INFLUENCE OF MITIGATION PROCEDURES COMBINED WITH ADR ON THE DEBRIS POPULATION EVOLUTION

DI-COSTANZO Romain (1), REVELIN Bruno (2) , DOLADO-PEREZ JC (3),

(1) CS-SI, 5 Rue Brindejonc des Moulinais, 31506 Toulouse (FRANCE), Email: romain.di-costanzo@c-s.fr
(2) CS-SI,, 5 Rue Brindejonc des Moulinais, 31506 Toulouse (FRANCE), Email: bruno.revelin@c-s.fr
(3) CNES, 18 Avenue Edouard Belin 31401 Toulouse (FRANCE), Email: juan-carlos.doladoperez@cnes.fr

ABSTRACT

The unceasing growth of space debris led space agencies to develop prediction tools to simulate in orbit objects evolution. The French space agency (CNES) started in early 2012 to develop its own population evolution tool, MEDEE (Modelling the Evolution of Debris in the Earth Environment). This tool has been developed to realistically simulate the different orbital events contributing to the space objects evolution. Among them, we can name explosions and collisions, natural and planned re-entries, and future launches. MEDEE is also able to simulate some mitigation measures to try to stabilize the future near earth orbit for the next centuries, and de facto the in-orbit population evolution [1]. This paper has a double goal: it first details a new post processing method used to simulate active debris removal (ADR) with various parameters values, without any rerun of time-consuming Monte Carlo simulations usually needed to converge to a reliable statistical result. The validation process is also described. The second goal of this paper is to use this post-processing method so as to apply it to previous MEDEE runs with different objects selection methods. Effects on the in orbit population evolution with different ADR configurations will be analysed.

1 INTRODUCTION

The near Earth debris population is increasing for more than 50 years of space activities. Some recent studies showed that the in-orbit population is instable, and will increase for the next centuries. Even with commonly adopted mitigation measures, such as the 25 years rule, which limits post-mission orbital lifetime of satellites to less than 25-year, it will not be possible to stabilize the low earth orbit, and the number of space debris will continue to increase. Some strategies, like ADR (Active Debris Removal) seems to be unavoidable if we want to preserve future space activities [2] [3].

In this paper, we are describing post-processing procedures applied on a previous set of MEDEE (Modelling the Evolution of Debris in the Earth Environment) runs. Those runs are compliant with the IADC study report test case [3]. The initial population used to compute the results presented on this paper was kindly provided by ESA/ESOC's Space Debris Office.

2 MEDEE

2.1 General principles

MEDEE is a Java-based long-term evolutionary model of space objects in the vicinity of Earth, using the STELA semi analytic propagator [4], and the OREKIT low-level open source space library [5]. MEDEE computes collision risk thanks to the cube algorithm described in [6]. The cube algorithm estimates the long-term collision probabilities by means of uniform sampling of the system in time and can be applied to any kind of orbit. Thus, at a specific snapshot, in every cube containing at least two objects, a collision probability $P_{ij}(t)$ is computed, and a random number is compared to the actual probability to decide if a collision occurred or not. The probability is given by:

$$P_{ij}(t) = s_i s_j \Delta V_{impact} \sigma dU \tag{1}$$

where s_i and s_j are the spatial densities of object i and j in the cube, ΔV_{impact} is the relative impact velocity between the two, σ is the cross sectional area for both objects and dU is the cube volume.

If a collision occurs, debris clouds are generated according to the NASA's Standard Breakup Model [7]. This model is able to generate debris issued from a collision, computing the appropriate distribution in length, area, mass, and ΔV.

