ABACO, An Autonomous Board for Avoiding Collisions

Dario Modenini
Department of Industrial Engineering and Interdepartmental Center for Industrial Research in Aerospace, Università di Bologna
Giacomo Curzi
Interdepartmental Center for Industrial Research in Aerospace (CIRI-AERO) Università di Bologna
Alfredo Locarini
Department of Industrial Engineering, Università di Bologna
Paolo Tortora
Department of Industrial Engineering and Interdepartmental Center for Industrial Research in Aerospace, Università di Bologna
Stefano Cicalò, Alessia Bertolucci, Francesca Guerra
SpaceDys s.r.l.
Cardone Mauro
Agenzia Spaziale Italiana

ABSTRACT

Space-based solutions are generating growing interest in service providers (internet, imaging, surveillance, etc.) thanks to cheaper access to space. One of the overall effects is a dramatic increase in the space traffic to be handled. Parts of the larger Space Traffic Management discipline, Conjunction Assessment (CA), Risk Analysis (RA) and Collision Avoidance (COLA) are crucial to ensure a safe operational environment but are becoming increasingly heavy tasks to be performed with the current operator-centric approach.

Conceived with such a context in mind, project ABACO develops the operational and physical architecture of a satellite subsystem aimed at shifting onboard the workload of CARA and COLA processes. The overall goal is the manufacturing of a Technology-Readiness-Level (TRL) 4 prototype of ABACO tested in a laboratory environment. This manuscript reports on the project status, focusing on the concept of operations, and the functional and hardware descriptions.

1. INTRODUCTION

Space Traffic Management (STM) can be defined as the discipline that regulates the access, stay, and disposal of human-made resident space objects for a safe, secure, and productive environment. In order to properly actuate traffic management, engineering, space law, and space economics disciplines are all rapidly evolving [1]. In this work, we are concerned with the in-orbit conjunction assessment and collision avoidance from an engineering point of view.

The increasing space traffic demands for a scalable approach in the screening and processing of candidate spacecraft collisions. In fact, if until a few years ago, the collision warnings leading to maneuvers were just a few units per year, they are expected to scale up considerably in the next future. Considering that the handling of a single event (including the decision phase and maneuver design) commits the flight dynamics team of the involved spacecraft for several hours [2], the current approach would make the conjunction processing soon unsustainable. This is especially true for the owners of satellite constellations. A recent article [3] reported that the Starlink constellation had to make 25000 avoidance maneuvers in 6 months. With a classical approach, the flight dynamics team should have had to design and program 138 maneuvers per day. Automatization of the process, as pioneered by SpaceX, seems to be the most accepted direction by the scientific community.

Automatizing the conjunction assessment process has, however, several difficulties: i) the index used for taking decisions is not unique and different indexes with different thresholds may lead to significantly different results [4];

ii) the objects involved in a collision have to coordinate with each other to produce an effective maneuver (if they are both active spacecraft) [5]; iii) the avoidance maneuver can significantly impact the operational state of the spacecraft/constellation, which has to be accurately designed also to avoid further potential collisions (tertiary conjunctions) [6]. It is not straightforward to handle all these issues through a deterministic, automated algorithm.

Project ABACO, funded by the Italian Space Agency, proposes the operational and physical architecture of an onboard satellite subsystem aimed at reducing the ground workload of CARA and COLA operations. ABACO physical architecture features a navigation unit, devoted to the autonomous, GNSS-based, orbit determination and prediction for the hosting spacecraft, plus a collision avoidance unit, which provides risk assessment and mitigation functionalities up to the computation of evasive maneuvers.

1.1 ABACO justification

ABACO implements an automatic collision avoidance process, meaning that the system is able to estimate the risk of collision, make a reliable decision on it, interface with a coordinating agent, and finally compute and program a collision avoidance maneuver. It is a system to be integrated into the hosting space mission and is designed to be transparent to the spacecraft operators unless they want to intervene. This approach potentially releases the spacecraft operators from the RA and COLA tasks.

The above can be implemented either as a ground-based solution or an onboard solution. ABACO opts for an onboard semi-automation, taking advantage, for the primary (hosting) spacecraft, of the onboard GNSS measurements for the autonomous orbit determination and propagation, while making use of ground support to retrieve the state vector and covariance of the secondary from uplinked Conjunction Data Messages (CDM). This potentially leads to an improvement in the accuracy of the orbit prediction at closest approach due to the quasi-real time use of the GNSS observations from the hosting spacecraft.

Automating COLA onboard also reduces the time between maneuver decision and close approach, because there is no need for a visibility window for uploading the maneuver command. This reduces the number of false alerts as supported by a scrutiny of the ESA's Kelvin database of CDM developed for the Spacecraft Collision Avoidance Challenge [7]. The risk metric value in those CDM typically decreases near the time of closest approach (TCA). Fig. 1 depicts the percentage of maneuvers that could have been avoided if, instead of taking the CDM at 12 hours prior to TCA for the decision, one would have taken the one at about 8 hours to TCA, as a function of the assumed probability of collision actionable threshold.

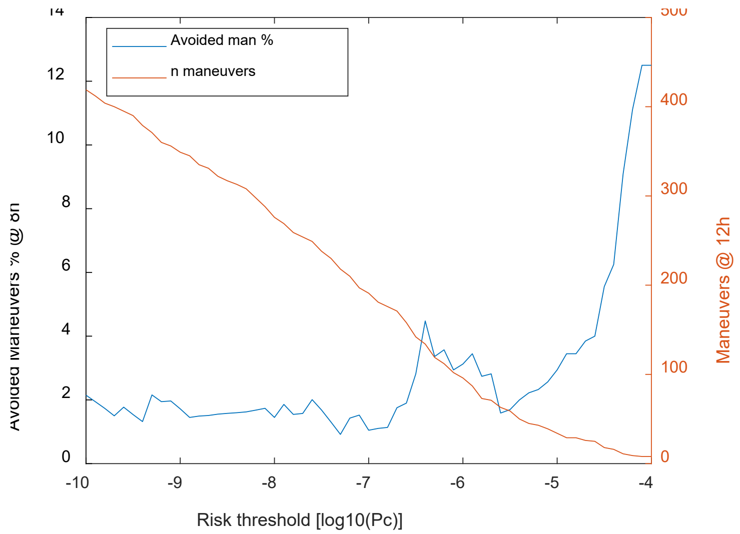

Fig. 1 Percentage of spared maneuver by waiting to 8h before taking a maneuver decision, with respect to 12h before conjunction, varying the decision threshold.

