mathbf{S}\mathbf{F} + \mathbf{S}\mathbf{G}\mathbf{B}^{-1}\mathbf{G}^T\mathbf{S} - \mathbf{A} - \mathbf{F}^T\mathbf{S}\\ &- \left(\frac{\partial \mathbf{F}}{\partial \mathbf{e}}\mathbf{e}\right)^T\mathbf{S} - \frac{1}{2}\left(\frac{\partial \mathbf{A}}{\partial \mathbf{e}}\mathbf{e}\right)^T\\ &+ \frac{1}{2}\left(\frac{\partial \mathbf{B}}{\partial \mathbf{e}}\mathbf{u}\right)^T\mathbf{B}^{-1}\mathbf{G}^T\mathbf{S} - \left(\frac{\partial \mathbf{G}}{\partial \mathbf{e}}\mathbf{u}\right)^T\mathbf{S}\\ &- \left(\frac{\partial \mathbf{C}}{\partial \mathbf{e}}\right)^T\mathbf{S}\end{aligned} \tag{6}$$

$$\mathbf{S}(t_f) = \mathbf{S}_f \tag{7}$$

$$\begin{aligned}\dot{\mathbf{D}} =& \mathbf{S}\mathbf{G}\mathbf{B}^{-1}\mathbf{G}^T\mathbf{D} - \mathbf{F}^T\mathbf{D} - \mathbf{S}\mathbf{C}\\ &- \mathbf{S}\mathbf{F}\mathbf{x}_r - \left(\frac{\partial \mathbf{F}}{\partial \mathbf{e}}\mathbf{x}\right)^T\mathbf{D}\\ &+ \frac{1}{2}\left(\frac{\partial \mathbf{B}}{\partial \mathbf{e}}\mathbf{u}\right)^T\mathbf{B}^{-1}\mathbf{G}^T\mathbf{D} - \left(\frac{\partial \mathbf{G}}{\partial \mathbf{e}}\mathbf{u}\right)^T\mathbf{D}\\ &- \left(\frac{\partial \mathbf{C}}{\partial \mathbf{e}}\right)^T\mathbf{D}\end{aligned} \tag{8}$$

$$\mathbf{D}(t_f) = \mathbf{0} \tag{9}$$

$$\mathbf{u} = -\mathbf{B}^{-1}\mathbf{G}^T(\mathbf{S}\mathbf{e}+\mathbf{D}) \tag{10}$$

This process is discretised to retain information on the control at previous time steps, with a 1-day time step, which is small enough to approximate continuous behaviour while reducing computational costs. Information on the state at future times in the back-propagation process is provided by solving a simple state-dependent Riccati problem. In this simplified solution, all derivative terms are omitted, and the control solution is obtained by solving algebraic equations (see [7]). It is important to note that the matrix factorisation used in Eqs. 2, 3 and 4, which allows any model and cost function to be written in linearised terms, is not unique. The choice of factorisation influences the performance of the controller. Methods for optimally selecting the factorisation are discussed in the literature [12] and may be explored in future developments of this work. The method described here is versatile and adaptable to any non-linear and non-affine system that can be factorised. Various metrics and target scenarios can be fed into the control logic for the automatic definition of optimal control allocations to achieve the desired target. Different dynamics and models can be inputted to the controller, as long as they can be expressed in matrix form.

3. APPLICATION

An application of the framework outlined in Fig. 1 and described in Section 2 is presented here. To highlight the versatility of the method, a non-linear performance metric is considered, with control applied to the state-space model in Eqs. 2, 3 to define the optimal allocation of active removals and disposals, subject to constraints and preferences on both the strategy and performance. A reference population is used to define the initial density distribution to be propagated with Eq. 1. The population in [13] is used here, with the reference epoch set to 2022. The launch traffic term in Eq. 1 accounts for the repetition of launches that occurred during the 5 years prior to the reference epoch (see [7] and Fig. 2). The controller adjusts the percentage β of disposed objects via post-mission re-orbiting at the end of their operational life in each shell above 630 km, and the percentage α of actively removed objects above this threshold. The 630 km altitude is set as the limit for the control action, meaning that objects below this threshold are considered disposable within an acceptable time. The number of removals per year is fixed at 5 ADRs per year. Both controlled

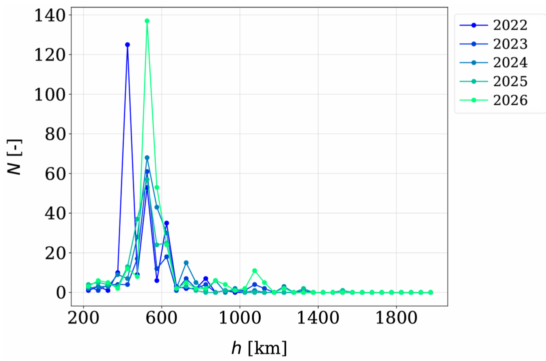


Figure 2. 5 years cycle of number of launched objects per each 50 km wide shell of the domain repeated during the simulations.

percentages are saturated at 99% for PMD and 100% for ADRs. The performance metric selected is the collision rate $\dot{\eta}$ in each shell of the environment. This is computed as shown in Eqs. 11 and 12, with separate contributions considered for collisions between intact objects and between fragments and intact objects (the cross-sectional area of the objects, σ_{obj}, is assumed much larger than the fragments' one).

$$\dot{\eta}_{i_{obj-obj}} = \frac{1}{2}\sigma_{obj} v_{r_i} n_{obj_i}(n_{obj_i} V_i - 1) \tag{11}$$

$$\dot{\eta}_{i_{obj-frag}} = \sigma_{obj} v_{r_i} n_{obj_i} n_{frag_i} V_i \tag{12}$$

The initial values for the collision terms are shown in Fig. 3, along with the desired final profile. The target is to halve the collision rate in all shells where it exceeds a threshold of 1×10^{-10} #/s. Since the collision rate is non-linear with respect to the state, it is factorised as in Eq. 13.

