ESA'S COLLISION AVOIDANCE SYSTEM AT ESOC – STATUS RECENT UPGRADES AND UPCOMING EVOLUTION

Volker Schaus(1), Steffen Weber(2), Quirin Funke(1), Vjeran Wertag(1), Raúl Domínguez González(3), Yannick Barthel(2), Frazer McLean(4), Klaus Merz(5), Jan Siminski(5), Vincenzo Vitiello(3), Alvaro Arroyo Parejo(3), Stephan Kranz(2)

(1) IMS Space Consultancy GmbH for ESA/ESOC, Robert-Bosch-Straße 7, 64293 Darmstadt, Germany, Email:{v.schaus, q.funke, v.wertag}@ext.esa.int
(2) Telespazio Germany GmbH, Europaplatz 5, 64293 Darmstadt, Germany, Email: {stephan.kranz, yannick.barthel, steffen.weber}@telespazio.de
(3) Indra Deimos, Ronda de Poniente 19, 28760 Tres Cantos, Spain, Email: {rdominguezgo, vvitiello, caarroyo}@indra.es
(4) CS Group – Germany, Campus Berliner Allee, Berliner Allee 65, 64295 Darmstadt, Germany, Email: Frazer.Mclean@ext.esa.int
(5) ESA/ESOC, Robert-Bosch-Straße 5, 64293 Darmstadt, Germany, Email: {klaus.merz, jan.siminski}@esa.int

ABSTRACT

Collision avoidance is an integral part of spacecraft operations. It is essential for safeguarding own assets as well as preventing the proliferation of space debris. The European Space Agency's (ESA) Space Debris Office provides operational collision avoidance support to ESA and third-party missions since around 20 years. The support covers missions in low-Earth orbit and highly eccentric ones, but also special cases such as conjunction analysis for Earth flybys of interplanetary missions.

The service has seen continuous evolution, coping with diversifying, and increasing mission needs and platform capabilities, the increase of tracked space debris objects in orbit, but also with the rise of more and better data sources, as well as means to better address uncertainties in the prediction process for conjunction events. Our most recent challenge results from accelerated launch traffic in particular of small satellites and the emergence of large constellations.

The overall processing chain is currently modernized to cope with expected future data loads and to be able to ingest conjunction warnings from multiple surveillance data providers. The upgrades aim at making the tool base more modular allowing for a gradual replacement of legacy tools as well as a more flexible process handling allowing for a mix of automated and manual workflows. The chain includes conjunction event detection, collision risk assessment and visualization, orbit and covariance propagation, manoeuvre optimisation, distributed task queue, process control and data handling.

This paper will describe the current approach focusing on the new developments and recent operational experience. It will present the changes already implemented and provide an overview of the planned further evolution. Topics include a new process control system and enhanced automation as well as replacement, refactoring and modernisation of key flight dynamics components.

The new process control framework is built on distributed computing using Celery task queues with a custom workflow and job implementation. A central scheduler is responsible for starting workflows and keeping track of their status. A web-based User Interface using Flask in the backend and React in the frontend allows the debris analyst to monitor and control the process chain.

The modernisation of flight dynamics components is based on the integration of the new ESA/ESOC Flight Dynamics library GODOT [1] [2] as infrastructure and an application library CORAL developed on top of GODOT in Python. It features covariance propagation and interpolation and various collision probability algorithms and a manoeuvre planning framework, overall aiming to replace the legacy software CORAM [3].

On the operations aspects we report on organisational changes, e.g. the support of the ESA-operated Sentinel spacecraft by a dedicated team, as well as special operations and approaches such as the ones employed during the long orbit lowering phase for Sentinel-1B and its frequent and large manoeuvres as well as lessons learnt.

1 INTRODUCTION

Collision avoidance is part of routine spacecraft operations in ESA to comply with Space Debris Mitigation requirements [4] and the recent Zero Debris Charta [5] and the associated booklet [6]. Operational conjunction analyses and collision avoidance activities at ESA started with ERS-2 and Envisat and nowadays concentrate on ESA's Earth Explorer missions, the Copernicus' Sentinel spacecraft in Low-Earth orbit (LEO), ESA's science missions in Highly-Elliptical

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)