It is possible to conclude that up to ≈13% of the performed maneuvers would have been unjustified if the decision had been delayed until 8 hours to TCA. Such a timeline is, however, typically incompatible with ground operations due to the limited ground contact opportunities. Nonetheless, avoidance maneuver can be still effective up to 4 orbital periods before TCA ([8], [9]), meaning that the collision avoidance process could potentially wait up to the maneuver time (e.g., a few hours to TCA) before taking the final decision. Note that the analysis has been limited to those conjunctions where only the trajectory of the primary is improved by the latest CDM (which is consistent with ABACO concept of operations) identified as those cases for which the covariance matrix determinant of the secondary position at the last CDM is not reduced with respect to the previous one.

1.2 Operational Architecture

To develop ABACO operational flow, we adopted the STM architecture developed at NASA Ames based on the work of Sreeja Nag et al. [10]. Using their nomenclature, ABACO belongs to the S3 block (STM Service Supplier).

The S3 service interacts as a provider with the spacecraft operators subscribed to the service, as a peer with other S3 entities for traffic coordination, and as a user client when retrieving conjunctions alerts and auxiliary data. By traffic coordination, here it means the action of planning spacecraft paths and broadcasting it to the relevant entities (SSA service, other operators). The service has authority only on the subscribed operators whose interface allows to: i) access the ABACO board through the owner's satellite link, ii) obtain guidance on maneuver options (used for maneuver computations).

The latter point is crucial in that the maneuver design is not up to the spacecraft operator, still the operator has to have authority on how its satellite is moving. The guidance on maneuver consists of collecting from the operators their operational constraints so that the traffic manager can make the most effective coordination with minimum impact on the spacecraft mission.

As part of S3 service, the main task of ABACO is to ingest CDMs as conjunction alerts, verify its effective menace (RA process), integrate it with onboard observations, and plan a maneuver for it (COLA). The task of traffic coordination is delegated to the ground where the involved space actors can be reached without satellite visibility constraints. The ABACO board therefore relies on an additional ground-based function for maneuver dispatching and coordination between active satellites operators to complete the S3 service.

The proposed information flow between entities is reported in Fig. 2.

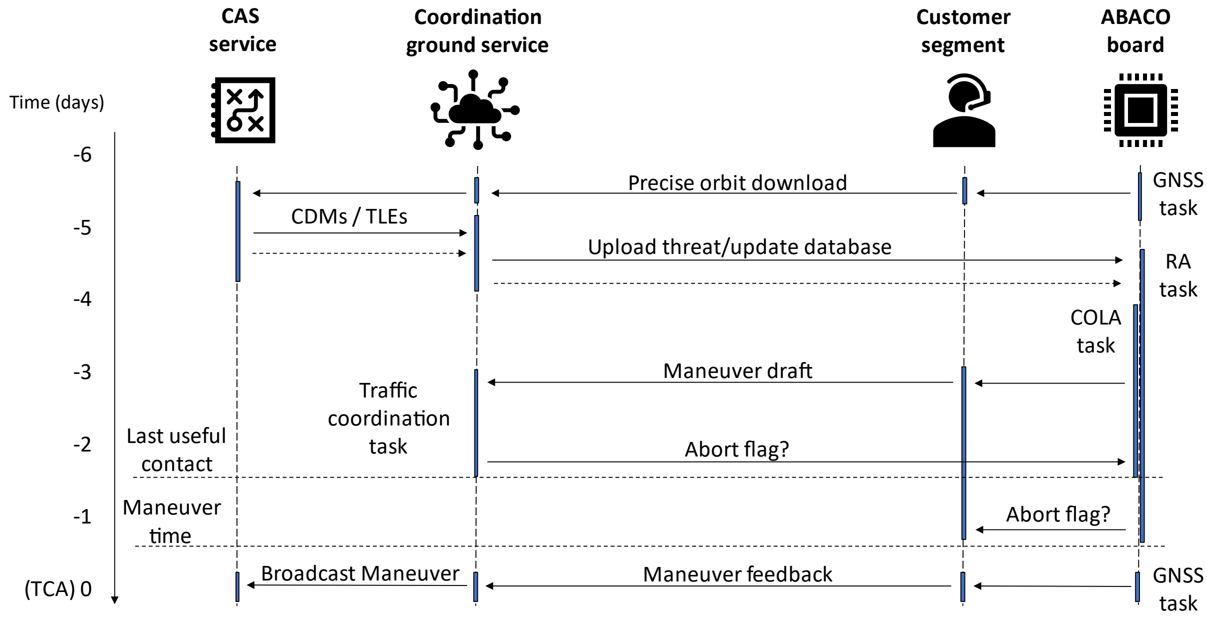

Fig. 2 Information flow of ABACO within the concept of operation

ABACO board implements three main functions within the S3 service: GNSS positioning, Risk Assessment, and Collision Avoidance.

The GNSS positioning function provides both the internal risk assessment function and the host platform with the precise ephemeris of the hosting spacecraft. The broadcasting of such information is also essential to the conjunction assessment service which uses it to scan for conjunctions within its central database.

The RA function is triggered from ground through the upload of a CDM for any conjunction. Once the primary assessment is completed using onboard data, a decision about the event is made.

If the event risk metric threshold is exceeded, the maneuver computation function is triggered, which provides a maneuver that accounts also for tertiary collisions. The maneuver is scheduled onboard for execution and sent to ground, through the host, for coordination and dispatching to ground tracking agencies. The event is kept monitored while ABACO reserves the decision to delete the scheduled maneuver as close as possible to the maneuver time. This reduces the quantity of the maneuvers, when the risk metric happens to decrease below the threshold.

The coordination function is not further detailed here because it is out of scope. However, the expected output from the coordinator is just an accept/decline decision. Neither the operator nor the ground service has therefore authority to change the maneuver details once it is proposed. In case of decline, ABACO sends an abort flag which deletes the scheduled maneuver.

Once the maneuver is completed the on-board estimated orbit is downloaded at the first contact opportunity. Those details are then communicated to the tracking agencies, to facilitate tracking of the object in the new orbit.

2. ABACO FUNCTIONAL DESCRIPTION

ABACO requires onboard electronics to autonomously execute reliable orbit predictions, collision risk computations, and general collision avoidance tasks. These tasks are performed by separate algorithms described in the following.

2.1 Navigation Algorithms

The Navigation (NAV) unit is composed of a GNSS receiver coupled with a Navigation Computer and is dedicated to the orbit determination task. It consists in processing GNSS (typically GPS and GALILEO) raw measurements and computing the host spacecraft position and velocity over time. ABACO implements two kinds of orbit determination strategies: a real time algorithm for fast orbit update, and a quasi-real time algorithm for a more accurate Orbit Determination (OD) and propagation, which is fundamental for the collision avoidance purposes.