$$\begin{bmatrix} \dot{\eta}_{obj-obj} \\ \vdots \\ \dot{\eta}_{obj-frag} \\ \vdots \end{bmatrix} = \begin{bmatrix} L_{obj_{i,i}} & 0 & 0 & \cdots \\ 0 & \ddots & & \\ 0 & & L_{frag_{N_s+i,N_s+i}} & \\ \vdots & & & \ddots \end{bmatrix} \begin{bmatrix} \mathbf{n}_{obj} \\ \vdots \\ \mathbf{n}_{frag} \\ \vdots \end{bmatrix} \tag{13}$$

where:

$$L_{obj_{i,i}} = \frac{1}{2}\sigma_{obj} v_{r_i}(n_{obj_i} V_i - 1) + \sigma_{obj} v_{r_i} n_{frag_i} V_i$$

$$L_{frag_{j,j}} = \sigma_{obj} v_{r_i} n_{obj_i} V_i$$

The ability to introduce limitations and constraints into the control action implementation is also exploited by modifying the $\mathbf{G}$ matrix in Eq. 2 to the form in Eq. 14. In particular, ADRs are applied to those shells where the collision rate exceeds the acceptable threshold $\dot{\eta}_{thr}$ of 1×10^{-10} #/s (see Eqs. 16 and 17). The 5 removals per year are distributed to the shells with the highest collision rates, based on the collision rate share of each shell in Eqs. 16 and 18. The remaining matrices in Eqs. 15 (where j defines the number of controlled shells) and 16 map the percentage of disposed and removed objects in each shell as described in Section 2 and [7].

$$\mathbf{G} = \begin{bmatrix} \mathbf{G}_{obj} \\ \mathbf{G}_{frag} \end{bmatrix} = \begin{bmatrix} \mathbf{G}_{PMD} & \mathbf{G}_{ADR} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{14}$$

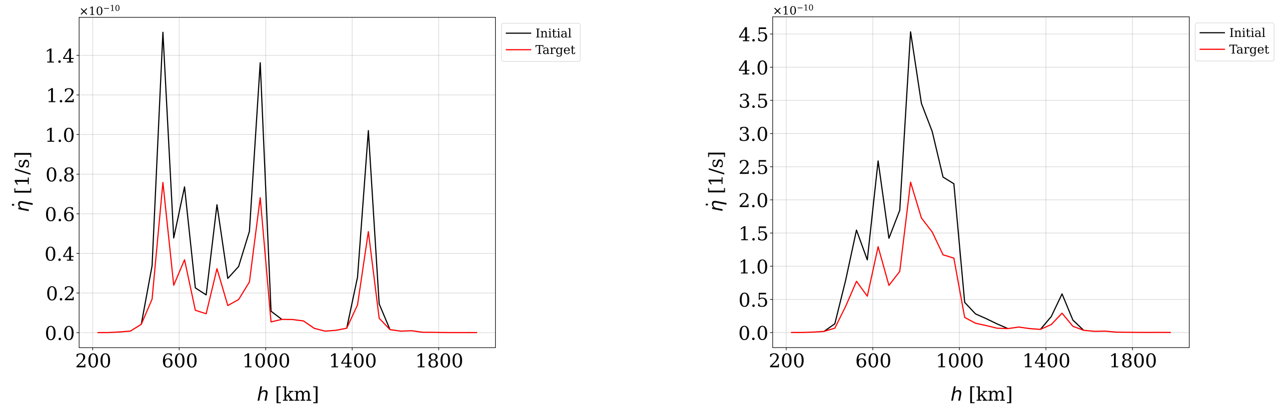
$$\mathbf{G}_{PMD} = \begin{bmatrix} 0 & \cdots & & \\ \vdots & \ddots & & \\ \frac{N_{PMD_1}}{V_1} & \frac{N_{PMD_2}}{V_2} & \cdots & \frac{N_{PMD_{N_s-j}}}{V_{N_s-j}} \\ -\frac{N_{PMD_1}}{V_1} & 0 & \cdots & \\ \vdots & \ddots & & \end{bmatrix} \tag{15}$$

$$\mathbf{G}_{ADR} = \mathbf{G}_{thr}\mathbf{G}_{\eta} \begin{bmatrix} 0 & \cdots & \\ 0 & \ddots & \\ -\frac{N_{ADR}}{V_1} & 0 & \cdots \\ 0 & \ddots & \end{bmatrix} \tag{16}$$

$$\mathbf{G}_{thr} = \begin{bmatrix} \frac{1}{2}\left(1 + \frac{\dot{\eta}_{obj_1} - \dot{\eta}_{thr}}{|\dot{\eta}_{obj_1} - \dot{\eta}_{thr}|}\right) & 0 & \cdots \\ 0 & \ddots & \\ \vdots & & \end{bmatrix} \tag{17}$$

$$\mathbf{G}_{\eta} = \begin{bmatrix} \frac{\dot{\eta}_{obj_1} + \dot{\eta}_{frag_1}}{\Sigma_i^{N_s}(\dot{\eta}_{obj_i} + \dot{\eta}_{frag_i})} & 0 & \cdots \\ 0 & \ddots & \\ \vdots & & \end{bmatrix} \tag{18}$$

Additionally, Eqs. 19-24 show an example of how constraints and non-homogeneous costs are translated into weighting terms in the performance function. The state weighting matrix $\mathbf{A}$ in Eq. 19 incorporates the maximum desirable error in the state during the simulation time t_f as $\Delta n_{obj/frag}$, which differs for intact objects and fragments to reflect their differing magnitudes. Errors are then weighted according to the local collision rates in each shell $\dot{\eta}_{obj} + \dot{\eta}_{frag}$, with more weight assigned where the collision rate is already high, which is translated with the matrix in Eq. 23. The control weighting matrix $\mathbf{B}$ considers the maximum acceptable control action during the simulation Δu, with control actions weighted by shell radius as in Eq. 24, so that removing objects at higher altitudes is considered more costly. Finally, the weights for ADR control are influenced by the local collision rates of Eq. 23, accounting for the more complex control implementation needed in regions where the collision risk is higher. This also results in greater weight being placed on ADRs rather than PMDs, privileging the latter in the mitigation strategy.



(a) Initial and targeted collision rate contribution from catastrophic object-object impacts.

(b) Initial and targeted collision rate contribution from non-catastrophic object-fragments impacts.