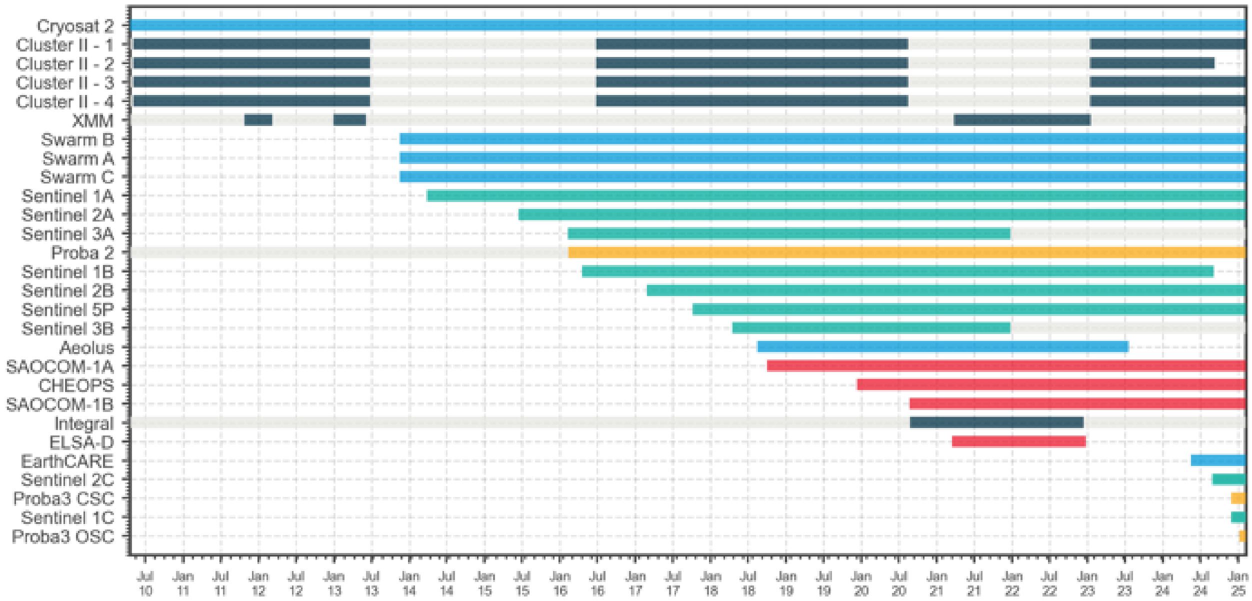

Figure 1: Missions supported by the Space Debris Office

orbits (HEO) as well as on external partner spacecraft – see Figure 1 for a historic overview of the missions supported by the operational collision avoidance process at the SDO and [7] [8] for the processing system. In the year 2024 four new missions have been added to the system. Additionally, several other missions were covered in the past under varying support levels and durations (e.g. LEOP): Proba-1, -V, Galileo/Giove, METOP-A/-B/-C, MSG-3/4, Artemis, Lisa-Pathfinder, Sentinel-6A.

The collision probability assessment is based on conjunction data messages (CDMs) provided by the US 18th and 19th Space Defense Squadron (18/19 SDS). The CDMs contain information on conjunctions between tracked objects in the US catalogue (chasers) and the own operated spacecraft (target) trajectories, in particular time of closest approach (TCA), separations, state vectors and covariances at TCA as well as auxiliary information on the orbit determination setup and quality. Due to a data sharing agreement between the US Strategic Command (USSTRATCOM) and ESA, signed on October 30th, 2014, the SDO has access to CDMs covering larger volumes around the target trajectories and longer lead times.

The collision risk assessment requires the following processing steps:

- Compute probability of collision using object geometries from DISCOS and the orbital state and uncertainty information from the CDM for the target and chaser.
- Compute the collision probability using the state and covariance for the chaser from the CDM and the state and covariance from an internal flight dynamics solution.
- Generate a collision avoidance manoeuvre plan in case the probability is above a reaction threshold.
- Search for conjunctions with ephemeris files provided by other operators or an internal catalogue built up from CDM chaser objects.

The executions of processing steps are triggered by new data, i.e. when new CDMs or new flight dynamics ephemeris files are available. Large batches of CDMs were initially provided multiple times per day. Then, the CDMs were processed in a sequential way. To deal with an increased number of CDMs from possibly multiple data providers, a new parallel collision avoidance processing framework needed to be developed.

Over time, the processing system was facing more and more challenges, such as:

- Increased processing demands:
  - by supporting more missions.
  - by seeing more conjunctions events, e.g., because of regular CDMs and operator ephemeris screenings.
  - by accessing other catalogue providers, e.g., EU-SST and LeoLabs.
  - by running parallel assessment of the conjunctions with alternative calculation methods, e.g. worst-case analysis or covariance scaling.
- Increased complexity for Collision Avoidance Manoeuvre (CAM) design: We are seeing recurring events with the same chaser and cases where while mitigating one conjunction with high probability, another subsequent event is then showing collision probabilities above the threshold for action. The two

cases fall in the same category. The standard approach was to design a manoeuvre against the current high probability case and then evaluate the probability of the other events. This sometimes leads to manual iterations of redefining the manoeuvre, until no other event is above the threshold. The local optimisation against a single event with potential manual iterations should be replaced by a multi-event optimisation which can design the avoidance manoeuvre against several conjunctions (global optimisation).