The real time OD algorithm implements an Extended Kalman Filter (EKF) with a 4-th order Runge-Kutta propagator, and it is based on [11]. The quasi-real time OD is achieved with a non-linear Least Squares Fit (LSF), with differential corrections algorithm, based on [12], with an Adams-Cowell multi-step numerical propagator. The first is run by default from the navigation computer which, at the need, switches to the quasi-real time algorithm when a risk assessment process is ongoing.

Both algorithms are based on a reduced-dynamics approach, where empirical accelerations are estimated in order to absorb modeling errors. In the real-time EKF the dynamical model is particularly simplified, in order to allow for very fast computations and a good orbit accuracy, thanks to the abundance of GNSS measurements and the effectiveness of the empirical acceleration technique. Since collision avoidance processes require the capability to compute very accurate orbit predictions, the NAV unit is provided also with a more accurate propagator, which is also adopted for the least squares fit in a quasi-real time OD (see Table 1).

The parameters that are estimated in the OD process are position and velocity of the satellite (at the last measurement epoch), atmospheric drag and solar radiation pressure scaling coefficients, empirical accelerations in radial, along-, and cross-track directions (RTW, on a piecewise-constant approximation), receiver measurement clock corrections, and carrier phase biases.

For the two OD algorithms, the dynamical models adopted are composed of the same main terms, but for the quasi-real time we implemented a more accurate version, with a more sophisticated numerical propagator, compatible with the on-board hardware resources. The dynamical model terms are described in Table 1. A more complete and independent numerical simulator will be used as ground truth to generate measurements and test the performance of ABACO. To achieve this, in a first step, the numerical simulator will be used to verify the correct functioning of the ABACO algorithm. In a second step, suitable error models will be introduced in the ground truth, in order to validate the reliability of the ABACO orbit and covariance computations over a more realistic test case scenario. Consider covariance and scaling techniques are likely to be necessary, especially for low altitude regimes, to take into account modeling errors (mostly due to atmospheric drag) and improve covariance realism.

Table 1 Dynamical models characteristics of ABACO

	ABACO Reduced Dynamics Model (for Real-Time)	**ABACO Improved Dynamical Model (for Orbit Predictions and Quasi-Real Time)**
Gravity field Spher. Harm.	up to degree and order 70	up to degree and order 70
Tides	None	Solid Tides IERS-2010
SRP	Cannon-ball model [13]	Cannon-ball model [13]
Drag	Cannon-ball model, static density model Harries-Priester [13]	Cannon-ball model, NRLMSISE-00 density model
Empirical Accelerations	$\boldsymbol{a}_{RTW}$ every new obs (piecewise-constant over time update steps)	$\boldsymbol{a}_{RTW}$ piecewise-constant (30min-1h)
Earth Rotation Model	Analytical Model [13]	IAU – SOFA (IERS 2010)
Relativity	None	Post-Newtonian correction
Luni-Solar Perturb.	Analytical series Model [13]	Analytical series Model [13]
Numerical Propagator	4-th order Runge-Kutta	Adams/Cowell Multistep of order 10

The OD algorithms are able to process raw pseudo-range and carrier-phase data, typically in a dual-frequency ionosphere-free combination. Due to the on-board and real time constraints, the use of GNSS broadcast ephemerides is the most suitable option to compute precise orbits. Nonetheless, ABACO can also ingest the Ultra-Rapid products (half-predicted), ideally provided from ground with the same frequency and latency reported by the IGS service [14], and interpolated with trigonometric basis functions. The ABACO software can currently handle both GPS and Galileo measurements and ingest GPS and Galileo ephemerides.

In the current version of ABACO, no attitude and receiver antenna position computations are implemented, yet. Thus, in the current simulations, the assumption of a negligible distance between the antenna phase center and the spacecraft center of mass has been retained; this assumption will be relaxed in a later software version. In general, at least meter-level accuracy is the goal to be achieved with the current ABACO navigation algorithm, while more accurate navigation is expected to be possible in the near future with proper modifications, e.g., following the approach discussed in [15]. Indeed, in this recent paper, it is shown that the accuracy of the GALILEO and modernized GPS broadcast ephemerides are likely to reach sub-meter-level accuracy in orbits, potentially allowing for a 3Drms decimeter-level and sub-mm/s real-time on-board orbit determination accuracy. Thus, GALILEO and the modern GNSS constellations have high potential to allow for precise real-time positioning with broadcast ephemerides, and the need to upload other sources of precise ephemerides on-board in near-real time might not be necessary in our case.

The real-time algorithm processes one observation at a time and does not need to keep in memory data related to previous orbit updates, therefore it is not very demanding in terms of RAM usage, as opposed to the quasi-real time batch LSF. In order to contain the memory consumption, while the real-time is running, only a subset of observations is stored for batch least squares OD (which is triggered only when a CDM arrives), over a suitable time span.

2.2 Decision Algorithm

A classical tool in literature to assess the risk of a conjunction between a primary and secondary object is the probability of collision ([16], [17]). Such index, denoted as Pc, is typically computed for short-terms encounters under the following assumptions:

- The two objects have a rectilinear motion with constant velocities during the encounter (also known as high velocity encounter assumption);
- There is no uncertainty in the velocity during the encounter.
- The position uncertainty during the encounter is constant and equal to the value at the estimated conjunction epoch.
- The position uncertainties can be represented by a independent Gaussian distributions.
- The two objects are spherical with radii R_1 and R_2 respectively.

The first hypothesis is consistent when the time duration is just a few seconds, and the second one is justified since the velocity uncertainty is usually in the order of meters/second whereas the time duration of the encounter is small. Under these simplifications, the probability of collision can be evaluated through a three-dimensional integral which can be further reduced into a two-dimensional one across the plane perpendicular to the relative velocity vector, called the collision plane, or b-plane [16]. If the plane coordinate axes *x, y* are defined to be aligned to those of the combined (primary + secondary) positional covariance ellipse, the two-dimensional P_c integral reduces to:

$$P_c = \frac{1}{2\pi\sigma_x\sigma_y} \int_{-R}^{R} \int_{-\sqrt{R^2-x^2}}^{\sqrt{R^2-x^2}} e^{-\frac{1}{2}\left[\left(\frac{x-x_m}{\sigma_x}\right)^2+\left(\frac{y-y_m}{\sigma_y}\right)^2\right]} dydx$$

where $R = R_1 + R_2$ is the "hard-body radius" (HBR) centered at the primary object, x_m and y_m are the respective components of the projected miss distance, and σ_x^2 and σ_y^2 are the respective diagonal terms of the combined covariance.