2.2 Benchmark scenario

In this study, we are using outputs from 60 MEDEE runs with the IADC scenario test case. The latter assumes a future launch traffic with a repeated 2001-2009 cycle. The commonly adopted mitigation measures with the 25 years rule with a success rate of 90% for spacecraft (S/C) and upper stages (i.e., rocket bodies, R/Bs), and 100% successful passivation (i.e., no future

Proc. '6th European Conference on Space Debris'
Darmstadt, Germany, 22–25 April 2013 (ESA SP-723, August 2013)

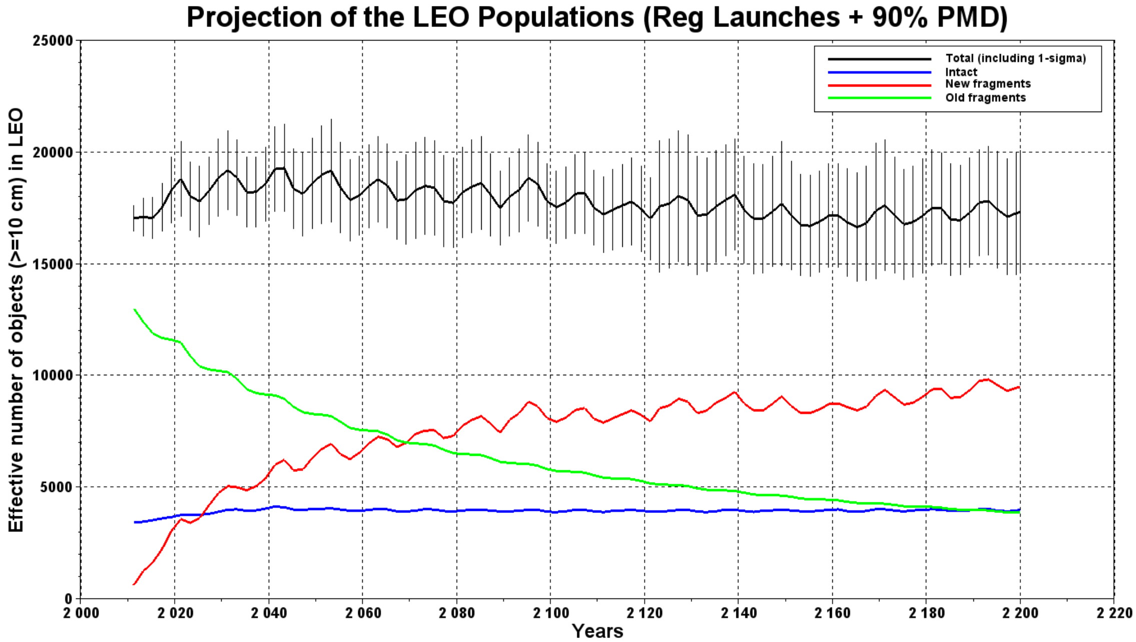

Figure 1: projected LEO population (>10cm), average of 60 MC runs

explosion) were also assumed. Collision avoidance manoeuvres were not allowed, and an 8-years mission lifetime for payloads launched after 1 May 2009 was adopted. The result obtained is plotted in Fig 1. We can observe a relatively constant number of objects in the LEO orbit. Most of the studies conducted until now present a different trend, with a slight increase of the LEO population over the next 200 years. Even if the mean evolution of the overall population given by MEDEE falls within the 1-sigma error region of some of the reference models, it is important to remark that MEDEE does not predict a significant increase on the total number of objects after 200 years. Taken into account that MEDEE developments started in early 2012, further studies are needed to consolidate the results presented in Fig 1. Conclusions and perspectives are discussed in [1].

Nevertheless, the post-processing method presented here can still be applied. Results on ADR should be taken carefully, but the general method should not be affected by our trend slightly lower than previous studies.

2.3 Description of MEDEE outputs

MEDEE provides several outputs used to get exhaustive 3-dimentional data of the temporal evolution of the in-orbit population. MEDEE has three different types of outputs.

The first one consists in recording the objects population at each snapshot. From those outputs, it is possible to analyse the effective population and its composition, like the intact objects, the old fragments, the new generated fragments, etc…

MEDEE also tracks every event happening to each single object during the simulation. Events can be re-entry (natural, or planned by mitigation), collision (partial or complete destruction), debris generation, new launch, etc… Every event is logged with its type, its date, the orbital elements of the associated object at date, and a unique identification. From those data, it is then possible to get all descendants generated from a collision, by creating a tree of colliding objects. The well-known cascade effect, also known as the Kessler syndrome, can thus be tracked, from the incriminated couple of objects generating a collision, to the complete descendants list of debris, themselves sometimes responsible of new collisions. The use of this tree will be the main aspect used to simulate ADR.