2 OVERVIEW OF THE MODERNISED PROCESSING CHAIN

The new framework is built on distributed computing using Celery [9] task queues with a custom workflow and job implementation. Workflows control the order of jobs, which perform the actual work, and track their progress; they also support interaction with the jobs, like changing priorities. A central scheduler is responsible for starting workflows and keeping track of their status. The system is designed to be scalable as new worker nodes can be easily added and jobs can be worked on in parallel. The system is described in Figure 2.
A web-based UI using Flask [10] in the backend and React [11] in the frontend allows the debris analyst to monitor and control the process chain.

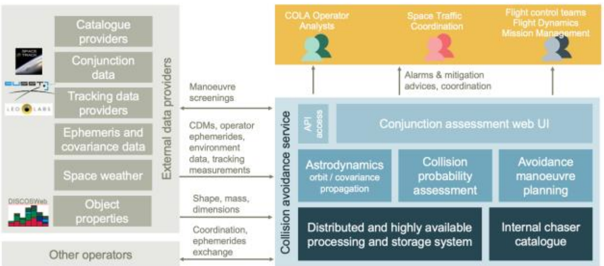

Figure 2: New Collision Avoidance Process at the ESA/ESOC Space Debris Office

The system has been introduced operationally last year. All new missions will directly be using the new system; in 2024 all four new missions, namely EarthCARE, Sentinel-1C, -2C and Proba 3 were directly supported via the new system. It is planned to transition existing mission support to the new system throughout the next months.

2.1 Process control

By basing the processing system on Celery with a custom workflow and job implementation we achieve scalability and flexibility. Celery's distributed task queue architecture enables us to scale our processing capabilities based on demand, ensuring seamless performance even during peak loads like LEOPs. Physical servers can easily be augmented by cloud resources.

As Celery's built-in workflow system (canvas) is relatively stiff, we went for a custom workflow and job implementation. This gives us the freedom to tailor the system to our specific needs, allowing for efficient task orchestration and streamlined operations.

Workflows can either be started manually via the graphical user interface (UI) or command-line interface (CLI), with automated time-based schedules, or when new orbit files are delivered. This offers a robust and adaptable solution for managing workflows across a diverse range of scenarios and requirements, including LEOPs and other special operations.

2.2 Data handling

The data handling revolves around a robust infrastructure centred on a PostgreSQL database [12] and MinIO for S3-like file storage [13]. Through data mirroring, this setup ensures efficient and reliable storage of both structured and unstructured data generated throughout the system's operation. A notable feature is the comprehensive persistence of all job inputs and outputs, stored within the central database and MinIO, respectively. This approach enables seamless reprocessing and debugging by providing a complete historical record of data transformations and outcomes.

2.3 Tools

A significant part of the development of the new framework and its workflows and jobs consists in encapsulating clear interfaces to the functionalities provided by the core numerical tools. Therefore, this activity also paves the way for the opportunity to replace legacy code one by one. This has already started with the replacement of CORAM with CORAL, a new collision avoidance library, based on the GODOT astrodynamics platform currently being developed by ESOC FD with contributions from SDO.

2.4 Interface

The web-based UI is built with React components and designed mobile-friendly with Tailwind CSS. A core concept is to support all configuration and tasks via the UI, also from mobile devices. The progress and logs of workflows and their jobs can be monitored live. Figure 3 shows a screenshot of it.

3 MODERNISATION OF THE FLIGHT DYNAMICS COMPONENTS

CORAL is a library and software tool that has been developed as a successor of CORAM, the operational software tool at the ESOC Space Debris Office for collision risk assessment and collision avoidance manoeuvre optimisation.

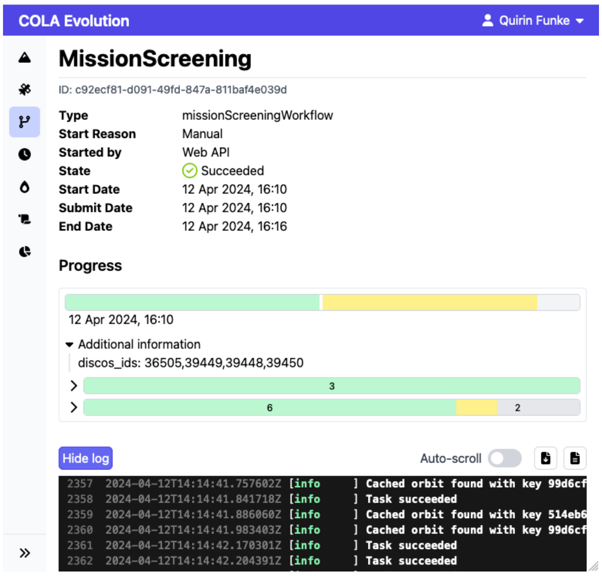

Figure 3: Workflow display UI

CORAL is written in python and uses the GODOT flight dynamics software [1], [2] developed by ESOC for orbit propagation, interpolation and collision detection. For most functionalities the GODOT python layer (godotpy) is used, while the Lagrange interpolation is accessed directly from the C++ GODOT library using the pybind 11 layer.