A simple approach to rank a conjunction as dangerous is to compare the probability of collision with a pre-fixed lower threshold (a common value is 1E-4). Another possibility, even easier, is to set an analogous limit for the Mahalanobis distance:

$$d_m = \sqrt{\left(\frac{x_m}{\sigma_x}\right)^2 + \left(\frac{y_m}{\sigma_y}\right)^2}$$

stating that a collision is safe if d_m is greater than such a limit. A more sophisticated strategy based on [18] involving both the probability of collision and the Mahalanobis distance aims to control missed alarms that result in a collision (defined as "Type II" errors), ABACO aims at this last approach.

Once the HBR is known, a relationship can be established, on one hand between the upper bound of Type II error probability and the Mahalanobis distance screening threshold, and on the other hand between the latter quantity and the probability threshold. Therefore, if we set the desired miss detection rate upper bound, we can derive the corresponding probability threshold to be considered for assessing the risk of the current conjunction.

2.3 Maneuver Computation

A typical and effective mitigation action to avoid a collision with a secondary object is changing the primary's trajectory by thrusting (assuming that the primary is in charge of such an action). However, the decision to perform an evasive maneuver shall be made very carefully to avoid unnecessary loss of services, effort and propellant. A trade-off to be considered in practical situations is the one between the mitigation maneuver size and the maneuver execution time, as suggested in [6]. In fact, if the decision to act is made well in advance of TCA, usually a small maneuver is enough to avoid the potentially dangerous conjunction. On the other hand, since the risk of most close approach events drops off close to TCA due to additional tracking and improved state estimates and covariances, waiting could increase the likelihood that the need for a mitigation action disappears. Nevertheless, the decision to wait followed by subsequent messages and analysis confirming the risk would imply the need for a more invasive maneuver.

A close approach is generally notified by a sequence of CDMs starting several days before the TCA. In ABACO, there are two tunable time intervals $\Delta T_1 > \Delta T_2 > 0$ that define respectively how long before the TCA to begin a maneuver draft and to actually perform a maneuver (see Fig. 3). Typically, we assume 1 day for ΔT_1, and 4 orbital periods for ΔT_2. The idea is that the period [TCA-ΔT_1, TCA-ΔT_2] represents a communication window with the Ground segment during which the operators have room for coordination activities and, possibly, interrupt the procedure. When it takes less than ΔT_2 to TCA, if no abort message was received, ABACO updates the maneuver with the currently available information and actually performs it. Once the collision is avoided, a procedure to return to the nominal orbit is also foreseen.

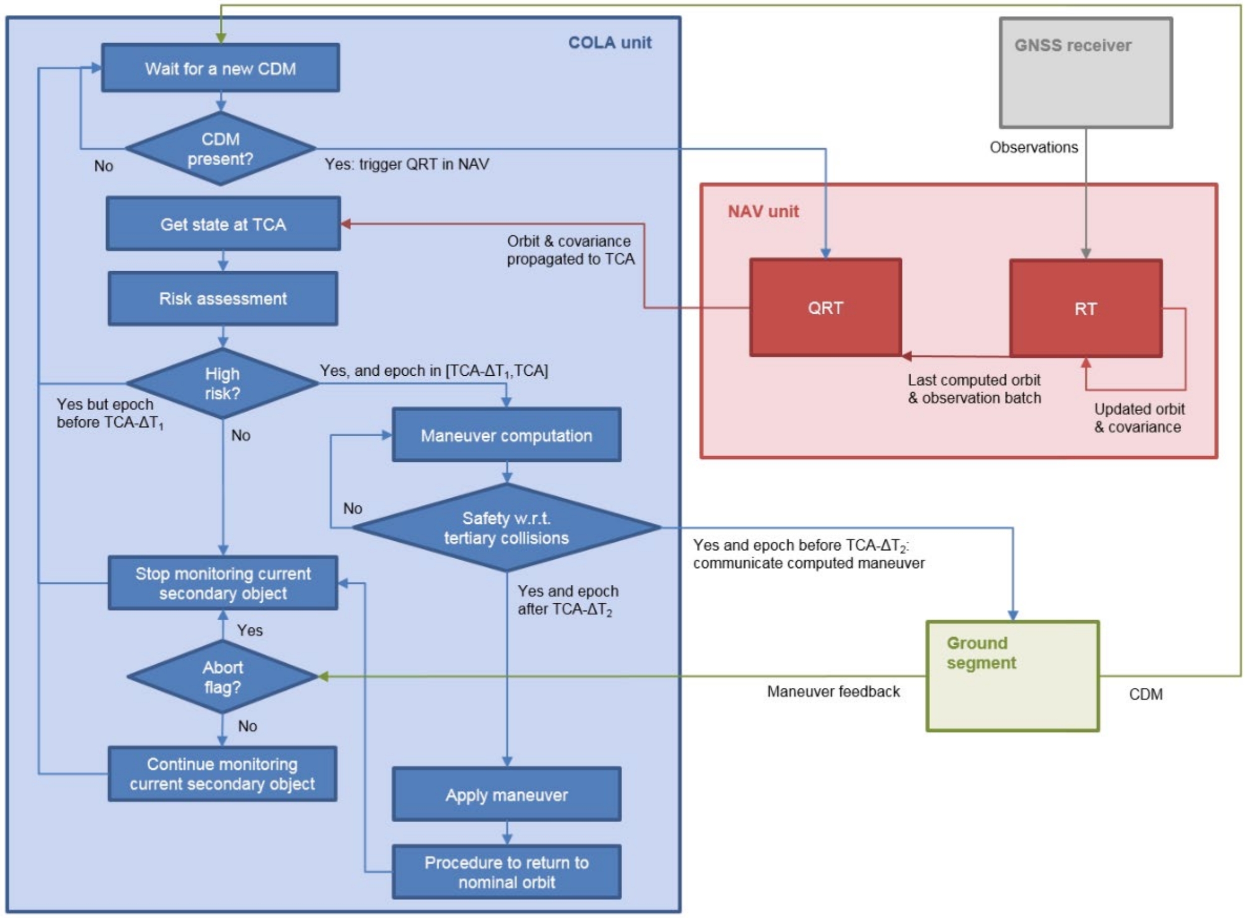

Fig. 3 Block diagram of the ABACO algorithm

Regarding the type of transfer, ABACO considers an impulsive maneuver computed according to [8]: in fact, since the collision avoidance maneuvers (CAMs) may have to be performed several times in the lifetime of a satellite, consuming an important amount of fuel and possibly hindering the mission, it is important to design maneuvers capable of saving resources and this is the reason why typically CAMs are computed through a constrained optimization problem. In [8], the constraint is expressed in terms of a maximum magnitude for the delta velocity, and two approaches are considered: the first one aims at maximizing the collision miss distance whereas the objective in the second one is to minimize the probability of collision. Both cases reduce to an algebraic eigenvalues and eigenvectors problem in a two-dimensional space, followed in the general case of a non-direct impact (that is, for a non-null minimum distance) by a one-dimensional iterative method.

