CHARACTERISING IN-ORBIT FRAGMENTATIONS WITH THE PUZZLE SOFTWARE

Matteo Romano[(1)], Andrea Muciaccia[(1)], Mirko Trisolini[(1)], Pierluigi Di Lizia[(1)], Camilla Colombo[(1)], Alessandra Di Cecco[(2)], Luca Salotti[(2)]

[(1)] *Politecnico di Milano, Via Giuseppe La Masa 34, 20156, Milano, Italy, Email: {matteo1.romano,andrea.muciaccia,mirko.trisolini,pierluigi.dilizia,camilla.colombo}@polimi.it*
[(2)] *Agenzia Spaziale Italiana (ASI), Via del Politecnico snc, 00133, Rome, Italy, Email: {alessandra.dicecco,luca.salotti}@asi.it*

ABSTRACT

As in-orbit brekup events become more frequent and more difficult to predict and avoid, the risk of collisions of debris with active satellites grows each year, threatening the safety of space operations. By constantly monitoring known and new objects, it is possible to verify whether they were created in a fragmentation. The PUZZLE software package was developed at Politecnico di Milano, under a contract with the Italian Space Agency (ASI), with the objectives of identifying which unclassified objects originated via a collision or explosion, and characterising the event in terms of mass and energy involved. The proposed approach focuses on the evolution of the osculating orbital elements of a large set of objects to identify common aspects of their motion, with pruning and clustering algorithms used to identify the epoch of a fragmentation and which known objects were involved.

1 INTRODUCTION

As the reliance of society on space-based services keeps growing, the number of launches into space and of satellites orbiting the Earth grows as well, leading to more frequent fragmentation events occurring each year. This leads, in turn, to an increasing probability of collisions and production of new debris [1]. While constant surveillance can help in anticipating and avoiding collisions, not all events can be predicted (e.g. explosions of rocket bodies or dismissed satellites) or avoided (e.g. collisions between objects in orbit). Debris produced by these events must be identified as soon as possible, to detect breakups and reduce the risk they pose for other satellites in the future in a reliable and efficient manner.

Existing works aimed at fragmentation detection tackled the issue by studying the orbital dynamics of the debris from different pints of view to estimate the time and place of the event and identify the parent objects.

Andrisan et al. [2] developed the Simulation of On-Orbit Fragmentation Tool (SOFT) with the goal of characterising recent fragmentations based on the detection of new debris by the Space Surveillance and Tracking (SST) network. This tool determines the type of fragmentation based on the number and the distribution of the detected fragments (with collisions presenting a higher number of fragments in larger distributions of orbital parameters than explosions), while it estimates the epoch and position of the breakup by studying the average distance between the objects in the debris cloud. Finally, the breakup is modelled based on the identification of the parent objects and compared with the observations to remove unrelated objects from the set.

Frey et al. [3] [4], instead, focus on the long term evolution of orbits (over periods of time of the order of years) to search for past fragmentations by considering an averaged dynamics with mean Keplerian elements and a semi-analytical propagation. Their method determines the epoch of the breakup by detecting a convergence of objects in the space of inclination and right ascension of the ascending node (RAAN) in low Earth orbit (LEO), as the objects involved since debris are likely to present similar orbital planes in the proximity of the fragmentation.

On a different note, Dimare et al. [5] identify fragmentations by defining a similarity function of the orbital elements of the observed objects. The epoch of the event and the parent objects involved in it are found by locating the minimum of the similarity function among the various objects, both on the short and the long term but under the assumption that a fragmentation had occurred. The D-criterion proposed by Southworth and Hawkins [6] was determined as the most suitable metric for this kind of problem.

This work introduces the PUZZLE software package, developed at Politecnico di Milano with the objectives of identifying which objects, inside a set obtained from a catalogue, originated via a breakup, and of characterising the event in terms of mass and energy, by modelling the distribution of the debris cloud after identifying the objects involved.

The proposed approach considers a set of unclassified objects in the form of Two-Line Element (TLE) data taken from a daily updated catalogue, with no a priori assumption about the recent occurrence of breakups, trying to determine whether one happened by analysing the evolution of the orbits backwards in time. Pruning and clustering algorithms are employed to identify possible orbital intersection that make close encounters between objects possible, removing unrelated ones and identifying those whose positions converge in the same region at the same time. These are then matched with a catalogue of known objects to provide a guess of the possible operative or dismissed spacecraft involved in the explosion or collision. The fragmentation is then modelled using the available NASA standard breakup model, which provides distributions of area-to-mass ratio and relative velocity of the fragments useful to identify the orbital regions at risk of possible collisions in the future.