Finally, MEDEE logs objects IDs and the probability given by Eq. 1. for each pair present in the same cube. We will use those results to compute the criteria used to select candidates objects for ADR.

Outputs are summarized in Table 1.

In this paper, we will show how those outputs can be used to post process data and simulate ADR from a set of simulations, without any rerun.

Output type	Data logged	Frequency
1. Populations	Whole population	User defined (2 years by default)
2. Events	Modifications at single object level	At event date
3. Cube	Encounter of 2 objects in the same cube	At each snapshot (5 days by default)

Table 1: MEDEE outputs

3 ADR COMPUTATION METHOD

3.1 Method description

MEDEE has the ability to simulate ADR in a single or multi-missions mode. A single mission will remove several objects at a time while the multi-mission is able to remove one object only, but may be repeated at different dates. To simulate ADR and have a reliable statistical result, several tens of time-consuming Monte Carlo (MC) simulations have to be done. This has been one of the motivation to try to re-use previous simulations data to simulate an ADR process instead of performing reruns.

From a simulation point of view, ADR consists in identifying potentially dangerous objects, and remove them at specific frequencies. It is actually possible to achieve this just by using the log file, which tracks every event occurring during each simulation. Indeed, as we are able to track the complete collision-induced descent of a given object, removing this object and its descendants is really straightforward.

The process is the following one:

- ✓ Identify objects to remove from previous runs
- ✓ Apply ADR on each simulation by removing descendants of objects that have been selected.

Key parameters are:

- ✓ Objects selection method
- ✓ Number of ADR per year
- ✓ Starting date to apply ADR

The advantage of this process is obviously its execution time, which allows to test various types of ADR object selection criteria.

3.2 Bias introduced

This method introduces a bias from what can be expected from a real set of Monte-Carlo simulations. Let's consider two colliding objects, A and B. If A is considered dangerous enough to be removed from the population, we will remove A and all its descendants. But in a real simulation, even if the current collision is avoided because A has been removed and cannot collide with B anymore, B could in theory collide with another object C, later in the simulation. This introduces a bias in the simulation, which is directly proportional to the number of collision avoided through the cascade effect. But we consider that this bias will not compromise the overall statistical results and the general trend of the in-orbit objects evolution, as object B will not collide with A in every simulations.

3.3 ADR effectiveness

To evaluate the effect of our ADR method, and for comparison purposes, two metrics are used [8].

First, the Effective Reduction Factor (ERF) quantifies the average number of LEO objects removed from the total population, for each object removed by ADR. So:

ERF=[total nb of LEO objects reduced as of 2200]/[total nb of objects removed by ADR as of 2200]

Second, the Collision Reduction Factor (CRF) quantifies the average number of objects removed by ADR necessary to avoid one collision. Thus:

CRF=[total nb of objects removed by ADR as of 2200]/[total nb of reduced collision as of 2200]

3.4 Post processing speed

An important point to underline is the computation time difference between a classical MC simulation and a post-processing run over a set of MC. A single MC run of MEDEE, with the classical benchmark scenario, is almost 4 days long, on a 12-cores machine. Whereas an ADR post-processing over 60 MC simulations is half an hour total, on a desktop computer.

4 METHOD VALIDATION

To validate this post-processing method, we compare a first set of 60 no ADR simulations + post-processing, and a second set of 30 ADR simulations. Three objects are removed per year, from 2020 to 2200.

First, post-processing is done with criteria (2.1). One of the outputs is a list of criteria-ordered objects which is used to post-process every no-ADR simulations. Secondly, we use this list for the ADR simulations, with a very basic scheme: during each ADR simulation, the next 3 objects of the list are removed from the objects population at the beginning of each year. If an object has already disappeared from the population at this date, the next object in line is used. Thus, the same number of objects is removed from each set of simulations (exactly 540, from year 2020 to year 2200).

For comparison purposes, ERF and CRF are presented in Table 2. For direct ADR simulations, total number of reduced objects is deduced by subtracting the ADR simulations average total population in 2200 to the no-ADR simulations average total population in 2200. The same process is used for the counting of the reduced number of collisions.