The force model can be configured to include Solar Radiation Pressure and Atmospheric Drag. Besides CAMs also existing predefined manoeuvres can be integrated in the propagation. Both impulsive and finite manoeuvres are supported.

Besides the full 6 by 6 covariance of the state, additional uncertainties can be configured on the solar radiation and drag uncertainty coefficients, as well as manoeuvre magnitudes and directions. All uncertainties are assumed to be stochastic and Gaussian.

Different propagation modes are supported for the state, as well as for the covariances: free propagation from an initial state or interpolation from a time series input. The modes can be mixed, e.g. the state can be interpolated from an existing trajectory, while the covariance is freely propagated from an initial value, resetting the state to the reference trajectory regularly. Target and chaser can be propagated in different modes.

For both propagation and interpolation, the GODOT library is used; free propagation is handled by a Runge-Kutta 7/8 integrator, and interpolation by a 3rd order Lagrange interpolator.

Collision detection is based on a grid search performed with GODOT's events library. The condition for a conjunction candidate is a zero crossing of the relative velocity between chaser and target with a positive 2nd derivative of the relative distance between them.

The collision probability of the encounters can be calculated with the following algorithms:

1. The method by Alfriend and Akella [14] computes the collision risk by two-dimensional integration of the hard body projection in the encounter plane.
2. In an improved method by Patera [15] the contour integration of the projection is performed instead, significantly speeding up the computation.
3. In the covariance scaling method, the covariance is scaled for both target and chaser, the scaling factor varying within a given interval. For every scale factor, the collision probability is evaluated using the Patera method. This method is useful for poorly constrained covariances.
4. In the method by Klinkrad [16], the collision risk is evaluated using the most pessimistic scaling factor of the combined covariance.
5. For encounters of low relative velocities, the implicit assumptions of the previous methods no longer hold, i.e. an instantaneous encounter in which the trajectories can be assumed to be linear and the covariances constant. For this case an interval-slicing method [17] is used, in which the collision interval is subdivided, and the collision risk is evaluated for every slice. Within each slice, the mentioned assumptions for high-speed encounters are valid. To calculate the collision risk of each slice, any high-speed collision risk algorithm can be used, properly scaled to take only the contribution of the slice into account. From the instantaneous risks of each slice the cumulated total collision risk can be derived.

In addition to the described methods, a Monte Carlo approach has been implemented. From the nominal initial state and SRP and drag parameters, modified samples are created based on the covariance and uncertainties for target and chaser. They are then propagated and the encounter finding algorithm is performed. The probability of collision is determined from the number of samples in which target and chaser come closer towards each other than their combined radius.

The number of runs can be either defined beforehand, or a dynamical criterion can be employed in which the simulation is performed until the accuracy of the calculated probability lies inside a confidence interval, using the Wilson method [18].

Avoidance manoeuvres can be added and their effects analysed by the operator by means of visual inspection of the dependence of the collision probability on manoeuvre parameters. A 1D scan of manoeuvre magnitude can be produced, for a manoeuvre in or against flight direction, half a revolution before the encounter, as illustrated on an example in Figure 4.

Also, a 2D scan of manoeuvre magnitude and epoch, in steps of 1, 3, 5, ... half-revolutions before the encounter can be created. The library can output plots for the operator to inspect and help decide on avoidance manoeuvres to execute.

Finally, the library offers functionality for manoeuvre optimisation, using the NLopt [19] optimiser together with the pygmo/pagmo optimisation library [20]. A wide range of optimisation algorithms is available from which the user of the library can freely choose. Manoeuvre magnitude and direction can be optimised to maximize the distance or minimize the collision probability at time of closest approach (TCA) given a maximum threshold on the manoeuvre delta V; or a minimum delta V can be determined fulfilling the requirement of a minimum allowed distance or maximum allowed collision probability at TCA.

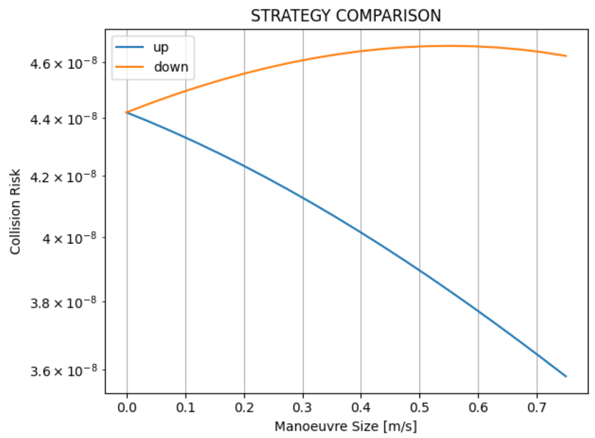

Figure 4 Example plot of collision probability as a function of manoeuvre size for a manoeuvre in-flight direction (blue), or against it (orange), computed by coral. The manoeuvre is applied half a revolution before the encounter.