Figure 3. Initial collision rate (black line) and targeted profile (red line) with halved rate values when above 1×10^{-10} *#/s at initial time.*

$$\mathbf{A} = \begin{bmatrix} \frac{1}{\Delta n^2_{obj_1} t_f} & 0 & \cdots & \\ 0 & \ddots & & \\ & & \frac{1}{\Delta n^2_{frag_{N_s+1}} t_f} & \\ & & & \ddots \end{bmatrix} \mathbf{W}_\eta \tag{19}$$

$$\tag{20}$$

$$\mathbf{B} = \mathbf{W}_\Delta \begin{bmatrix} \mathbf{1} & \\ & \mathbf{W}_\eta \end{bmatrix} \mathbf{W}_h \tag{21}$$

$$\mathbf{W}_\Delta = \begin{bmatrix} \frac{1}{\Delta u^2_{PMD_1} t_f} & 0 & \cdots & \\ 0 & \ddots & & \\ \vdots & & \frac{1}{\Delta u^2_{ADR_1} t_f} & \\ & & & \ddots \end{bmatrix} \tag{22}$$

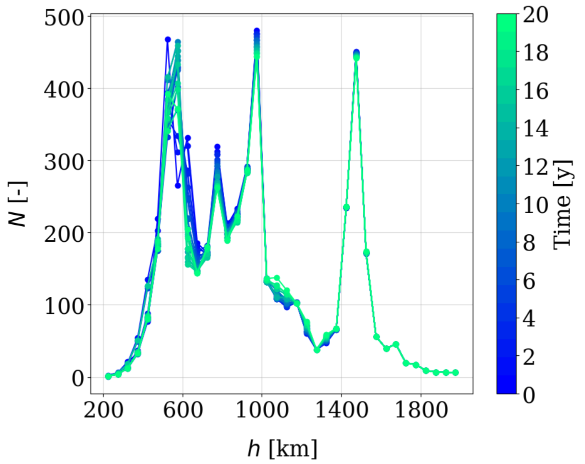
$$\mathbf{W}_\eta = \begin{bmatrix} \frac{1}{1 - \frac{\dot{\eta}_{obj_1} + \dot{\eta}_{frag_1}}{\Sigma_i^{N_s} (\dot{\eta}_{obj_i} + \dot{\eta}_{frag_i})}} & 0 & \cdots \\ 0 & \ddots & \end{bmatrix} \tag{23}$$

$$\mathbf{W}_h = \begin{bmatrix} \frac{1}{1 - \frac{r}{\Sigma_i^{N_s} r}} & 0 & \cdots \\ 0 & \ddots & \end{bmatrix} \tag{24}$$

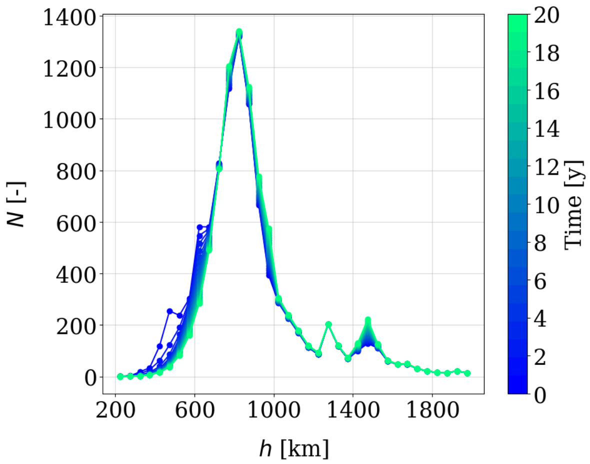
The regulation problem is simulated over 20 years. The yearly evolution of the number of intact objects and fragments is shown in Figs. 4, 5 and Fig. 6. The number of objects in Fig. 4a gradually decreases in most shells, except between 1000 km and 1400 km and above the peak at 1475 km. When compared with the collision rate distribution in Fig. 3, this behaviour is consistent with control action being applied only where the collision rate exceeds 1×10^{-10} #/s. The reduction in the number of fragments of Fig. 4b is more challenging above 630 km, due to the already large presence of fragments in that region, which contributes to collisions with intact objects and generate new fragments. Moreover, the reduction in intact objects is smaller above 800 km. This can be attributed to the launch traffic in Fig. 2, where very few objects are launched above 1200 km during the simulation. The few launched objects during the operational period decay to lower altitudes, making PMD more active there. Consequently, at the higher altitude peaks, ADRs are expected to compensate for the lack of disposed objects. However, since the number of ADRs is limited to 5 per year, they are distributed to the shells with the highest collision rates. With all these factors, the overall number of intact objects in Fig. 6a decreases, exhibiting an oscillatory behaviour caused by PMD disposals following the 5-year launch traffic cycle. While the number of fragments in Fig. 6b is initially reduced, it eventually begins to increase again as collisions accumulate, and the control actions are no longer effective enough to maintain the reduction.

As seen in Figs. 7 and 8 showing the yearly evolution of automatically selected disposals and removals, the number of PMDs exceeds the number of ADRs. The availability of objects for disposal is higher than the fixed number of removals. Additionally, as shown in Eq. 23, PMDs are preferred over ADRs. It must be noted that the results in Fig. 7b display ADRs from the continuous model of Section 2, where removals can be expressed as percentages of objects. The total number of PMDs exceeds that of ADRs, with an ADR rate of approximately 4 #/y.

Finally, the control performance is shown in Figs. 9-11 which provide the evolution of the collision rate across all shells of the domain, in terms of spatial and temporal distribution of the collision rate contributions in Eqs. 11 and 12 along with the overall cumulative rate in time. Fig. 12 shows the performance function in Eq. 4 over time. The collision rate is forced by the controller to decrease over the simulation period, exhibiting in Fig. 11 an oscillatory behaviour that reflects the trade-off between reducing state error and minimising control effort. However, as the simulation progresses the overall performance in-

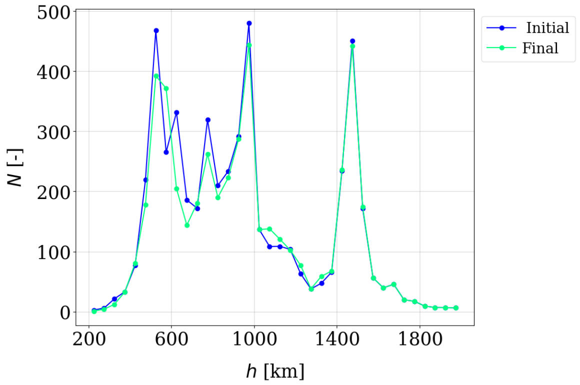


(a) Yearly evolution of the number of intact objects distribution in the shells of the domain.