The tool described above was developed under a contract with the Italian Space Agency (ASI) as part of a more general software for the support of SST services and the study of space debris. In Section 2 its general architecture will be explained with attention to the operations performed within each module. Then, in Section 3, the application of the software to actual fragmentation events will be shown alongside numerical results and performance data. Section 4 will summarise and comment the main results.

2 SOFTWARE ARCHITECTURE

This section describes the general architecture of the PUZZLE software.

The software takes a set of TLE data as input with the goal of detecting and characterising possible in-orbit fragmentations occurred in the near past, on time scales of the order of a few days up to a month. For this reason, the TLEs are taken from a catalogue updated daily, in which the estimated orbital states of unclassified objects are correlated to identify a breakup event. Several are the other parameters used to perform this analysis:

- the length of the interval in which the search is done, of the order of a few days;
- the distance and time thresholds used to detect a close encounter between two objects;
- the parameters used to define whether two objects have similar orbits or not;
- the parameters useful to model the fragmentation (if any is found) and estimate the the number and characteristics of the fragments likely generated in the event.

The analysis is divided into two parts: in the first one, the input data is studied searching for a possible breakup, trying to identify which bodies were involved in it, the fragments produced in the event the parents, and when it occurred; in the second (optional) part, the fragmentation, if any is detected, is modelled with the goal of estimating the total number of fragments that were produced, as well as their distribution of physical characteristics and of orbital parameters. This second analysis provides useful knowledge to facilitate the detection of the largest fragments following the vent, and to estimate the risk of collisions between known objects (among which are operative satellites) and the new fragments in the regions that are the most affected by the recent breakup.

The software is composed of a series of five distinct modules, each performing a specific task, as represented in Figure 1:

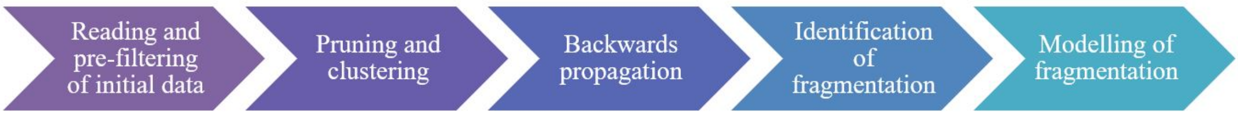


Figure 1: Block diagram of the software architecture.

Module 1: reading and pre-filtering

The first module reads the input data and pre-processes it with the goal of removing those TLEs with possible outlier values of orbital elements from the pool. These values are treated as errors produced in the initial orbit determination process, and might reduce the accuracy of the analysis in the following phases. A diagram of this first block of operations is shown in Figure 2.

The module reads the input TLE data and selects a subset for the analysis as specified by the user. Then, for each series of TLEs corresponding to the same Satellite Catalogue Number, the filtering process proposed by Lidtke et al. [7] is applied to remove possible outliers, following five successive steps:

1. removing the TLEs corresponding to a correction of the immediately previous element, according to a minimum threshold of the update time between two subsequent TLEs;
2. identifying large gaps between TLEs to define time windows in which outliers will be searched;
3. removing the TLEs with values of mean motion that are not coherent within the same temporal window, using a sliding window approach;

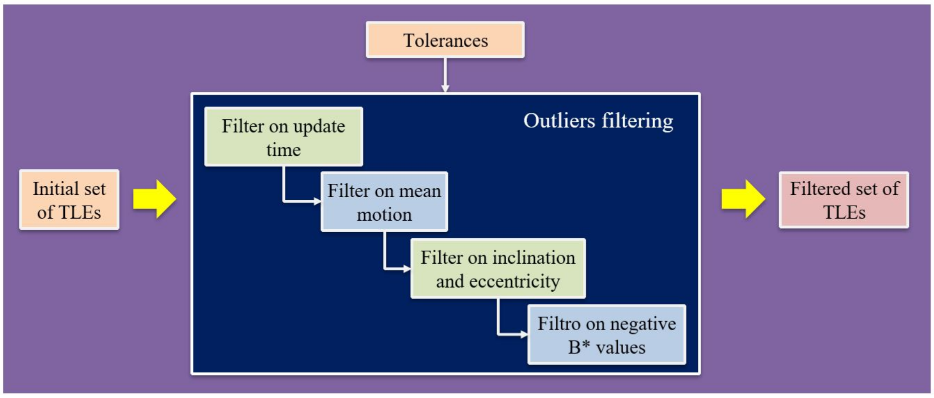


Figure 2: Block 1: module containing the routines for reading and pre-filtering the initial data.

4. removing the TLEs with values of inclination that are not coherent;
5. removing the TLEs with values of eccentricity that are not coherent;
6. removing the TLEs with negative values of B* drag term.