	ADR post-processing	Direct ADR simulations
Nb objects removed by ADR	540	540
Nb obects reduced	4372	3851
ERF	8.1	7.1
Nb of collisions avoided	16.5	14.0
CRF	32.7	38.0

Table 2: ERF and CRF for ADR post-processing and direct ADR

There is a 15% discrepancy between the two columns for both factors, logically in opposite ways (ERF should increase as CRF decreases). Nevertheless, we consider our process validated, but keeping the previously discussed bias (3.2) in mind. However, this bias should not hinder comparisons between different criteria.

5 SELECTING OBJECTS FOR ADR

We will now focus on the selection of objects for the ADR process.

5.1 Assumptions for objects selection

We are here only considering intact objects present in the initial population. Intact objects are Rocket Bodies (R/Bs), Spacecrafts (S/Cs) and mission related objects (MRO). We only consider intact objects because in orbit debris represent just about 3% of the total mass, the remaining 97% being in R/Bs or S/Cs. We only consider objects present in the initial population because simulations, from which the post processing has been computed, simulate a post mission disposal for every new R/Bs and S/Cs injected in orbit, with a 90% success rate. A quick analysis of the 60 MC runs comforts this assumption: Table 3 relates the repartition of catastrophic collisions for every possible pairs of object origins.

Object origins	Mean number of catastrophic collisions / run
Old intact / Old debris	5.2
Old intact / New intact	4.7
Old intact / Old intact	4.2
Old Debris / New intact	3.1
New intact / New intact	2.3
Old intact / New debris	1.9
Old debris / Old debris	1.6
New intact / New debris	1.4
Old Debris / New debris	1.2
New Debris / New Debris	0.7

Table 3: collision distribution (object origin)

"Old" refers to objects present in the initial population, and "New" refers to objects added later to the population (new launches, debris generated by a collision etc…). Most objects involved in catastrophic collisions appear to come from the initial population.

Another way to comfort the idea of selecting only intact objects can be seen through Table 4:

Object types	Mean number of catastrophic collisions / run
Payload / Debris	6.0
R/B / Debris	5.0
R/B / Payload	4.3
Payload / Payload	3.6
Debris / Debris	3.5
R/B / R/B	1.4
Payload / MRO	1.1
R/B / MRO	0.7
Debris / MRO	0.7
MRO / MRO	0.1

Table 4: collision distribution (object type)

Intact objects are mainly involved in catastrophic collisions. However, this selection is not sufficient and some more restrictive criteria must be found.

5.2 Criteria selection

We are searching here for ADR candidates that would generate more debris in case of a collision. The first important parameter is the object's mass. Colliding objects generate a number of debris directly proportional to the mass of both objects [7]. The collision probability also has a great importance in object selection. Probability (Eq. 1) depends on relative velocity and objects' size (cross section).

Thus, for a given object with an instantaneous collision probability $P(t)$ and a mass m, the following different criteria can be computed:

$$R_1 = m\sum_S \sum_T P(t) \tag{2.1}$$

$$R_2 = m\sum_S \sum_{\Delta t} P(t) \tag{2.2}$$

$$R_3 = m\sum_S \sum_{\Delta t} k_Z P(t) \tag{2.3}$$

$$R_4 = mMax_{ST}(P(t)) \tag{2.4}$$

Where *S* is the whole set of simulations, *T* is the whole time span of the simulation (2009 to 2200), and Δt a given period of time from the current date (typically 2 years).

In (2.3) a region dependent weight k_Z is added to the criterion. Regions are pre-defined by inclination intervals and altitude intervals. The goal is to focus on areas where most of the collisions actually occur.

All the above criteria are applied over the whole set of simulations, and none are simulation-specific. The reason for this is to avoid as much as possible of the bias discussed in 3.2: when ADR objects are selected in a single simulation, and applied to this same simulation, too many collisions are avoided because in our process each object's counterpart in the avoided collision has no chance to participate in a later collision with a third object. Whereas if ADR objects are selected over the whole set of simulations, the bias is statistically diluted over the whole set.

We also restrict our selection to non-GTO objects, as they spend just a fraction of time in the LEO orbit. It thus seems neither realistic nor efficient to plan an ADR mission on such objects. Those criteria are in accordance with previous studies [8][2].