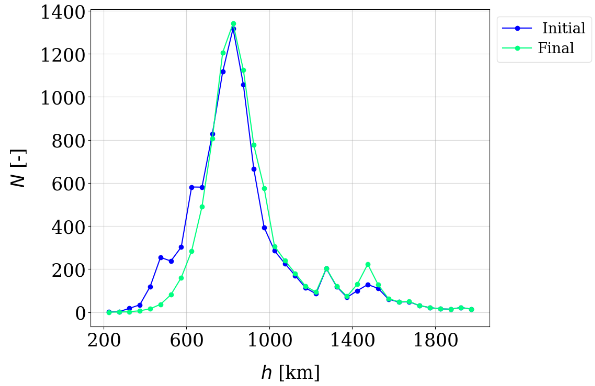


(b) Yearly evolution of the number of fragments distribution in the shells of the domain.

Figure 4. Number of intact objects (a) and fragments (b) distributions per-year simulated.

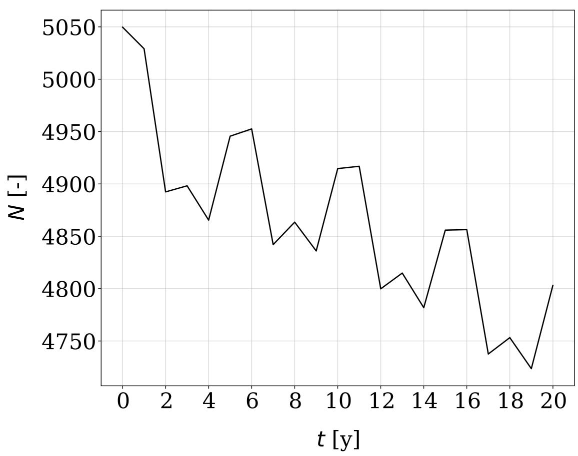


(a) Initial and final intact objects distribution in the shells of the domain.

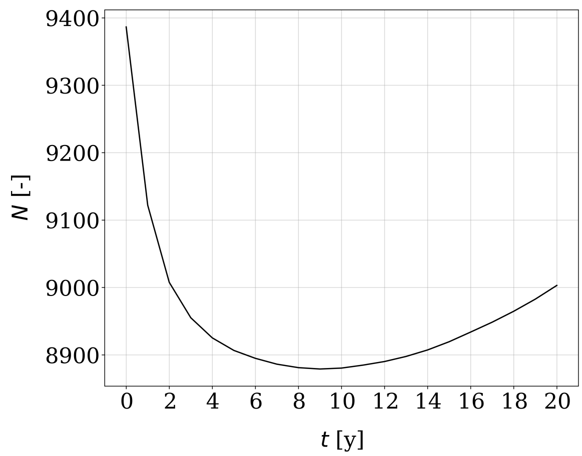


(b) Initial and final fragments distribution in the shells of the domain.

Figure 5. Number of intact objects (a) and fragments (b) initial and final profiles.

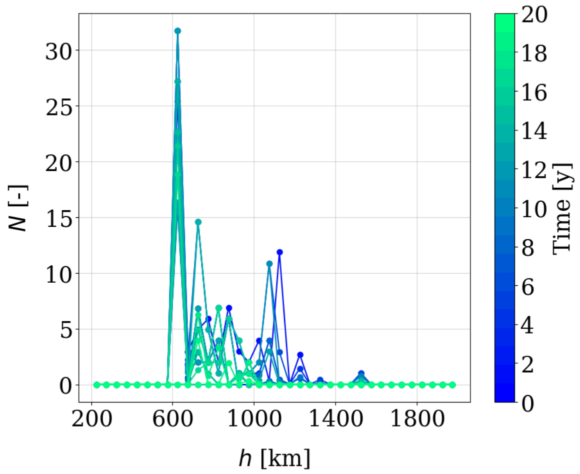


(a) Yearly evolution of the total number of intact objects over the entire domain.

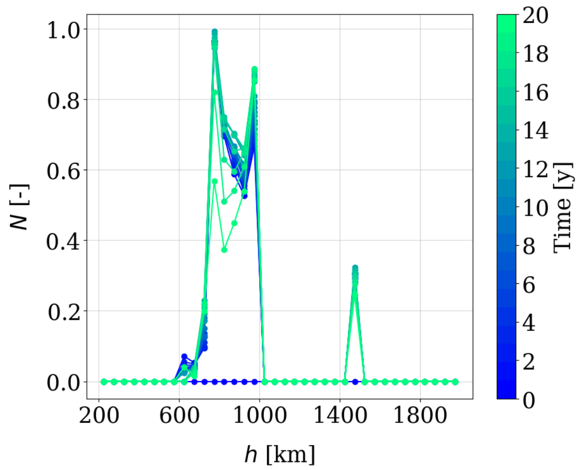


(b) Yearly evolution of the total number of fragments over the entire domain.

Figure 6. Yearly evolution of the total number of intact objects (a) and fragments (b) in the domain.

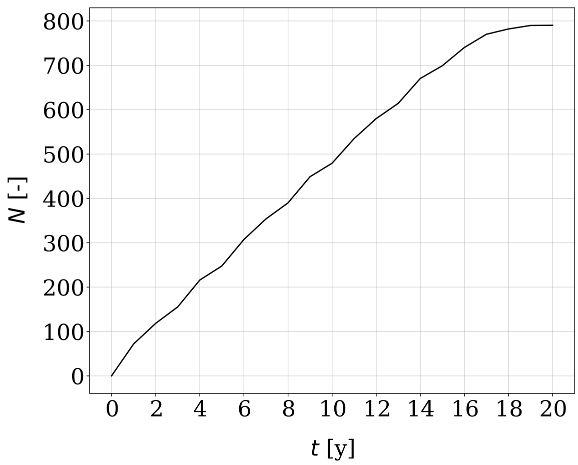


(a) Yearly evolution of the number of post-mission disposals performed in each shell of the domain resulting from the control action.