Module 2: pruning and clustering

The second module searches for possible orbital intersections between the objects described by the TLEs with the goal of excluding the ones that cannot have close encounters with each other due to the geometry of their orbits. The method chosen for this purpose is the triple-loop filter proposed by Hoots et al. [8], as represented in the block diagram in Figure 3.

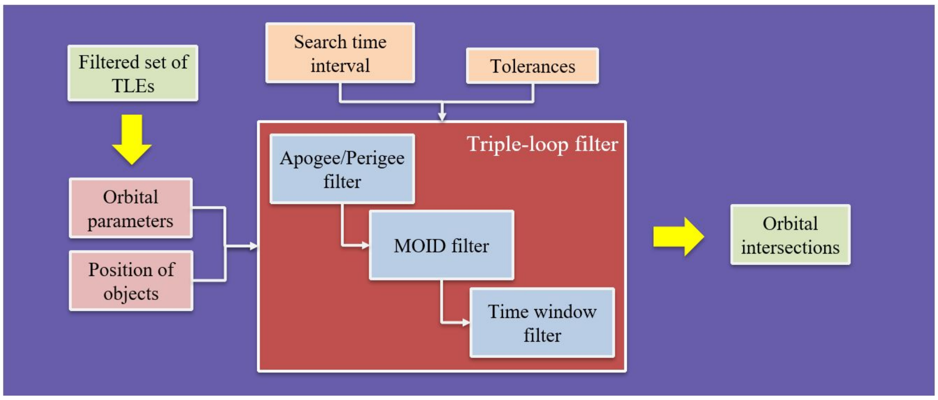


Figure 3: Block 2: module containing the routines for the pruning and the formation of families among the objects being analysed.

Each couple of orbits contained in the TLE set is compared against each other through three filters working in series, of which two are based on orbital geometry and one is based on time.

The first geometrical filter compares the heights of the apogee and perigee of the two orbits to estimate if their relative geometry allows close approaches between the two objects; the quantities

$q = max(r_{P,1}, r_{P,2})$ and $Q = min(r_{A,1}, r_{A,2})$ are defined, as shown in Figure 4, and then compared against a given threshold to defined whether the two orbits pass the filter:

$$q - Q \leq \Delta \tag{1}$$

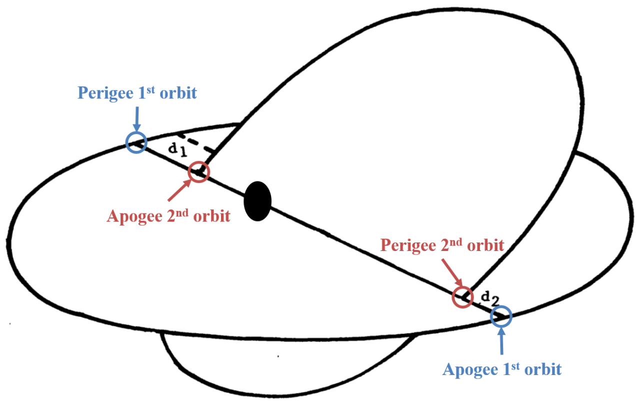


Figure 4: First step of the triple-loop filter: comparison between perigees and apogees of the two orbits. Image modified from [8].

The second geometrical filter checks whether the minimum orbital intersection distance (MOID) between the two orbits is below a given threshold. The MOID is computed here using the algebraic method proposed by Gronchi [9]. While the MOID does not correspond to the actual minimum distance reached by the two objects moving along the orbits, it is the minimum possible distance between the two orbits; thus, when it is too large, no close encounter can occur between the objects.

The third and last filter defines angular windows around the positions of the MOID along the two orbits to check whether it is possible for the two objects to be in proximity of the MOID at the same time within a selected time period. As shown in Figure 5, an aperture angle u_R is defined around the position of the MOID between the orbits. The angular windows are converted to time windows using Kepler's equation, and, by adding multiples of the periods of both orbits to the endpoints of each window, a sequence of time windows are defined throughout the interval set for the search of the fragmentation. The windows are then cross matched for possible overlaps: if at least two intervals along the orbits overlap, the two objects are able to experience a close encounter within the specified time frame; otherwise, no close encounter is possible as their orbital motion is out of phase.

If both objects satisfy the three filters, a close encounter is possible within the search time interval; otherwise, the TLEs representing them are removed from the set.

After pruning the set of TLEs from the objects incapable of having close encounters between each other, the actual encounter distance is computed in order to identify the closest approach within the research window: this will be treated as the breakup event for the two objects, under the assumption that they share a common origin (which will be verified in a later phase). For each overlapping time window, the time when the minimum distance between the two objects is reached is computed via an iterative process, whose starting point is set as the midpoint of the overlap.