4 OPERATIONAL ASPECTS OF THE SENTINEL FLEET

The operations of the ESA-operated Sentinel spacecraft were transferred to a new contractual and team setup (COP2) in 2022 which includes a dedicated team responsible for the collision avoidance of Sentinel 1A, 1B (until September 2024), 1C (from December 2024), 2A, 2B, 2C (from September 2024) and 5P. The responsibility of the team is not only the operations but also the monitoring and maintenance of the software and hardware infrastructure (where applicable).

The handover to the new team took place in January 2022 supported by a dedicated instance of the processing chain:

- Two new redundant servers were prepared to be dedicated exclusively for the Copernicus missions. These servers were sized to cope with the computational workload required for the Sentinels.
- A new, empty database (i.e., with no previous CDMs) was deployed. This database is routinely updated by copying data from the DISCOS database. The synchronization involves only the information required for the collision avoidance service.
- The legacy processing software was deployed on the new servers. The legacy software was updated were needed to cope with the new requirement that Sentinel missions were to be operated from different servers. It was decided to perform no branching of the operational software for the Sentinels chain, to ensure that the subsequent software maintenance activities are not duplicated.
- The new team has an independent, dedicated on-call service via a dedicated phone. The automated alerts from the system are directed to that phone.
- External contact information was updated so that third parties trying to contact the Sentinel operations team for collision avoidance topics can contact directly the COP2 team.

The COP2 contract includes several specific requirements. The team had to adapt the software systems and/or procedures to meet them.

- Access to the operational servers for the COP2 team and 24/7 physical access to ESOC when needed.
- The COP2 team is required to formally report the status of tasks. This is achieved by means of an external task management software tool. Tasks are declared as tickets in this tool, and to make this reporting as simple as possible for the team, as well as reliable, an effort was made to automate the ticketing management by leveraging its API.
  - The operational procedure for escalating a conjunction includes a step to create the corresponding ticket. This step involves running a script that extracts the relevant event data (involved objects, TCA) directly from SCARF.
  - For each escalated event, the procedure includes the creation of a small report with the insights of the involved person(s) as well as lessons learnt. For simple conjunctions, this report can be very short (thus requiring no effective work), while it allows to capture the lessons learnt from interesting conjunctions.
  - The operational procedure for supporting orbit control manoeuvres includes a step to create and update a related ticket. Again, the ticket creation involves running a script that extracts the manoeuvre information from the related manoeuvre prediction files.
  - Monthly reports are generated from a script that extracts and summarizes the relevant tickets from the ticketing tool, producing an editable MS Word document. Thus, the task of reporting once per month primarily involves reviewing

this automatically generated document.

- Part of the responsibilities of the COP2 team is to train newcomers. Within COP2, a formal training process (called certification) is required by ESA for all operations team members. The collision avoidance team has created this process starting from 2022 and has successfully trained the initial three team members, as well as three additional persons in 2022, 2023 and 2024. The certification process includes a theoretical assessment with a pool of questions that is maintained by the COP2 team, and a practical assessment that involves a simulated exercise. With future evolutions of the COLA framework system, we expect to be able to perform the exercises using the COLA framework system with manually crafted cases.

The operations of the spacecraft that were active at the handover (1A, 1B, 2A, 2B and 5P) were and are still being carried out with the legacy software system, as this was the system that was set up before the kick-off of the COP2 contract. However, as the plan is to phase out the legacy system in favour of the COLA processing system described in section 2, the LEOP and operations of spacecraft launched after 2022 are carried out using the COLA processing system, retaining the legacy chain for cross-checking purposes only.

The COLA processing chain system was adapted to the COP2 servers from January 2023 and deployed by May 2024 (before the Sentinel-2C simulations campaign before its launch). This was the second time that the COLA processing chain system was used in a LEOP (first one was EarthCARE in May 2024). The LEOP with the new system worked with no relevant issues and was a large improvement over LEOPs conducted with the legacy system in terms of workload. After the LEOP was declared complete, the operations transitioned from LEOP to routine status, continuing to use the new COLA framework system. In December 2024, the same process was carried out for the Sentinel-1C LEOP.

4.1 Sentinel 1B disposal

In December 2021, the Sentinel-1B payload failed unexpectedly. An investigation was immediately started to determine the causes of the failure and to attempt to reactivate the payload, which turned out to be not possible. In August 2022, it was decided to terminate the Sentinel-1B mission. As the spacecraft was in a healthy state (except for the payload), this marked the kick-off of the disposal campaign. The campaign aimed to lower the spacecraft orbit to ensure a re-entry 25 years after disposal, according to the guidelines at the time. The first step was to move Sentinel-1B orbital position by 30 degrees from its nominal position.