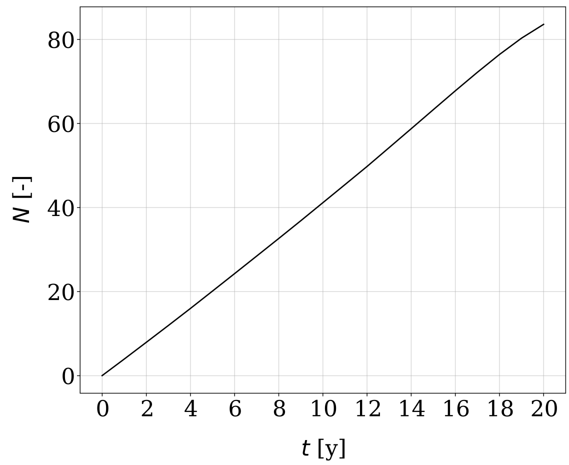


(b) Yearly evolution of the number of active debris-removals performed in each shell of the domain resulting from the control action.

Figure 7. Yearly evolution of the number PMDs (a) and ADRs (b) in each shell of the domain in the controlled simulation.



(a) Yearly evolution of the total number of post-mission disposals performed in all the shells of the domain resulting from the control action.



(b) Yearly evolution of the total number of active debris-removals performed in all the shells of the domain resulting from the control action.

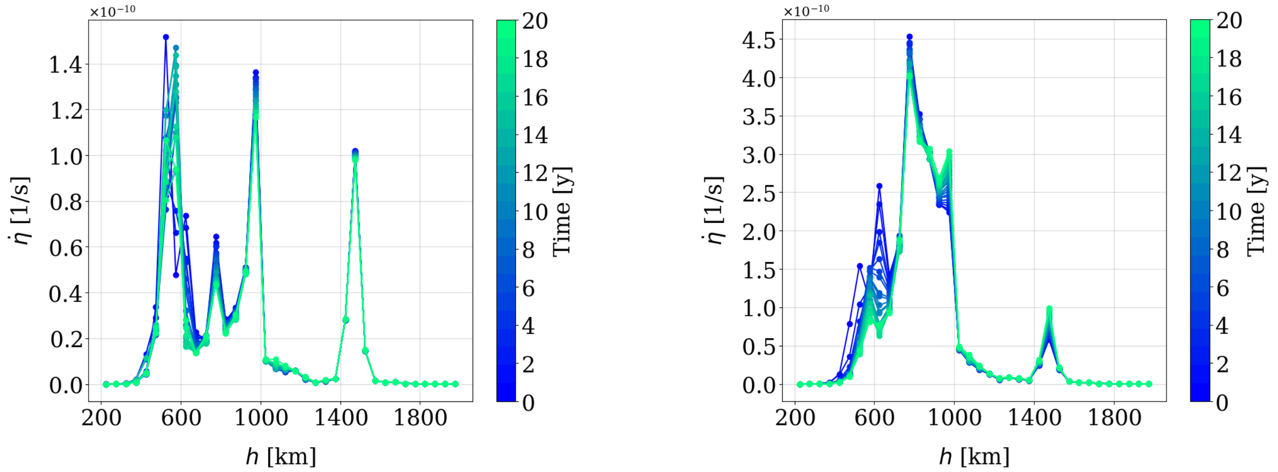
Figure 8. Yearly evolution of the total number PMDs (a) and ADRs (b) performed in all the shell of the domain in the controlled simulation.

dex of Fig. 4 is reduced, proving the correct action of the feedback controller.

The application described here illustrates the controller's ability to move closer to the target, utilising the available control capabilities, while incorporating constraints and preferences in their implementation. This enables the system to approach the desired target scenario in an efficient and automated manner.

To validate this, the results are compared with a 20 years simulation where the maximum control effort is applied without automatically defining its allocation, and without constraints or preferences such as those provided through the control mapping matrix $\mathbf{G}$ and weight matrices $\mathbf{A}$ and $\mathbf{B}$. The model used is the same as described in Section 2. PMDs are performed at their maximum rate, with 99% of the objects available for disposal being re-orbited to a lower shell at each time step. ADRs utilise all 5 available removals per year, but since ADRs have an additional degree of freedom than PMDs (in terms of where to perform the removals), a priori criteria for their implementation in space are needed. Since the goal is to halve the collision rate, as shown in Fig. 3, the 5 removals are placed at the locations with the highest collision rates: 775 km, 975 km, 625 km, 525 km, and 1475 km, with one removal performed in each shell each year. This case is labelled as 'Limit case' in the following results and the controlled case previously presented is labelled as 'Targeted case'.

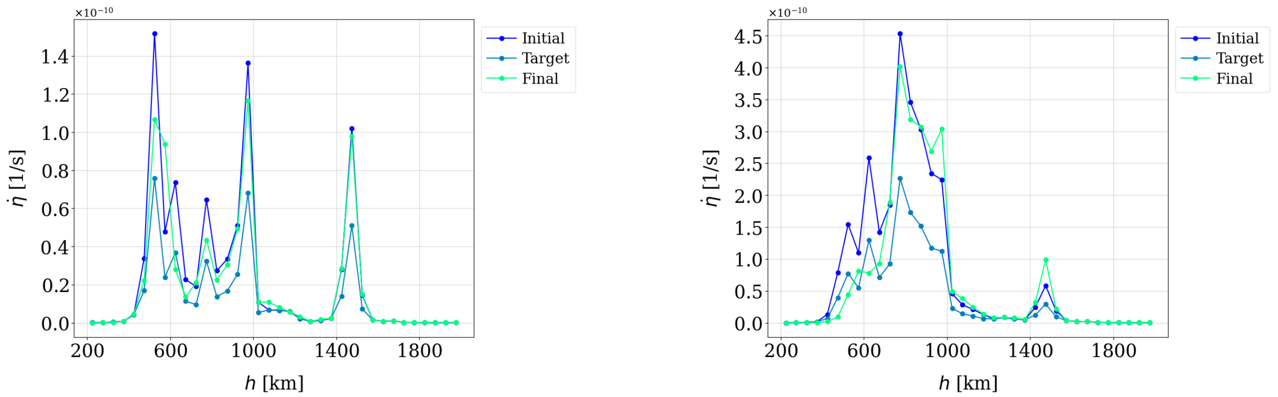
The initial and final distributions of the number of objects per shell, and the cumulative evolution over time for both intact objects and fragments in the Limit case, are



(a) Yearly evolution of the collision rate in Eq. 11 in each shell of the domain during the controlled simulation.