With reference to Figure 6, $\mathbf{r}_1(t)$ and $\mathbf{r}_2(t)$ are defined as the position vectors of the two objects at time t, $\mathbf{v}_1(t)$ and $\mathbf{v}_2(t)$ as the velocity vectors, and $\mathbf{a}_1(t)$ and $\mathbf{a}_2(t)$ as the accelerations acting on the

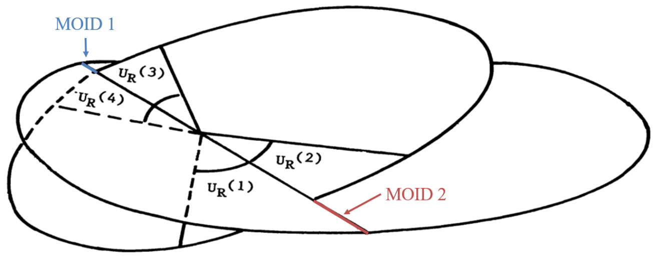


Figure 5: Third step of the triple-loop filter: definition of the angular windows around the MOID. Image modified from [8].

two. The square of the relative distance between the two objects is then defined, using the rule of cosines, as:

$$\begin{aligned} d^2(t) &= r_1(t)^2 + r_2(t)^2 - 2\left(\mathbf{r}_1(t)\cdot\mathbf{r}_2(t)\right) \\ &= \left(\mathbf{r}_1(t)\cdot\mathbf{r}_1(t)\right) + \left(\mathbf{r}_2(t)\cdot\mathbf{r}_2(t)\right) - 2\left(\mathbf{r}_1(t)\cdot\mathbf{r}_2(t)\right) \end{aligned} \tag{2}$$

To search for the minimum of $d^2(t)$ means to search for the zeros of its derivative R, defined as:

$$R = \frac{\mathrm{d}d^2}{\mathrm{d}t} = (\mathbf{r}_1\cdot\mathbf{v}_1) + (\mathbf{r}_2\cdot\mathbf{v}_2) - (\mathbf{v}_1\cdot\mathbf{r}_2) - (\mathbf{r}_1\cdot\mathbf{v}_2) \tag{3}$$

where the dependence on the time t is left implicit for the sake of simplicity.

Newton's iterations are used to search for the zeros of the function R defined in (3):

$$t_{i+1} = t_i - \frac{R}{\dot{R}} \tag{4}$$

where

$$\dot{R} = v_1^2 + (\mathbf{r}_1\cdot\mathbf{a}_1) + v_2^2 + (\mathbf{r}_2\cdot\mathbf{a}_2) - (\mathbf{a}_1\cdot\mathbf{r}_2) - 2\left(\mathbf{v}_1\cdot\mathbf{v}_2\right) - (\mathbf{r}_1\cdot\mathbf{a}_2) \tag{5}$$

Module 3: propagation

The third module propagates backwards in time the objects that might have a close encounter in the near past, trying to identify a possible convergence of the corresponding orbits. A diagram of these operations is shown in Figure 7.

The analytical SGP4 (Standard General Perturbations 4) model [10] [11] is used to propagate the sets of TLEs, since the information contained in a TLE is a set of averaged orbital elements that are specific to the SGP4 propagator. It considers secular and periodic variations due to Earth's oblateness, solar and lunar gravitational effects, gravitational resonance effects, and orbital decay using a drag model. The SGP4 propagator generates ephemeris in the True Equator Mean Equinox (TEME) coordinate system based on the epoch of the specified TLE. Due to the simplifications introduced by the analytical modelling of the perturbations, the accuracy of the propagation is generally limited to

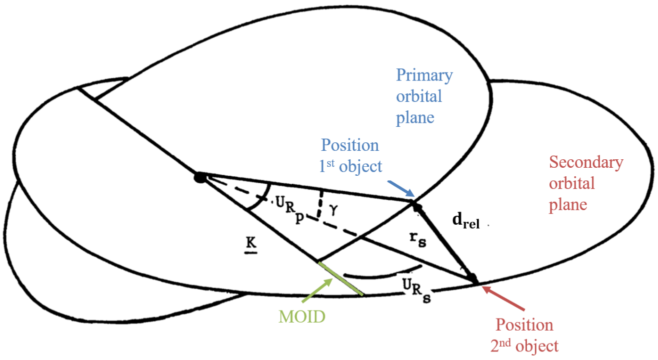


Figure 6: Computation of the global minimum encounter distance between a couple of objects inside each time window around the MOID between the orbits. Image modified from [8].