Furthermore, the designed maneuver shall not give rise to new conjunctions. According to [6], a maneuver can be considered safe when it brings the violating conjunction down to a Pc of 3.2E-06 without introducing or raising any other conjunction Pc values above 1E-04. Therefore, the maneuver computation in ABACO is actually an iterative process: a first attempt of maneuver is obtained according to [8] to separate the trajectory of the primary from the one of the secondary. Then, ABACO compares the orbit the primary would have if this maneuver was performed with the orbits of a proper subset of other objects, scanning for the so-called tertiary conjunctions. If the maneuver leads to a potentially dangerous approach (evaluated in terms of Mahalanobis distance) with an object of that subset, then ABACO rejects it and computes another maneuver by adding a small heuristic correction on the impulse tangential component, and so on, until a safe maneuver is found. Eventually, the state of the primary if the safe maneuver would apply is communicated to ground, and operators have the opportunity to intervene. If no abort message is received, then ABACO performs the maneuver.

To account for tertiary conjunctions, ABACO shall be capable of predicting the trajectory of other objects than primary and secondary. This is achieved by storing on board the TLEs of up to 100 RSOs which are regarded as potentially dangerous, using SGP4 to propagate the trajectories, and screening their closest approaches with the primary's post-maneuver orbit. TLEs predictions are known to be affected by errors in the order of tens of kilometers for propagation horizons of a few days from epoch. To increase the fidelity of the tertiary conjunctions screening and provide a covariance estimate, ABACO uses neural networks (NN) trained to compensate for the SGP4 modeling errors ([19], [20]) and estimate the positional error covariance through a simple differencing technique [21].

Specifically, for a given tertiary object and a prediction epoch: i) the SGP4 is used to propagate to that epoch an ensemble of the last 30 available TLEs of that same object; 2) the NN is applied to reduce the propagation errors; iii) from the ensemble of corrected positions, the covariance is estimated with the differencing technique of [21]. From our tests, an ensemble of 30 TLEs offers a good compromise between statistical representativity and propagation error. In fact, the older the TLE the larger the propagation error of that sample within the ensemble.

The position error standard deviation obtained in this way is consistent (under Kolmogorv-Smirnov test) within the ensemble and stays below ≈10 km at 2 days-prediction. Its realism is however poor, indicating that statistically the real position may not obey the obtained covariance. This is the reason why TLEs and AI roles in ABACO have been confined to handle tertiary objects only, so that, in case of failure, the system can recover the conjunction as a primary one and avoid catastrophic collisions.

Inspired by Peng [19], ABACO implements a feedforward NN with 20 neurons in the hidden layer and ReLU activation function. The 67 input features are all available from TLEs history of an object and consist of times and differential position errors with respect to the latest TLE as in [19]. The NN output is a scalar representing the error prediction for SGP4 propagation along the radial, transversal and normal directions. Therefore, there shall be one network for each position component.

Because of the lack of a sufficient amount of real data, the network has been trained on synthetic data. Nonetheless, the simulated performance shows a good error prediction potential. Fig. 4, shows the results of the network-driven TLE prediction enhancement applied to a 1-year test set with 32 random LEO orbits. It can be seen how the residual error is reduced in a prediction window of 14 days for the radial and transverse components, while there is no significant gain on the normal component. For this reason, ABACO implements NNs for correcting the radial and transversal position components only.

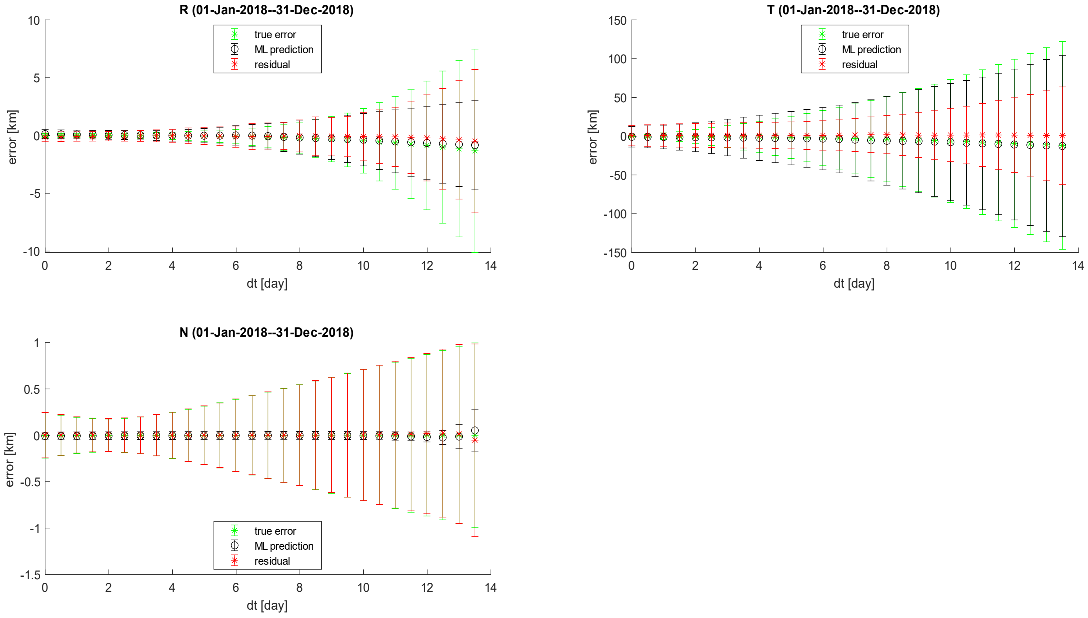

Fig. 4 Performance of the neural network on the synthetic test set (32 synthetic objects propagated for one year)

3. PHYSICAL DECRIPTION

The ABACO board developed for this project is a TRL 4 prototype i.e., an elegant breadboard to be validated in a laboratory environment. Although its design is not compliant with space hardware design, the computational power and the high-level interface protocols reflect those actually used in an operative environment to validate the operational concept.

Fig. 5 depicts the final breadboard designed for this work. On the left side, the main ABACO board can be seen featuring two Micro-Controller Units (MCU), the non-volatile memory banks, and the communication interfaces. On the right side of the figure, the commercial GNSS front-end can be seen, cabled onto the bottom part of the main ABACO board.