The Sentinel-1B spacecraft has a set of thrusters for in-plane orbit control manoeuvres. In principle, these thrusters could be used for deorbiting. However, their thrust is limited to only ~1.5 cm/s per burn. Even though this limit is enough for routine orbit control, deorbiting using them would have involved an extremely large number of manoeuvres over a long period. Instead, it was decided to change the spacecraft attitude for deorbiting and to employ the out-of-plane thrusters, which were not affected by such limitation.

With these, the plan was to perform batches of three retrograde 70 cm/s manoeuvres every Tuesday and Thursday. The manoeuvres were performed three orbits apart at either the perigee or apogee.

4.2 Need for worst case approach

When predicting conjunctions happening after a future manoeuvre, it is necessary to propagate the covariance through the manoeuvre, considering the uncertainty added in by the manoeuvre. Operationally, we typically model this with a 1-sigma uncertainty of 1.5% of the Delta-V for the magnitude of the manoeuvre, and no uncertainty for the direction. If the propagation goes through one or several manoeuvres, the covariance is inflated for each of them. For typical manoeuvres with a size on the order of 1 cm/s, the corresponding uncertainty is far less than 1mm/s. However, in the case of the disposal manoeuvres (210 cm/s), the uncertainty is on the order of cm/s. This brings several undesirable effects in the operational process.

- The difference between the nominal and the actual manoeuvre is quite noticeable. For example, a misperformance of 1.5% of 210 cm/s is equivalent to 3.15 cm/s, and it is only known a few hours after the manoeuvre has been executed.
    - Conjunctions that were considered non-risky with the nominal trajectory can become risky with the real trajectory (as the miss distance may change significantly).
    - The conjunctions that were predicted beforehand were known in a screening volume around the nominal trajectory. The real and nominal screening volumes drift with respect to each other at a rate of ~8 km/day. This means that after the calibrated manoeuvres, new conjunctions that were previously outside the screening volume may suddenly appear.
- The covariance inflation may lead to an underestimation of the risk of conjunctions evaluated before the manoeuvres with TCAs after the manoeuvres. Even though the covariance inflation and the subsequent risk dilution make sense mathematically, the operational consequence is that sometimes risky conjunctions appear immediately after computing a calibrated orbit, even if the manoeuvre performance is near the nominal.

These effects are negligible in routine operations (routine orbit control manoeuvres and collision avoidance manoeuvres). However, during the disposal campaign, the collision avoidance engineer had to deal with them twice per week.

Ideally, we would like to keep the covariance inflation (as the manoeuvre performance uncertainty is a real phenomenon that needs to be accounted for), and at the same time mitigate the undesirable consequences (the risk dilution). We did this by setting up an operational procedure, called the worst-case scenario, that we describe here. This is an application of the work described at [8].

This scenario is computed for all existing conjunctions predicted before the manoeuvre execution and with TCA after the manoeuvre. It comprises two steps:

- Compute and apply a displacement of the target assuming linearized dynamics near the TCA. This displacement can be either interpreted as a displacement in the along-track direction or as a time shift applied to the target. With this displacement of the target, a new TCA is computed that minimizes the miss distance. The range of allowable displacements for the target is bounded by operationally observed values, as will be explained later.
- Compute the worst possible probability of collision at that TCA by selecting a scaling factor for the target covariance that maximizes the risk. The range of valid scaling factors is restricted to ensure that only reasonable values are used.

In both cases, it is necessary to determine the range of reasonable displacements at the TCA. This is achieved by studying all CDMs generated internally (MiniCat) for Sentinel-1B, as seen in Figure 5. Each CDM contains a propagation of the operational Sentinel-1B covariance from the epoch of the orbit determination to the TCA, affected by real world effects (such as space weather) and different initial conditions. Therefore, the blue points in the figure characterize the real-world operational covariance propagation process. We use them as a predictor for the range of acceptable along-track covariance values as a function of the propagation time, by tabulating the values in the plot. These are used both to consider the acceptable range of along-track displacement of the target near the TCA and the expectable range of covariances.