(b) Yearly evolution of the collision rate in Eq. 12 in each shell of the domain during the controlled simulation.

Figure 9. Yearly evolution of the collision rate contributions in each shell of the domain during the controlled simulation.



(a) Comparison between initial, targeted and final collision rates in Eq. 11 in each shell of the domain.

(b) Comparison between initial, targeted and final collision rates in Eq. 12 in each shell of the domain.

Figure 10. Comparison between initial, targeted and final collision rates contributions in each shell of the domain resulting from the controlled simulation.

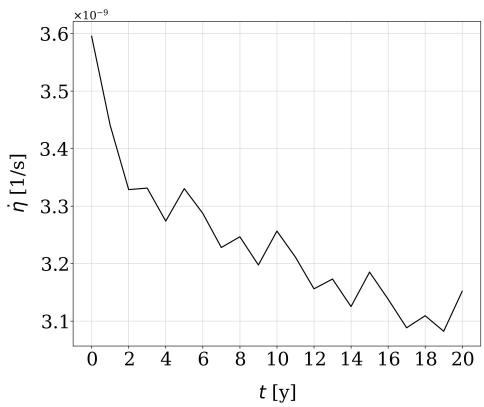
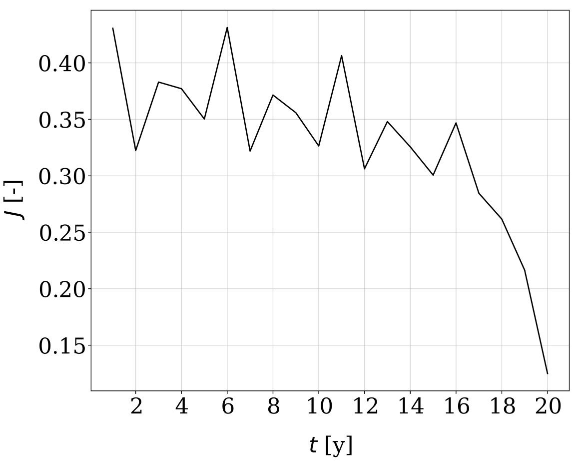
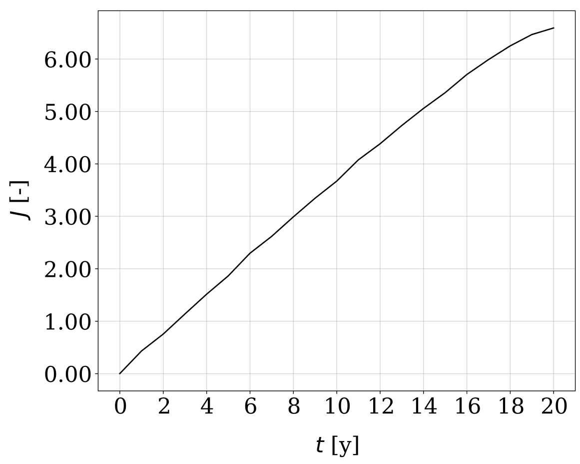


Figure 11. Yearly collision rate evolution computed as the sum of the contributions from Eqs. 11 and 12 over the entire domain during the controlled simulation.

provided in Figs. 13 and 14 and compared with the Targeted case previously described. It can be observed that the distributions and numbers of objects and fragments in the two cases are very similar. As expected, maximum mitigation actions results in more removed objects in the Limit case, with few less intact objects left in the environment at the end of the simulation and minimum effect on the number of fragments. In particular, the ADR removals that have been placed at the peaks locations previously defined cause an overall final distribution of intact objects in the Limit case that is lower than the number of intact objects in the same shells in the Targeted case. However, Fig. 15 shows that more PMDs and ADRs are implemented in the Limit case over the 20 years to get similar distributions in the number of objects of the Targeted simulation. Figs. 16 and 17 provide the effect of this larger control action on the collision rate evolution and performance function. The collision rate in Fig.

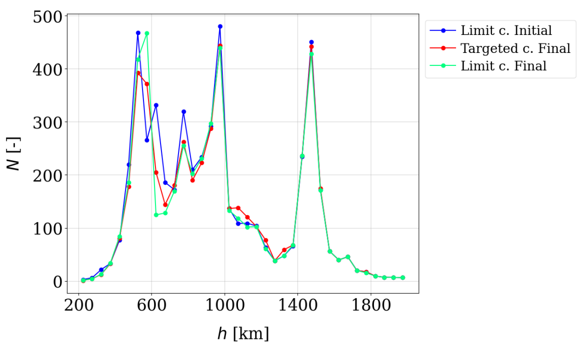


(a) Yearly evolution of the value function in Eq. 4 during the controlled simulation.

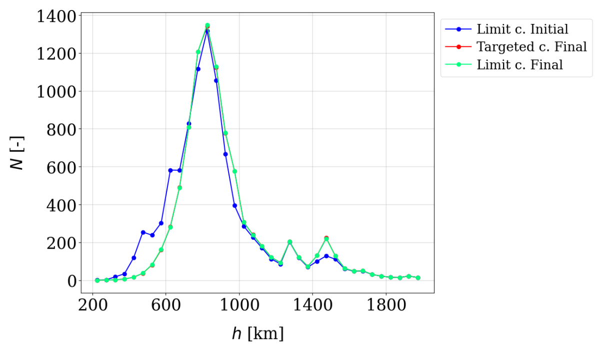


(b) Yearly evolution of the cumulative value function in Eq. 4 during the controlled simulation.

Figure 12. Performance metric evolution throughout the controlled simulation.