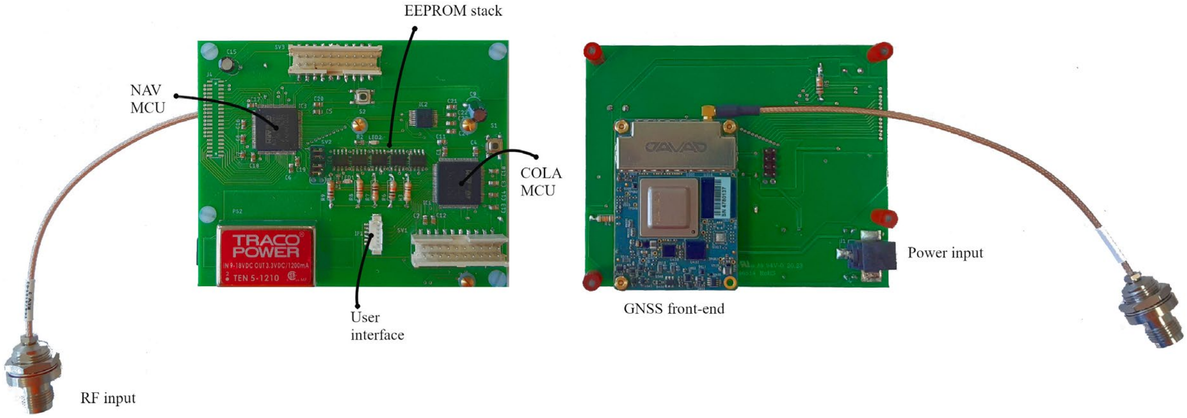

Fig. 5 Assembly of the final ABACO board prototype (top view on the left, bottom view on the right)

3.1 COLA and NAV processors

The computational resources critical to ABACO are computational speed and memory. The former follows from the consideration that, in a high-traffic situation, ABACO shall be capable of analyzing, on average, one conjunction every 40 minutes. Such estimate can be obtained from the event rate of a spacecraft in high traffic condition on SOCRATES [22], a free online tool providing conjunction assessment using TLEs. A Starlink satellite typically experiences a maximum of 5 new events/day. Suppose that each event is discovered, on average, 7 days in advance, then a total of up to 35 events is expected to be processed in 24h. To accommodate this requirement, an MCU operating at a clock frequency above 400MHz has been chosen.

With respect to the memory, the navigation algorithm has to retain in memory GNSS observations and process them as a batch for the least squares fit. Depending on the observable window used in the fit, the required memory can go from 300 kbytes up to 1.5Mbytes. Additionally, the maneuver design task makes use of TLEs for considering tertiary collisions and AI to enhance their orbit prediction accuracy. In a regime of high traffic (2e-6 units/km^3 [23]), ABACO should be capable of loading in memory TLEs for a maximum of about 100 objects (for the last 30 days) in optimizing an evasive maneuver through AI enhancement. This is the number of objects expected to enter in a 20km toroidal control volume about the spacecraft's LEO orbit. Even optimizing the format storing of TLEs, this implies loading about 200 kbytes of data.

These two requirements have driven the ABACO physical architecture to adopt two MCUs, one dedicated to the navigation task and the other dedicated to the collision avoidance computations. Splitting the two tasks between two physically separated units allows for the memory resources of each unit to be fully committed to the intended task.

Table 2 reports the relevant characteristics of the adopted STM32H742VG MCU, while the hardware architecture is depicted in Fig. 6. Two MCUs communicate through an SPI bus on which a 2 Mbytes EEPROM memory stack is connected for storing non-volatile, long-term information, such as TLEs and synaptic weights of the neural networks.

Table 2 Main features of the ABACO's MCU

Feature	Value
Max clock frequency [MHz]	480
RAM (effectively usable) [Kbytes]	384
Power consumption (@ 200MHz) [W]	0.25

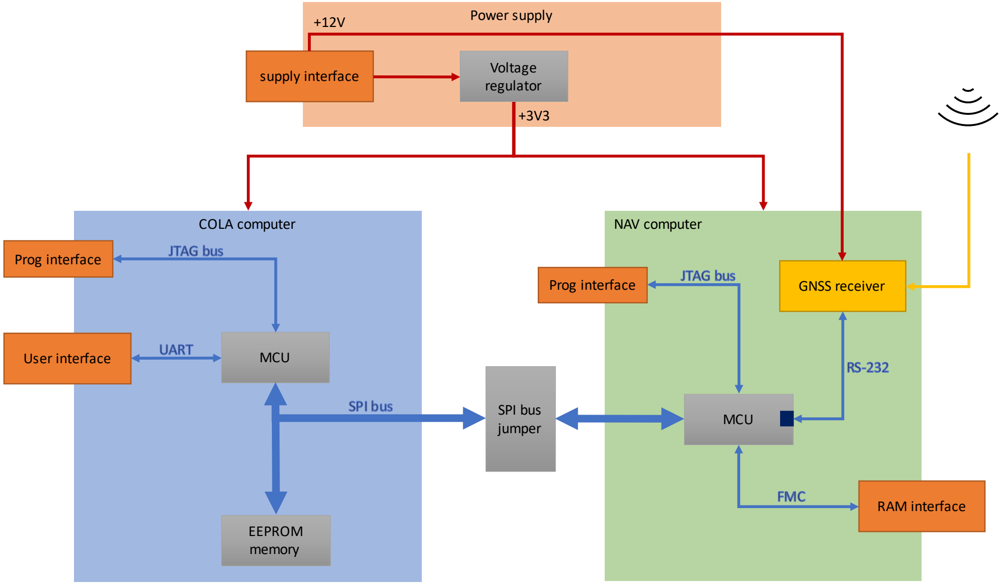

Fig. 6 Physical architecture diagram of ABACO

An external Random Access Memory (RAM) interface is also installed for future needs requiring more volatile memory. This is done leveraging the Flexible Memory Control (FMC) of the adopted MCU.

3.2 GNSS receiver

ABACO relies on a Commercial-Off-The-Shelf front-end to decode the GNSS signal from GPS and GALILEO constellations. It adopts a TR-2S receiver from JAVAD, a device having flight heritage. Table 3 summarizes the characteristics relevant to ABACO.

The receiver offers access to the raw GNSS observable and broadcast ephemeris used by ABACO's navigation algorithm. The data are delivered to the NAV MCU using a standard RS-232 protocol, as depicted in Fig. 6.

Table 3 Main JAVAD T-2S receiver characteristics

GNSS constellations	GPS, GLONASS, GALILEO, BeiDou, QZSS, SBASS, IRNSS
Time-to-first-fix	35s
Supply voltage	4.5 - 40V
Max power consumption	1.8W

3.3 External Interfaces

The power supply and data connections constitute all of the external interfaces used in operation. The power supply is a coaxial connector accepting unregulated voltage between 12V and 24V. The estimated maximum power consumption is 3.3W with the processors and GNSS front-end running at maximum capability.