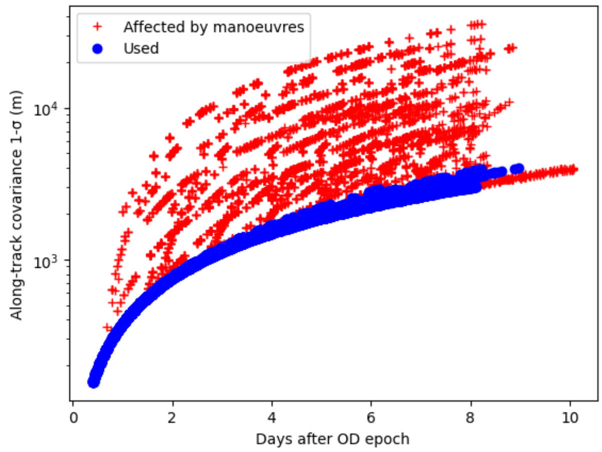

Figure 5. 1-sigma along-track operational covariances used at ESOC for Sentinel-1B for the period 2022-2023.

4.3 Operational approach

The disposal plan involved two batches (OCM1 and OCM2) of three retrograde 70 cm/s manoeuvres every Tuesday (OCM1) and Thursday (OCM2). The manoeuvres were performed three orbits apart at either perigee or apogee.

Each manoeuvre batch had an overall nominal Delta-V of 210 cm/s each. For pre-manoeuvre screenings, we assumed nominally a 1-sigma uncertainty of 1.5% of the Delta-V magnitude, and no uncertainty in the direction of the Delta-V, as there was not enough statistically significant data to compute a better estimate.

The weekly calendar for collision avoidance was:

- <u>Sunday</u>: Submit ephemeris with OCM1 and OCM2 included for screening, and ephemeris without the manoeuvres.
- <u>Monday morning</u>. Perform worst-case analysis 1.
- <u>Monday morning</u>. Decide a GO/NOGO for OCM1 based on the screening and worst-case analysis 1 results.
- <u>Monday to Tuesday</u>: OCM1 execution
- <u>Tuesday afternoon</u>: First calibrated ephemeris after OCM1 execution
- <u>Tuesday afternoon</u>: Perform worst case analysis 2.
- <u>Tuesday afternoon</u>: Submit ephemeris with OCM2 included for screening, and ephemeris without the manoeuvre.
- <u>Wednesday morning</u>: Perform worst-case analysis 3.
- <u>Wednesday morning:</u> Decide a GO/NOGO for OCM2 based on the screening and worst-case analysis 3 results.
- <u>Wednesday to Thursday</u>: OCM2 execution
- <u>Thursday afternoon</u>: First calibrated ephemeris after OCM2 execution
- <u>Thursday afternoon</u>: Perform worst case analysis 4.

The worst-case analyses 1 and 3 were performed at the decision time before performing manoeuvres. These cases were used to reveal conjunctions that were not relevant if assuming the nominal manoeuvre performance but could become risky in case of a certain manoeuvre misperformance.

Worst-case analysis 2 was performed as soon as the first calibrated ephemeris after OCM1 was available. This was the first time that real data was available after the manoeuvre was executed, providing the first indication of the manoeuvre performance. For interface reasons, the calibrated ephemeris had an orbit determination epoch set before the end of the manoeuvre batch. Thus, even though the manoeuvre batch was already calibrated, the resulting conjunction predictions were affected by risk dilution. Therefore, the worst-case analysis 2 served a double purpose. We executed it once disallowing displacements in the along-track direction to remove the effect of risk dilution. After that, a second execution allowing displacements revealed the risky cases that may happen after OCM2.

Worst-case analysis 4 was performed as soon as the first calibrated ephemeris after OCM2 was available. In this case, the analysis was executed with the along-track displacement constrained, in order to remove the effect of covariance inflation.

When a risky conjunction was detected shortly after the predicted manoeuvre execution time either with this method or the conventional one, the approach was to modify the times of the batch executions, while keeping the overall Delta-V. This adjustment helped to avoid most of the conjunctions (as just shifting the time of the manoeuvres resulted in a large change at the time of TCA) and minimized the impact on long-term planning for the mission control team. If this approach turned out to be not feasible, the next option considered was to abort the entire manoeuvre batch.

It is also worth noting that for all the manoeuvre batches, we screened both options assuming the manoeuvre batch executes nominally or not at all. Screening the free-propagation case allows to cover the case in which the manoeuvre batches were aborted on short notice by the Fault Detection, Isolation and Recovery (FDIR) of the spacecraft. We also considered submitting several ephemerides based on the execution of one, two or three manoeuvres from the batch, however, this approach was not followed.

Overall, this approach resulted in one manoeuvre batch modified because of a conjunction that would not have been raised otherwise. In another instance, the flight control team was pre-warned of a risk that finally did not materialize. Additionally, a small number of cases were monitored by the collision avoidance team, which required no further action from the other teams.

5 SUMMARY AND OUTLOOK

The paper outlines ESA's advancements in collision avoidance, driven by increasing mission complexity, the growing number of space debris objects, and new satellite constellations. A modernised processing framework has been developed and successfully introduced with the most recent ESA mission launches in 2024.