5.3 Objects selection

To select object with the criteria defined above, we analyse each output of type 2 (see Table 1) for each simulation. As those outputs log the necessary data for every pair of objects identified in the same cube, including the associated probability of collision, the above criteria (2.X) are easily computed.

Figure 2 displays as a density map the mean number of collisions per simulation occurring in 1-degree bins in inclination and in 50 km bins in altitude. Some areas have a higher collision rate than others, particularly the crowded sun-synchronous orbits.

Objects selected by the ADR process are plotted on the same density map. A (+) represents a perigee, a (X) an apogee. Time of removal is displayed both by size and color of marks: the bigger (and hotter) the mark is, the earlier the object will be removed (by 10 years wide intervals). For this example, our validation case is used (criterion 2.1, 3 ADR/year,…).

Areas with a high collision risk are quite well defined by the criterion. More importantly, the first objects removed will be in critical areas A (i∈[81°, 84°] / alt∈[700km, 1000km]) and B (i∈[96°, 100°] /

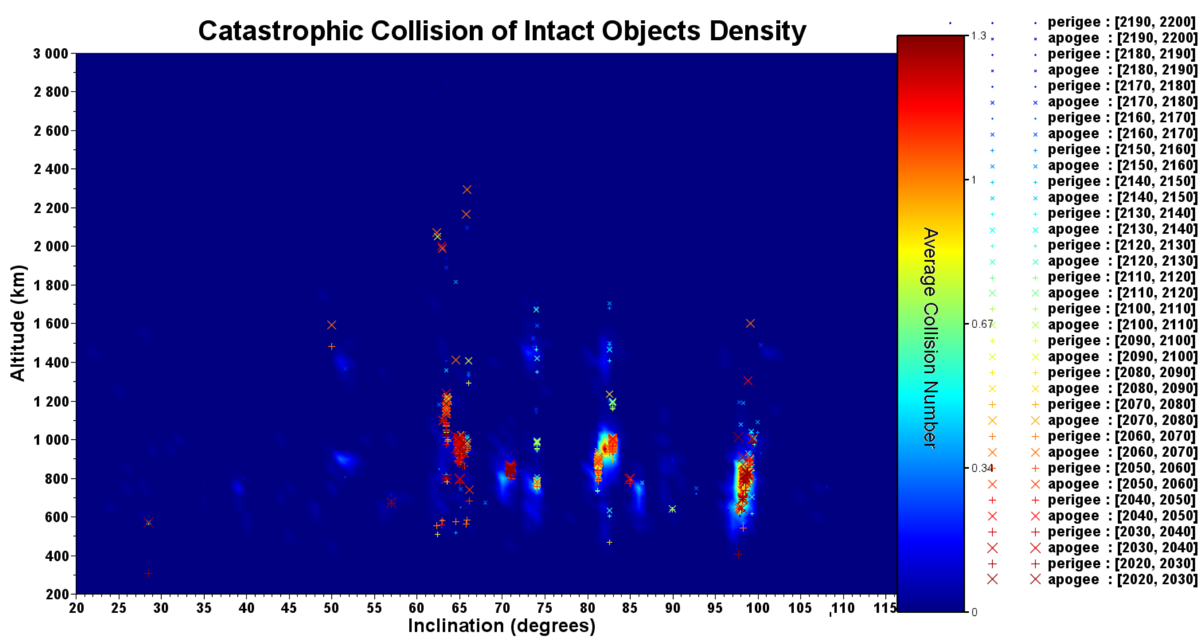

Figure 2: average collision density, and ADR selected objects (average over 60 MC simulations). 3 ADR / years, 2020 to 2200.

alt∈[600km, 900km]).

However, area C around i∈[60°, 76°] / alt∈[600km, 1200km] seems over-represented with criterion 2.1: a lot of objects are removed in the first tens of years, despite the fact that the mean number of collisions occurring in this area seems weak.

6 EFFECT OF ADR ON THE POPULATION EVOLUTION, CRITERIA BENCHMARK

Thanks to the post-processing speed, we are able to estimate the efficiency for various possible criteria.. Table 5 and Table 6 summarize the results in terms of ERF and ECF.