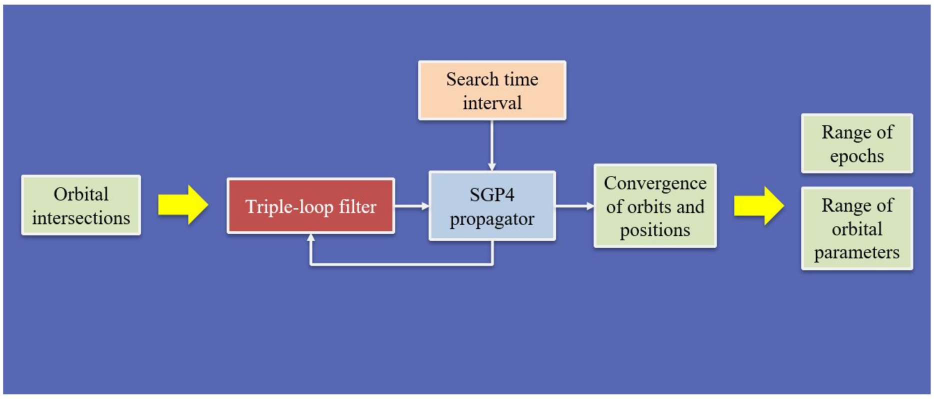


Figure 7: Block 3: module containing the routines for the propagation and the search for a convergence of the orbits being propagated.

intervals of the order of a few days [12]. For this reason, the software limits the search for possible fragmentations to a maximum of 14 days. Future work will focus on the extension of the fragmentation search to longer time scales (of the order of months or years) using a semi-analytical formulation and averaged orbital elements.

This module uses the triple-loop filter to identify the possible windows where intersections between any couple of orbits are possible and estimate the minimum distance reached by the corresponding bodies. Here the SGP4 propagator is used to include the effects of perturbations in the evolution of the orbits: each couple of objects is propagated to the next encounter window to re-compute the approach windows and distances between them, repeating the process until the end of the research time frame. A global minimum distance is then estimated for the couple.

Module 4: fragmentation search

The fourth module identifies the possible objects involved in the fragmentation as well as the probable parents. A scheme of it represented in Figure 8.

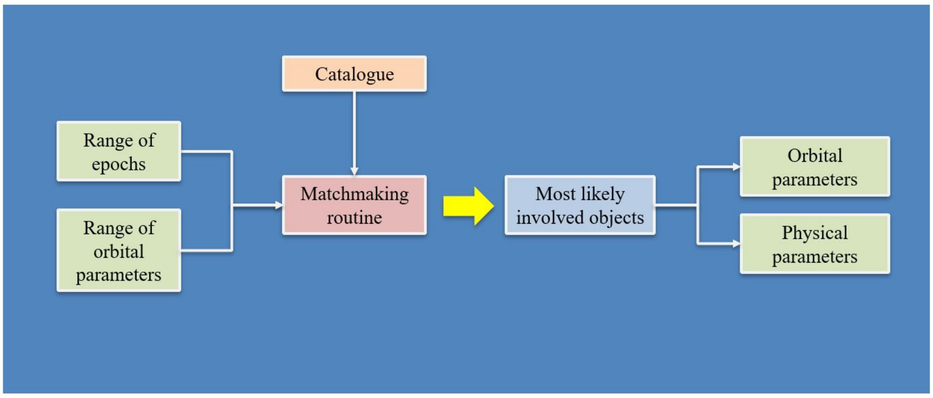


Figure 8: Block 4: module containing the routines for the identification of the fragmentation epoch and of the involved objects.

First, a time window is identified around the possible epoch of the breakup event: using the results of the previous phase, the time interval of the research is divided in bins and the one presenting the most close encounters is selected as the possible epoch, under the assumptions that a high enough number of fragments from the breakup is present in the considered TLE set, and that they all have close encounters around the epoch of the event within specified time and distance margins.

Second, clusters are formed based on the orbital parameters of the objects found in the selected interval around the possible epoch of the event. The goal of this phase is to search for a convergence of object based on the similarity of their orbits, which makes them likely to share a common origin. The clusters are defined by following the single-linkage hierarchical clustering method initially proposed by Zappala et al. [13] for the definition of asteroid families.

This method defines a similarity distance function δv as a metric to measure the separation of the various objects in the space of orbital elements. In this work, the definition of the similarity distance proposed by Zappala et al. is modified to include all orbital elements: semi-major axis a, eccentricity e, inclination i, RAAN Ω, and the argument of periapsis (AoP) ω. This choice was done to account for the wider range of values of that these two parameters have when considering Earth orbiting objects compared to asteroids in deep space. The similarity distance is, thus, defined here as

$$\delta v = na\sqrt{k_1\left(\frac{\delta a}{a}\right)^2 + k_2\left(\delta e\right)^2 + k_3\left(\delta i\right)^2 + k_4\left(\delta \Omega\right)^2 + k_5\left(\delta \omega\right)^2} \tag{6}$$

where n is the mean motion, and the k_i are weights associated to each difference of orbital parameters. Similarly to the definition proposed by Zappala et al., δv has the dimensions of a velocity increment, with the underlying idea that the similarity between two orbits is related to a deviation in velocity generated by disturbances.