Modernisation around Flight Dynamics tools and the collision probability calculation methods have been implemented. The main goal is to replace legacy implementations in Fortran with a modern setup based on python and the new community licensed astrodynamics library of ESA called GODOT.

The third part of the paper presents the special challenges of the Sentinel fleet. This includes a section of the external setup of the collision avoidance processing for all Sentinel missions, as well as a description of a worst-case scenario approach which was developed to manage large manoeuvre uncertainties and mitigate unexpected conjunction risks to support the de-orbiting phase of Sentinel 1B.

REFERENCES

[1] "GODOT community." Accessed: Mar. 24, 2025. [Online]. Available: https://godot.space-codev.org/community/

[2] "GODOT documentation." Accessed: Mar. 24, 2025. [Online]. Available: https://godot.io.esa.int/godotpy/

[3] J. Cobo, N. Sánchez-Ortiz, I. Grande, and K. Merz, "CORAM: ESA's Collision Risk Assessment and Avoidance Manoeuvres Computation Tool," in *2nd IAA Conference on Dynamics and Control of Space Systems*, Rome, Mar. 2014.

[4] ESA Space Debris Mitigation Working Group, "ESA Space Debris Mitigation Requirements," Oct. 2023. Accessed: Mar. 24, 2025. [Online]. Available: https://technology.esa.int/upload/media/ESA-Space-Debris-Mitigation-Requirements-ESSB-ST-U-007-Issue1.pdf

[5] "Zero Debris Charter – Towards a safe and sustainable space environment." Accessed: Mar. 24, 2025. [Online]. Available: https://esoc.esa.int/sites/default/files/Zero_Debris_Charter_EN.pdf

[6] "Zero Debris Technical Booklet." Accessed: Mar. 24, 2025. [Online]. Available: https://esamultimedia.esa.int/docs/spacesafety/Zero_Debris_Technical_Booklet.pdf

[7] K. Merz *et al.*, "Current Collision Avoidance

service by ESA's Space Debris Office," in *7th European Conference on Space Debris*, T. Flohrer and F. Schmitz, Eds., Darmstadt, Apr. 2017. [Online]. Available: http://spacedebris2017.sdo.esoc.esa.int,

[8] K. Merz *et al.*, "ESA's Collision Avoidance Service: Current Status and Special Cases," in *8th European Conference on Space Debris (virtual)*, Darmstadt, Apr. 2021. Accessed: Mar. 24, 2025. [Online]. Available: https://conference.sdo.esoc.esa.int/proceedings/sdc8/paper/296/SDC8-paper296.pdf

[9] Celery Project, "Celery – Distributed Task Queue." Accessed: Mar. 24, 2025. [Online]. Available: https://docs.celeryproject.org/

[10] Pallets Project, "Flask documentation." Accessed: Mar. 24, 2025. [Online]. Available: https://flask.palletsprojects.com/

[11] Facebook Inc, "React - A JavaScript library for building user interfaces." Accessed: Mar. 24, 2025. [Online]. Available: https://reactjs.org/

[12] The PostgreSQL Global Development Group, "PostgreSQL: The World's Most Advanced Open Source Relational Database." Accessed: Mar. 24, 2025. [Online]. Available: https://www.postgresql.org/

[13] Minio Inc, "MinIO | S3 Compatible Storage for AI." Accessed: Mar. 24, 2025. [Online]. Available: https://min.io

[14] K. T. Alfriend, M. R. Akella, J. Frisbee, J. L. Foster, D.-J. Lee, and M. Wilkins, "Probability of Collision Error Analysis," *Space Debris*, vol. 1, no. 1, pp. 21–35, 1999, doi: 10.1023/A:1010056509803.

[15] R. P. Patera, "General Method for Calculating Satellite Collision Probability," *Journal of Guidance, Control, and Dynamics*, vol. 24, no. 4, pp. 716–722, Jul. 2001, doi: 10.2514/2.4771.

[16] H. Klinkrad, *Space Debris : Models and Risk Analysis*. Springer Praxis, 2006.

[17] R. P. Patera, "Collision Probability for Larger Bodies Having Nonlinear Relative Motion," *Journal of Guidance, Control, and Dynamics*, vol. 29, no. 6, pp. 1468–1472, Nov. 2006, doi: 10.2514/1.23509.

[18] E. B. Wilson, "Probable Inference, the Law of Succession, and Statistical Inference," *J Am Stat Assoc*, vol. 22, no. 158, pp. 209–212, 1927, doi: 10.2307/2276774.

[19] Steven G. Johnson, "The NLopt nonlinear-optimization package." Accessed: Mar. 24, 2025. [Online]. Available: https://github.com/stevengj/nlopt

[20] F. Biscani and D. Izzo, "A parallel global multiobjective framework for optimization: pagmo," *J Open Source Softw*, vol. 5, no. 53, p. 2338, Sep. 2020, doi: 10.21105/joss.02338.