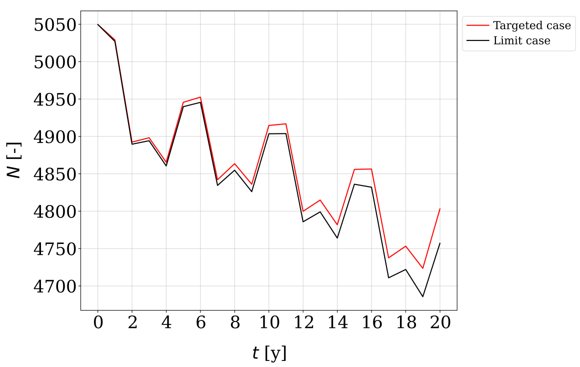


(a) Initial and final intact objects distribution in the shells of the domain for the Limit case, which are compared to the final intact objects profile of the Targeted case (red line).

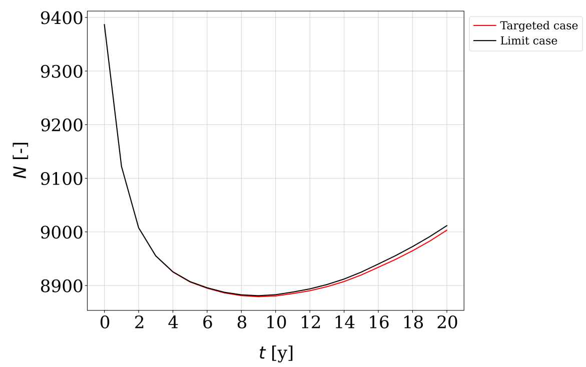


(b) Initial and final fragments distribution in the shells of the domain for the Limit case, which are compared to the final fragments profile of the Targeted case (red line).

Figure 13. Number of intact objects (a) and fragments (b) initial (blue) and final (green) profiles of the Limit case compared with the final profiles (red) of the Targeted case.



(a) Yearly evolution of the total number of intact objects over the entire domain for the Limit case (black), which is compared to the same evolution for the Targeted case (red).



(b) Yearly evolution of the total number of fragments over the entire domain for the Limit case (black), which is compared to the same evolution for the Targeted case (red).

Figure 14. Yearly evolution of the total number of intact objects (a) and fragments (b) comparison between Limit (black) and Targeted (red) cases.

16 for the Limit case simulation closely mirrors that of the Targeted case simulation, but the latter is capable of achieving better results in terms of collision rate reduction. Moreover, the cost of achieving this result, in terms of the performance function, is much higher in the Limit case, as shown in Fig. 17.
The results in Figs. 13-17 demonstrate that defined some control actions and their associated constraints and limitations, which can be translated into control information, the framework in Fig. 1 is capable of applying them efficiently to reach a target.

4. CONCLUSIONS

The discussion on sustainability in space continues, while the urgency to implement mitigation measures is shaping the future of space utilisation. The development and definition of tools that can provide and analyse various mitigation approaches to address any future scenario or any definition of desirable sustainable space would be highly beneficial. This work presents a general framework that connects environmental models of space activities and debris growth with a controller that efficiently allocates mitigation resources to reach a specific target. The proposed approach offers a way to incorporate different definitions of metrics and figures of merit to evaluate the sustainability of space activities and the efficiency of debris mitigation measures. It directly links these evaluations with the definition of an appropriate strategy to achieve the desired behaviour of these metrics.
A statistical one-dimensional model is used to analyse the expected future evolution of the space environment under various modelled effects, including atmospheric drag, in-orbit collisions, launches, post-mission disposals, and active debris removals. The state-dependent differential Riccati approach is applied to control the model in feedback, extending it to its most general form that can be used on non-linear and non-affine models. This tool can be used with any model that can be factorised in linear matrix form, and any evaluation metric for the sustainability of space activities that can be expressed as a quadratic performance function can be analysed. Additionally, since many factors affect the implementation of mitigation measures, beyond merely targeting a specific scenario, limitations on control applications and the relative importance of different parts of the control strategy can be incorporated as weights in the performance function.
These extensions of the method open up a variety of scenarios to be explored. An application has been presented to demonstrate how constraints and preferences in mitigation actions can be translated into information for a controlled-system framework. The results confirm the controller's ability to efficiently allocate resources with limited cost in terms of the value function, bringing the system as close as possible to the target scenario.
Future work will focus on further applications of the developed model and on enhancing the model's realism for example by incorporating explosion modelling and subdividing objects into more species.

ACKNOWLEDGMENTS

The research received funding from the European Research Council (ERC) under the European Union's Horizon Europe research and innovation program as part of the GREEN SPECIES project (Grant agreement No - 101089265). The simulations have been performed using as reference population the data in [13], kindly provided by the European Space Agency.

REFERENCES

1. Lewis H. G., White A. E., Crowther R., Stokes H. Synergy of debris mitigation and removal. *Acta Astronautica*. Vol. 81, Issue 1, 2012, pp. 62-68, ISSN 0094-5765, https://doi.org/10.1016/j.actaastro.2012.06.012.

2. Letizia F., Lemmens S., Bastida Virgili B., Krag H. Application of a debris index for global evaluation of mitigation strategies. *Acta Astronautica*. Vol. 161, 2019, pp. 348-362, ISSN 0094-5765, https://doi.org/10.1016/j.actaastro.2019.05.003.

3. Letizia, F., Colombo, C., Rossi, A., Muciaccia, A., Giudici, L., Harada, R., Kawamoto, S., Böttcher, L., Ruch, V. and Taillan, C. Mission-based and environment-based approaches for assessing the severity of a space debris evolution scenario from a sustainability perspective. *In Proceedings: 75th International Astronautical Congress (IAC)*. Pp. 983-994. 2024. https://doi.org/10.52202/078360-0090

4. White A. E. and Lewis H. G. An adaptive strategy for active debris removal. *Advances in Space Research*. Vol. 53, no. 8, pp. 1195–1206, 2014. doi: 10.1016/j.asr.2014.01.021.

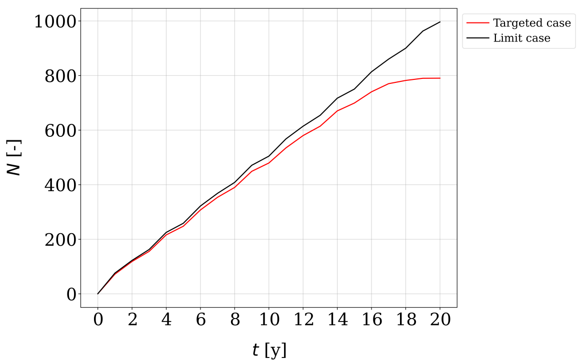
5. Somma G. L. and Supervisors: Lewis H. and - Colombo C. Adaptive remediation of the space debris environment using feedback control. *Ph.D. dissertation*. 2019. Available: https://eprints.soton.ac.uk/447843/.