Finally, for each cluster, the physical distance between each object in it is computed at the time identified as the possible epoch of the fragmentation. The objects presenting the lowest average distance are the ones selected as possible objects involved in the fragmentation, since their vicinity likely represent an epoch close to the actual fragmentation. The positions of the objects so identified are then compared with the ones of known objects (that is, with a Satellite Catalog Number and a Classification) to identify the possible parent objects and obtain their physical properties and orbital information.

Module 5: fragmentation modelling

The fifth module uses the results of the fragmentation search and estimates the distribution of the possible fragments generated in the event, according to the orbital states of the parents and the dynamics of the event. A scheme of the module is shown in Figure 9.

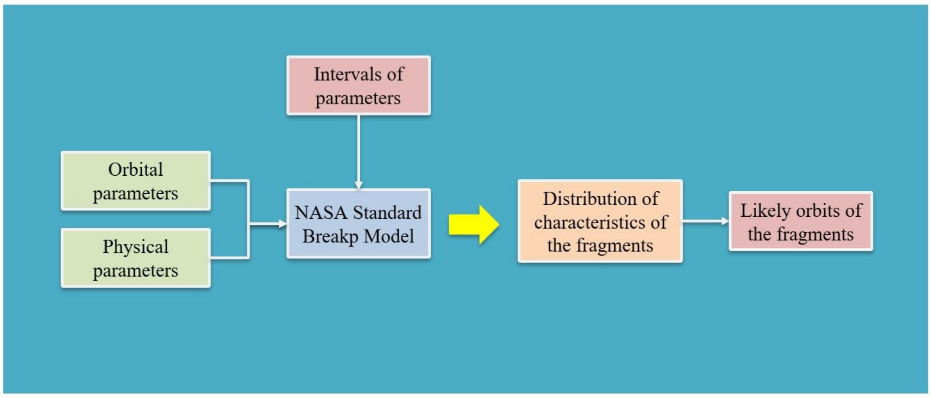


Figure 9: Block 5: module containing the routines for the modelling of the fragmentation using the NASA Standard Breakup Model.

The current NASA Standard Breakup Model [14] [15] is used to characterise the fragmentation and to provide an estimation of the number of fragments formed in the fragmentation, as well of the distribution of their physical attributes (i.e. size, mass, relative velocity) based on the type of the fragmentation (whether a collision or an explosion), the type of object(s) involved (whether payload or rocket body), the total mass involved in the event, and, possibly, the collision speed. For a detailed explanation on how the statistical distributions are defined, the reader is referred to the references [14] [15].

3 APPLICATION

As a validation, the software is applied to the detection of the collision the Iridium 33 and Cosmos 2251 communication satellites occurred on 10th February 2009. The TLEs of the fragments are taken from a daily catalogue dating back to some days after the event, and analysed together with the ones of random objects detected on that day. The accuracy of the analysis is evaluated by judging the correct identification of the fragmentation epoch, of the parent object(s), and of the number of involved objects that were included in the TLE set, while the efficiency is measured via the computational time[1]. The main input conditions and results are reported in Table 1.

The initial set contains 2000 TLEs (23 of which referring to the two satellites and their 19 detected fragments) dating to 17th February 2009, 7 days after the event. The fragmentation is searched within the 10 days prior to the generation of the TLEs. Figure 10 shows the orbits of the TLE set as it is processed during the analysis from the beginning in (a), to the results of the triple-loop filtering in (b), to the identification of those clusters of orbits with presenting close encounters in proximity of the possible epoch of the event in (c). It is to be noted that, while they show only the LEO region, the initial set of 2000 TLEs actually contained objects in all kinds of orbits, ranging from LEO to GEO.

Figure 11 shows how the fragments generated in the collision are distributed according to the NASA breakup model: Figure 11 (a) shows the Gabbard diagram to highlight the change in orbital period and in the perigee and apogee of the orbits with respect to the orbits of the parent objects, while Figure 11 (b) show the distribution of physical characteristics of the fragments.

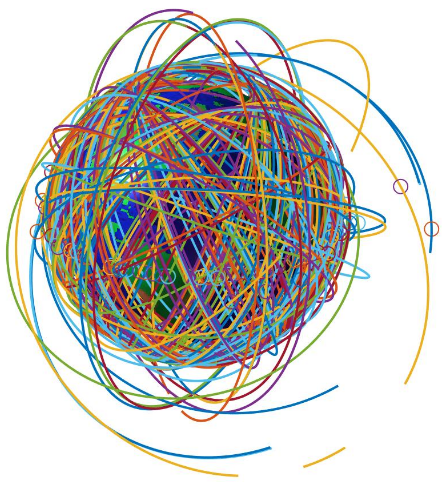
Table 1 contains the main results of the analysis. The software was able to detect the fragmentation at the correct epoch, and to identify correctly the 19 objects involved in it (the 2 parent objects and the fragments) whose TLEs where present in the initial set. The computational time is of the order of a few minutes, due to the relatively low number of objects non related to the fragmentation in the initial set of TLEs. More time (of the order of hours) would be required in case the full set of TLEs (around 15000) published on 17th February 2009 had been analysed, or in case more fragments from the event were included, due to the higher number of close encounters occurring in the search window.