The hosting spacecraft interacts with the board through a serial port implementing the UART physical layer protocol. At a higher level, the communication protocol is the Packet Utilization Standard (PUS) defined by the ECSS-E-70-41C [24], this allows the ABACO prototype to validate PUS applicability to its operational concept.
Table 4 reports the services and sub-services providers implemented by ABACO and those expected to be provided by the hosting platform.

Table 4 PUS services implemented by ABACO

	ABACO	**Hosting spacecraft**	
Service	**Subservice**	**Subservice**	**Usage**
Request verification	Acceptance and reporting Execution and reporting	Routing and Reporting	Check the integrity of a packet while the hosting spacecraft is just a bridge between ABACO and ground
Device Access	Device Access		Set and get ABACO parameters
Event Reporting	Event Reporting		Broadcasting of maneuvers and periodic precision ephemeris, as well as runtime errors
Large Packet Transfer	Large packet downlink Large packet uplink		Upload TLEs and binary files for neural networks. Download maneuver reports.
Time-based scheduling		Time-based scheduling	schedule and delete avoidance maneuver

4. VALIDATION CAMPAIGN

To verify that the ABACO board is able to evaluate conjunctions, compute maneuvers and properly interface with the operational scheme, hardware in the loop test campaign is foreseen. For the navigation algorithms in particular, the test campaign assesses the achievable accuracy in orbit estimation and propagation with the ABACO current algorithms implementation for the specific purposes of collision avoidance.

All the tests share a similar hardware setup as depicted in Fig. 7.

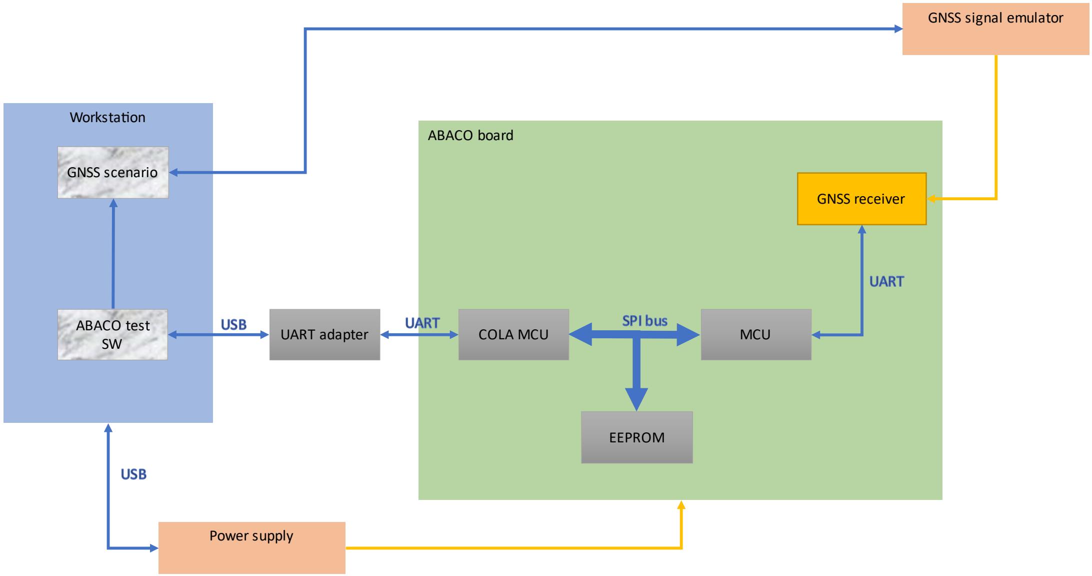

Fig. 7 Block diagram of the test setup architecture

A workstation is used to emulate the ABACO board user i.e. the ground interface through the hosting spacecraft. A commercial UART-to-USB adapter connects the workstation to the ABACO user interface.

A GNSS signal emulator (QA707 from QASCOMM, Italy) is used to generate the RF signal that can be decoded by ABACO's front-end. It is a software application running on a personal computer and leveraging a Software Defined Radio with precise external clock reference for streaming the RF signal to the front-end. The current version of the software is able to generate signals for spacecraft in low earth orbit (LEO) coming from GPS and GALILEO constellations in the bands L1 and L5 (or E1 and E5a). With the current hardware setup only L1/E1 or, alternatively, L5/E5a can be generated.

Finally, the power supply unit feeds electrical power to all the equipment, while its USB connection to the workstation allows logging the power consumption.

An outline of the objectives and contents of the tests is given in Table 5, the validation plan includes the following tests:

- navigation performance
- miss detection rate
- secondary conjunction avoidance
- operational failures
- full-chain test

The test campaign is expected to be performed in Fall 2023.

Table 5 List and short description of the tests for ABACO

Test Title	**Objective/Description**	**Configuration detail**	**Measured parameter**
Navigation Performance	Measuring the orbit determination accuracy of the real time and quasi-real time algorithms with respect to a simulated orbit. For quasi-real time, covariance consistency is also measured.	The GNSS signal emulator is providing the RF signal while the COLA MCU acts as a UART-to-SPI adapter for querying the NAV MCU.	- position error - Mahalanobis distance (from truth)
Miss detection rate	Measuring the miss detection rate produced by the decision algorithm when tested against a set of encounter scenarios generated from a synthetic dataset.	NAV unit is bypassed, COLA unit interfaces with the workstation to obtain the encounter data.	- miss detection rate
Tertiary conjunction avoidance	Assessing the capability of the system to design maneuvers that avoid collision with other objects than the secondary. A synthetic dataset is created containing examples of primary encounters with an associated avoidance maneuver and then tertiary objects are artificially positioned.	NAV unit is bypassed, COLA unit interfaces with the workstation to obtain the encounter data.	- final distance between the target and tertiary objects
Operational failures	Assessing the capability of the navigation algorithms to tolerate GNSS signal interruption with relative warnings are delivered to the user.	full configuration (Fig. 7)	- position error - list of warnings delivered to the user
End-to-end	Validating the operational workflow of ABACO. The workstation runs the software for creating the orbit scenario of a high-risk conjunction, generating de-facto a simulated SSA made available to ABACO. The scenario consists of: - the ABACO trajectory provided through QUASCOM simulator; - the menace trajectory provided with synthetic CDM; - the trajectory of nearby objects through the generation of synthetic TLEs from the simulated ground truth. The primary conjunction is considered avoided if the maneuver is scheduled and the risk metric with the “maneuvered” orbit and last available secondary info (CDM data) is below the threshold. Additionally, all the necessary warnings are issued to the user. The secondaries are considered avoided if the true minimum distance is larger than 10km, considering the ground truth orbits.	full configuration (Fig. 7)	- history of risk metric - distances from tertiary satellites - issued periodic precise ephemeris - issued maneuver scheduling

5. CONCLUSIONS

In this work, we have presented the operational concept and physical realization of ABACO, an autonomous board for avoiding spacecraft collisions within the GNSS service volume.