6. Mia Tian R., Xi K., Lavezzi G., Lifson M., Servadio S., and Linares R. Optimizing active debris removal strategies with feedback control for a sustainable space environment. *In Proceedings: Astrodynamics Specialist Conference*, 2024.

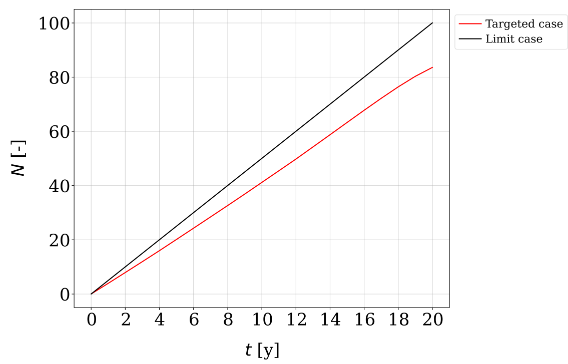
7. Rusconi M., Giudici L., Colombo C., Application of active feedback control for investigation of debris mitigation strategies on a density-based model of the population evolution. *In Proceedings: 75th International Astronautical Congress (IAC)*. Pp. 1905-1917, 2024. https://doi.org/10.52202/078360-0184.

8. King-Hele D. G. Theory of Satellite Orbits in an Atmosphere. *Butterworths Mathematical Texts*. 1964. doi: 10.1002/qj.49709038627.

9. Nekoo S. R. Tutorial and review on the state-dependent riccati equation. *Journal of Applied Nonlinear Dynamics*. Vol. 8, pp. 109–166, Jun. 2019. doi: 10.5890/JAND.2019.06.001.



(a) Yearly evolution of the total number of post-mission disposals performed in all the shells of the domain for the Limit case (black), which is compared to the same evolution for the Targeted case (red).



(b) Yearly evolution of the total number of active debris-removals performed in all the shells of the domain for the Limit case (black), which is compared to the same evolution for the Targeted case (red).

Figure 15. Yearly evolution of the total number PMDs (a) and ADRs (b) performed in all the shell of the domain in the Limit case (black) compared to the Targeted case (red).

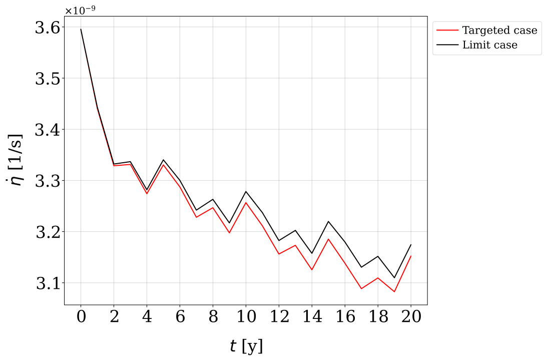
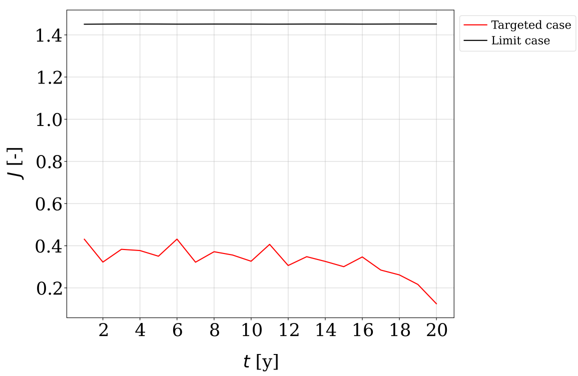
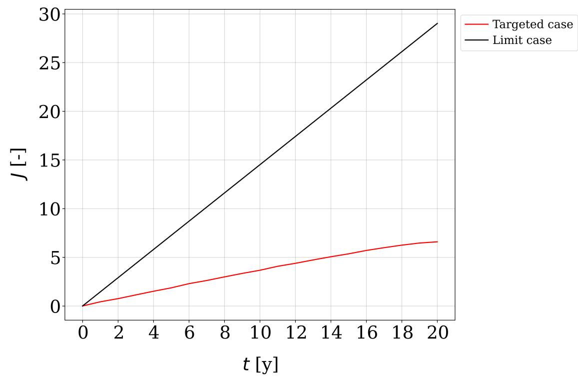


Figure 16. Yearly collision rate evolution computed as the sum of the contributions from Eqs. 11 and 12 over the entire domain for the Limit (black), which is compared to the same evolution for the Targeted case (red).

10. Beeler S. and Cox D. State-dependent Riccati equation regulation of systems with state and control nonlinearities. *NASA Technical Report*. 2004.
11. Korayem M. H. and Nekoo S. R. Finite-time state-dependent riccati equation for time-varying nonaffine systems: Rigid and flexible joint manipulator control. *ISA Transactions*. Vol. 54, pp. 125–144, 2015. doi: 10.1016/j.isatra.2014.06.006.
12. Topputo F., Miani M., and Bernelli-Zazzera F. Optimal Selection of the Coefficient Matrix in State-Dependent Control Methods. *Journal of Guidance, Control, and Dynamics*. Pp. 861-873 pp. 2015. https://doi.org/10.2514/1.G000136.
13. Letizia F., Bastida Virgili B., and Lemmens S. Assessment of orbital capacity thresholds through long-term simulations of the debris environment. *Advances in Space Research*. Vol. 72, no. 7, pp. 2552–2569, 2023. doi: 10.1016/j.asr.2022.06. 010.



(a) Yearly evolution of the value function in Eq. 4 for the Limit case (black), which is compared to the same evolution for the Targeted case (red).



(b) Yearly evolution of the cumulative value function in Eq. 4 for the Limit case (black), which is compared to the same evolution for the Targeted case (red).

Figure 17. Performance metric evolution throughout the years of the Limit (black) and Targeted (red) cases.