Criterion	2.1			2.4		
Nb ADR/Year	3	5	7	3	5	7
Nb objects removed by ADR	540	900	1260	540	900	1260
Nb obects reduced	4372	5483	6277	4466	5377	6186
ERF	**8.1**	**6.1**	**5.0**	**8.3**	**6**	**4.9**
Nb of collisions avoided	16.5	22.7	25.9	15.5	20.5	24.4
CRF	**32.7**	**39.6**	**48.6**	**34.8**	**43.9**	**51.6**

Table 5: 3/5/7 ADR per year, with criterion 2.1 and 2.4

First, as is noted in [2] and [8], effectiveness of ADR decreases as the number of removed objects increases: first objects are more important than the next ones. Therefore it proves that criterion based on Mass*Probability are effective.

The main conclusion from Table 5 is that criteria based on cumulative probability are somewhat better than 2.4 based on max probability. Cumulative probability gives better priority to objects repeatedly involved in potential hazard, in the same simulation or over several different runs.

In Table 6, criterion 2.3 uses k_Z=3 for areas A (i∈[81°, 84°] / alt∈[500km, 1300km]) and B (i∈[96°, 100°] / alt∈[500km, 1100km]). k_Z=3 for area C around i∈[60°, 76°] / alt∈[400km, 1500km] (k_Z=1 for everywhere else). Those areas have a wider range in altitude than zones described in 5.3, to consider slightly non-circular orbits in the zone selection process. Different weight values were tested. Moreover, selection of each year's set of objects is done in the same area, to better reflect real ADR conditions of single multi-ADR missions: each year, the 3 ADR objects are selected in the same area, to minimize propulsion needs and mission costs. However, such a logistic constraint is expected to reduce blunt ADR efficiency, as for a given selection chosen objects will not all necessary be firsts in our criterion order.

Both columns of Table 6 display better results than Table 5: limiting our probability gathering to a shorter but immediate time interval (2 years in our tests) is more efficient than taking the whole time span into account. Potential collisions are targeted more accurately this way. And last, figure 3 shows the effect of post-processing ADR (with the above criterion 2.2) over the population evolution. Reduction of new fragments (red lines) is the main effect of ADR performed over intact objects (blue line). Our result here is a stabilization of new LEO fragments around 5000 objects.

Actually this discrepancy is compensated because giving more precedence to specific areas induces some advantage over the more straightforward computation by selecting more crowded areas first.

Criterion	2.2 (Δt=2 years)	2.3 (Δt=2 years)
Nb objects removed by ADR	540	540
Nb obects reduced	4703	4672
ERF	**8.7**	**8.7**
Nb of collisions avoided	18	19
CRF	**29.9**	**28.3**