The high number of fragments estimated by the breakup model results from the type of fragmentation event: a catastrophic collision between satellites, with an impact speed of 11.647 km/s (estimated from the propagation of the TLEs of the two objects to the time of the event).

4 CONCLUSIONS

The population of space debris surrounding the Earth is growing at an alarming rate, due to the increasing levels of human activity in space, causing fragmentation events to become more frequent and posing a threat to the safe operation of satellites. Thus, the early detection of these fragmentations

[1]For this purpose, the following machine specifications are considered: Intel(R) Core(TM) i7-7700 CPU @ 3.60 GHz 3.60 GHz



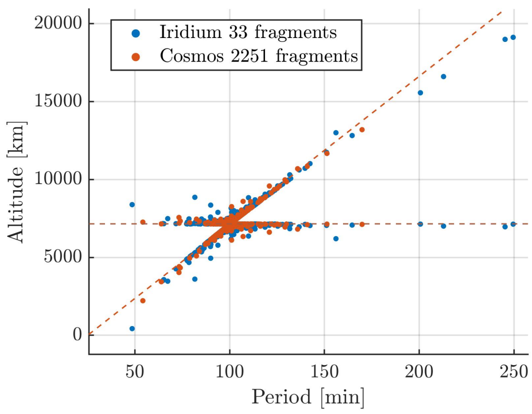
(a)

(b)

(c)

Figure 10: (a) Orbits of the 2000 objects initially included in the TLE set for the Iridium-Cosmos test case, dating to 7 days after the collision (focus on the LEO region). (b) Orbits of the objects after passed the pre-filtering and the triple-loop filter for the Iridium-Cosmos test case. (c) Orbits of the objects presenting close encounters around the possible epoch of the fragmentation: the blue cluster is compatible with the orbit of Iridium 33, while the red cluster is compatible with the orbit of Cosmos 2251.

(a)



(b)

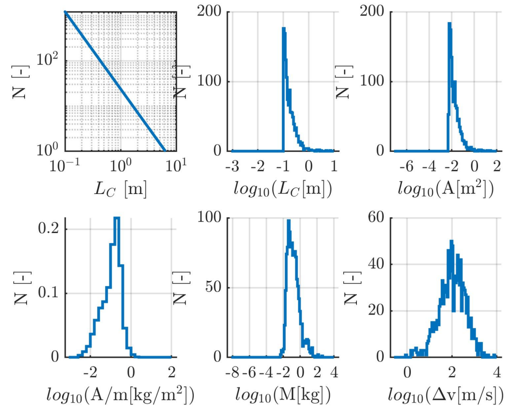


Figure 11: Distributions of the Iridium-Cosmos fragments generated via the breakup model. (a) Gabbard diagram, with the fragments from Iridium 33 in blue, the fragments from Cosmos 2251 in red. (b) Distributions of (from top left to bottom right) cumulative distribution of characteristic lengths, distribution of characteristic length, of cross-sectional area, of area-to-mass ratio, of mass, and of relative velocity.

Table 1: Main parameters used to detect the Iridium-Cosmos fragmentation and main results.

Initial size of TLE set	$\sim$ 2000
Date of generation	17th February 2009
Time interval selected	10 days
Estimated epoch of the event	10th February 2009, 16:55:55
Number of objects involved	19
Probable parent object(s)	Iridium 33 (ID 24946), Cosmos 2251 (ID 22675)
Estimated number of fragments	1208 (367 from Iridium 33, 841 from Cosmos 2251)
Computational time	10.8 min

becomes a key task in maintaining the safety of operative satellites.

The PUZZLE software was developed with the goals of characterising in-orbit fragmentations starting from a set of unclassified TLEs, by detecting whether a fragmentation has occurred in the recent past, determining which objects were involved, and estimating the distributions of characteristics of the fragments. This paper presented the architecture of PUZZLE and the operations performed within the five modules of the software. A test case was presented and discussed, to show the capabilities of the software in detecting a fragmentation and the computational cost.