The operational concept is designed to shift the collision avoidance workload from human operators on ground to onboard, where data is timely available and there are no contact limitations. The process is designed to be automatic and transparent to the spacecraft operator.

The ABACO software algorithms allow for the estimation of collision risk integrating quasi-real time onboard measurements, maneuver design including tertiary objects, and a statistically safe maneuver decision strategy. These algorithms have been deployed on an elegant breadboard hardware prototype that features two MCUs implementing the navigation and COLA functions respectively. Preliminary tests suggest that ABACO software complexity is compatible with the available embedded hardware resources.

The test campaign for validating ABACO at TRL 4 will be performed in fall 2023. Tests will assess the navigation performance, miss detection rate, tertiary conjunctions avoidance, tolerance to GNSS signal outage, and the end-to-end operational workflow: results will be reported in future work.

6. ACKNOWLEDGEMENTS

We would like to thank QUASCOM for having granted us a free license for the QA707 GNSS signal simulator.

7. REFERENCES

[1] D. L. Oltrogge, "The “We” Approach to Space Traffic Management," in *SpaceOps Conferences*, Marsille, France, 2018.

[2] K. Merz, B. Bastida Virgili, V. Braun, T. Flohrer, Q. Funke, H. Krag, S. Lemmens and J. Siminsky, "Current Collision avoidance Service by ESA's Space Debris Office," *7th European Conference on Space Debris,* pp. 1-11, 18-21 April 2017.

[3] T. Pultarova, "SpaceX Starlink satellites had to make 25,000 collision-avoidance maneuvers in just 6 months — and it will only get worse," 2023. [Online]. Available: https://www.space.com/starlink-satellite-conjunction-increase-threatens-space-sustainability. [Accessed 18 August 2023].

[4] M. D. Hejduk, "Satellite Conjunction Assessment Risk Analysis for “Dilution Region” Events: Issues and Operational Approaches," 2019.

[5] . European Space Agency, "ESA spacecraft dodges large constellation," 2019. [Online]. Available: https://www.esa.int/Space_Safety/ESA_spacecraft_dodges_large_constellation. [Accessed 19 August 2023].

[6] F. Krage, NASA Spacecraft Conjunction Assessment and Collision Avoidance Best Practices Handbook, Cambridge, Massachusetts London, England: NASA, 2020.

[7] T. Uriot, D. Izzo, L. F. Simões, R. Abay, N. Einecke, S. Rebhan, F. Letizia, J. Siminski and K. Merz, "SPACECRAFT COLLISION AVOIDANCE CHALLENGE: DESIGN AND RESULTS OF A MACHINE LEARNING COMPETITION," *preprint,* 2020.

[8] C. Bombardelli, J. H. Ayuso and R. G. Pelayo, "COLLISION AVOIDANCE MANEUVER OPTIMIZATION," *AAS,* pp. 1-15, 2014.

[9] J. Hernando-Ayuso and C. Bombardelli, "Low-Thrust Collision Avoidance in Circular Orbits," *Journal of Guidance and Control,* vol. 44, no. 5, pp. 983-995, 2021.

[10] S. Nag, D. D. Murakami, N. A. Marker, M. T. Lifson and P. H. Kopardekar, "PROTOTYPING OPERATIONAL AUTONOMY FOR SPACE TRAFFIC MANAGEMENT," in *70th International Astronautical Congress*, Washington D.C., USA, 2019.

[11] O. Montenbruck and R.-B. Pere, "Precision real-time navigation of LEO satellites using global positioning system measurements," *GPS solutions,* vol. 12, pp. 187-198, 2008.

[12] O. Montenbruck, T. Van Helleputte, R. Kroes and E. Gill, "Reduced dynamic orbit determination using GPS code and carrier measurements," *Aerospace Science and Technology,* vol. 9, no. 3, pp. 261-271, 2005.

[13] O. Montenbruck, E. Gill and F. Lutze, Satellite orbits: models, methods, and applications, Berlin, Germany: Springer-Verlag, 2002.

[14] . International GNSS service, "Products," 2020. [Online]. Available: https://igs.org/products/. [Accessed 19 August 2023].

[15] O. Montenbruck and A. Hauschild, "Precise real-time navigation of LEO satellites using GNSS broadcast ephemerides," *NAVIGATION: Journal of the Institute of Navigation,* vol. 68, no. 2, pp. 2419-432, 2021.

[16] 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, pp. 23-35, March 1999.

[17] R. P. Patera, "General Method for Calculating Satellite Collision Probability," *JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS,* vol. 24, no. 4, pp. 716-722, July 2001.

[18] S. Alfano and D. Oltrogge, "Probability of Collision: Valuation, variability, visualization, and validity," *Acta Astronautica,* vol. 148, pp. 301-316, 2018.

[19] H. Peng and X. Bai, "Machine Learning Approach to Improve Satellite Orbit Prediction Accuracy Using Publicly Available Data," *The Journal of the Astronautical Sciences,* vol. 67, p. 762–793, 2019.

[20] G. Curzi, D. Modenini and P. Tortora, "Two-line-element propagation improvement and uncertainty estimation using recurrent neural networks," *CEAS Space Journal,* vol. 14, p. 197–204, 2022.

[21] J. Geul, E. Mooij and R. Noomen, "TLE uncertainty estimation using robust weighted differencing," *Advances in Space Research,* vol. 59, no. 10, pp. 2522-2535, 2017.

[22] T. Kelso, "SOCRATES Plus," 2022. [Online]. Available: https://celestrak.org/SOCRATES/. [Accessed 19 August 2022].

[23] A. C. Boley and M. Byers, "Satellite mega-constellations create risks in Low Earth Orbit, the atmosphere and on Earth," *Scientific Reports,* vol. 11, no. 1, p. 10642, 2021.

[24] . European Space Agency, "ECSS-E-ST-70-41C – Telemetry and telecommand packet utilization," 2016. [Online]. Available: https://ecss.nl/standard/ecss-e-st-70-41c-space-engineering-telemetry-and-telecommand-packet-utilization-15-april-2016/. [Accessed 10 April 2019].

[25] J. V. Cabrera, S. Nag and D. Murakami, "AN INITIAL ANALYSIS OF AUTOMATING CONJUNCTION ASSESSMENT AND COLLISION AVOIDANCE PLANNING IN SPACE TRAFFIC MANAGEMENT," *AIAA/AAS Spaceflight Mechanics Workshop, Maui, Hawaii,* pp. 1-11, 2019.