Table 6: 3 ADR per year, with criterion 2.2 and 2.3

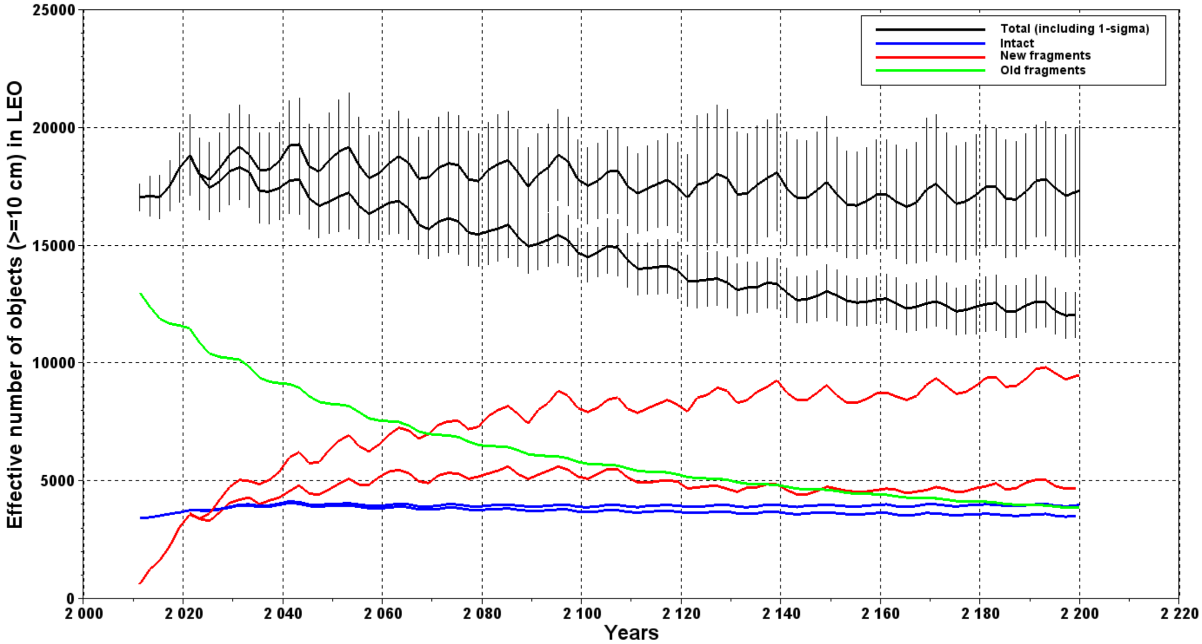

Figure 3: LEO population evolution (>10cm), no ADR vs 3 ADR/year. Criterion 2.2. Δt=2 years

7 CONCLUDING REMARKS

ADR can actually be performed in many different ways, with different strategies depending on mission type, costs, technical solutions, etc… Our post-processing based ADR simulation provides a fast and easy means to test and compare different criteria for the selection of ADR objects, and to optimize them for better effectiveness in terms of population reduction and collision reduction.

A bias is systematically introduced with our post-processing ADR method. Future studies will focus on evaluating it more accurately. As stated in 3.2 our present computation is slightly optimistic, as counterpart objects in avoided collisions cannot be reintroduced in the simulations, and possibly be part in later hypothetical collisions. A pessimistic computation would take those counterpart objects into account as ADR objects: if a collision at t_1 between objects A and B is avoided because A is selected for an ADR at $t_0<t_1$, then B may be automatically introduced in the ADR selection at t_1. The bias would then be compensated, even if a new one, hopefully smaller, would be introduced, as object B would not have been selected for ADR otherwise.

Nevertheless, our results tend to demonstrate that a dynamical selection of objects along time is more efficient than *a priori* selection over the 200 years time span.

We have also shown that the efficiency of ADR is reduced when the number of eliminated object increases. This is easily understandable, as the first eliminated objects are posing higher threat to the environment.

The post-processing method shown on this paper shows a fairly good accordance when compared with ADR results computed from Monte Carlo simulations. This means that our approach offers a good alternative to intensive computational simulations when we are interested in having a fast insight on the quantification of the effectiveness of different ADR scenario to control of orbital population in LEO.

8 ACKNOWLEDGEMENTS

The space debris initial population used to compute the results presented on this paper was kindly provided by ESA/ESOC's Space Debris Office.

9 REFERENCES

1. Dolado-Perez J.-C, Di-Costanzo R., Revelin B., Introducing MEDEE – A new orbital debris evolutionary model.

2. Liou J.-C., Johnson N, Hill N.M., Controlling the Growth of Future LEO debris Populations with

Active Debris Removal, *Acta Astronautica*, 66, 648-653, (2010).

3. Report of IADC Study, Stability of the Future Leo Environment, 50th Session of the Scientific and Technical Subcommittee – UN COPUOS. 11 – 22 February 2013.

4. Fraysse, H., Morand, V., Le Fevre, C., Cauhert, A., Lamy, A., Mercier, P., Dental, C., and Deleflie F., “STELA a Tool for Long Term Orbit Propagator,” Proceedings of the 5th International Conference on Astrodynamics, 29 May – 1 June 2012, ESA/ESTEC, Netherlands.

5. OREKIT – www.orekit.org

6. Liou J.-C, Collision activities in the future debris environment, *Adv. Space. Res*, 38 (2008), 2102-2106

7. Johnson N.L, Krisko P.H., Liou J.-C., Anz-Meador P.D., NASA’s new Breakup Model of EVOLVE 4.0, *Advances in Space Research*, 28(9), 2001, 1377–1384

8. Liou J.-C., Johnson N, A sensitivity study of the effectiveness of active debris removal in LEO, *Acta Astronautica*, 64, 236-243, (2009).