The discussion presented here shows that PUZZLE is capable of detecting a fragmentation and correctly identify the epochs and the objects involved in most cases. This is achieved through the use of algorithms which identify common aspects among the motion of the objects under examination, focusing on the evolution of the osculating orbital elements: the occurrence of a large number of close encounters between objects in a short time span via the triple-loop filter, and the similarity of the orbits of such objects via hierarchical clustering. In this way, the software can detect whether a set of objects converge to the same region of space going backwards in time, thus identifying a possible fragmentation event occurred at a specific epoch.

Limitations exist due to the sensitivity of the algorithms employed in the analysis from the size of the initial TLE set and the amount of fragments represented in it, and from the various parameters used to gradually prune the TLE set in search of a subset showing a behaviour compatible with a fragmentation. Some of them have been already identified (such as the distance margins to detect close encounters), however further study is required to identify other such influential parameters and, possibly, optimal values to ensure accurate and fast results.

Future work to the PUZZLE software will focus on improving some of the main algorithms used to detect fragmentations. The identification of the fragmentation epoch will be performed in a more efficient way, by considering variable time intervals in which the close encounters are clustered in order to avoid incorrect selection of an epoch based solely on the absolute number of close encounter in its proximity. The fragmentation search will be extended to longer time scales (of the order of months or years) using a semi-analytical formulation and averaged orbital elements. Finally, uncertainties over the states represented by the TLEs will be included, to allow for a better correlation between the orbital states of objects involved in the fragmentation, and between the unclassified objects and the known ones.

ACKNOWLEDGEMENTS

The research activities described in this paper were performed within the European Commission Framework Programme H2020 “SST Space Surveillance and Tracking” contracts N. 785257 (2-3SST2016).

REFERENCES

[1] D. S. McKnight, “Orbital debris hazard insights from spacecraft anomalies studies,” *Acta Astronautica*, vol. 126, pp. 27–34, 2016, Space Flight Safety.

[2] R. L. Andrișan, A. G. Ioniță, R. D. González, N. S. Ortiz, F. P. Caballero, and H. Krag, "Fragmentation event model and assessment tool (fremat) supporting on-orbit fragmentation analysis," in *Proceedings of the 7th European Conference on Space Debris*, 2017.

[3] S. Frey, C. Colombo, and S. Lemmens, "Advancement of a density-based debris fragment model and application to on-orbit break-up localisation," *36th IADC Plenary Meeting*, 2018.

[4] ——, "Density based modelling and indication of break-up location and epoch from fragments using backwards propagation," *5th European Workshop on Space Debris Modelling and Remediation*, 2018.

[5] L. Dimare, S. Cicalò, A. Rossi, E. M. Alessi, and G. Valsecchi, "In-orbit fragmentation characterization and parent bodies identification by means of orbital distances," in *Proceedings of the 1st International Orbital Debris Conference (IOC)*, 2019.

[6] R. Southworth and G. Hawkins, "Statistics of meteor streams," *Smithsonian Contributions to Astrophysics*, vol. 7, pp. 261–285, 1963.

[7] A. Lidtke, D. Gondelach, and R. Armellin, "Optimising filtering of two-line element sets to increase re-entry prediction accuracy for GTO objects," *Advances in Space Research*, vol. 63, pp. 1289–1317, 02 2019.

[8] F. R. Hoots, L. L. Crawford, and R. L. Roehrich, "An analytic method to determine future close approaches between satellites," *Celestial Mechanics*, vol. 33, pp. 143–158, 06 1984.

[9] G. F. Gronchi, "An algebraic method to compute the critical points of the distance function between two keplerian orbits," *Celest. Mech. Dyn. Astron.*, vol. 93, pp. 295–329, 09 2005.

[10] T. Kelso, F. Hoots, and R. Roehrich, "Spacetrack report no. 3-models for propagation of norad element sets," *NASA, Tech. Rep*, 1988.

[11] D. Vallado, P. Crawford, R. Hujsak, and T. Kelso, "Revisiting spacetrack report no. 3: Rev," 08 2006.

[12] AGI (Ansys Company), "SGP4 propagator," https://help.agi.com/stk/11.0.1/Content/stk/vehSat_orbitProp_msgp4.htm, 2016, [Online; accessed 1-May-2021].

[13] V. Zappala, A. Cellino, P. Farinella, and Z. Knezevic, "Asteroid families. I. Identification by hierarchical clustering and reliability assessment," *The Astronomical Journal*, vol. 100, pp. 2030–2046, 11 1990.

[14] N. Johnson, P. Krisko, J.-C. Liou, and P. Anz-Meador, "NASA's new breakup model of EVOLVE 4.0," *Advances in Space Research*, vol. 28, pp. 1377–1384, 12 2001.

[15] NASA, "Proper implementation of the 1998 NASA breakup model," *Orbital Debris Quarterly News*, vol. 15(4), pp. 4–5, 2011.