3rd IAA Conference on Space Situational Awareness (ICSSA)

4th-6th April 2022, Tres Cantos, Madrid, Spain

IAA-ICSSA-22-0000
An Intelligent System for Robust Decision-Making in the All-vs-All Conjunction Screening Problem

Luis Sánchez[1], **Emma Stevenson**[2], **Massimiliano Vasile***[1], **Victor Rodríguez-Fernández**[2], **David Camacho**[2]

[1] *Aerospace Centre of Excellence, University of Strathclyde, 75 Montrose Street, G1 1XJ, Glasgow, United Kingdom. [luis.sanhcez-fdez-mellado; massimiliano.vasile]@strath.ac.uk*

[2] *Universidad Politécnica de Madrid, Calle de Alan Turing, 28038 Madrid, Spain. [emma.stevenson; victor.rfernandez; david.camacho]@upm.es*

Keywords: *Space Traffic Management, Artificial Intelligence, Robust Decision-making, Conjunction Assessment, Collision Avoidance Manoeuvre.*

The progressive increase of traffic in space demands new approaches for supporting automatic and robust operational decisions. CASSANDRA, Computational Agent for Space Situational Awareness aNd Debris Remediation Automation, is an intelligent system for Space Environment Management (SEM) intended to assist operators with the management of space traffic by providing robust decision-making support. This paper will present the automatic conjunction screening and collision avoidance manoeuvre pipeline within CASSANDRA, connecting the some of CASSANDRA's modules: Automated Conjunction Screening (ACS), Robust State Estimation (RSE), Intelligent Decision Support System (IDSS) and Collision Avoidance Manoeuvres (CAM). The pipelines allows to screen the catalogue to detect potential conjunctions, perform a detailed analysis of the encounter accounting for uncertainty (aleatory and epistemic) and new observations, provide robust decisions based on the available information and, if necessary, proposed robust optimal CAMs and analyse the impact of the new orbit on the background population. This paper will present the pipeline described above along with an example that illustrates how CASSANDRA can be used to generate robust decisions on the execution of CAMs in an automated way.

1. Introduction

With the progressive increase of space traffic and the growth of the number of resident objects in the space environment around the Earth comes the need for new tools to deal with the management of operations in orbit. The automation of operations, both on-board and on ground, is progressively becoming a necessity. In Space Traffic Management (STM), the increasing number of catalogued objects and conjunctions events, combined with the uncertainty associated with the estimation of the probability

*Corresponding author.

of collisions, have created the need for robust and automated decision systems. Artificial Intelligence (AI) can play an important role in this context. It can provide a quick response to complex scenarios, bypassing complex and lengthy calculations, and can support robust decision making by integrating several pieces of information. The application of AI to STM has emerged as a promising area of research [1], with possible uses stretching from efficient catalogue screening [2], to the classification of high-risk conjunction events [3], the planning and implementation of collision avoidance manoeuvres [4, 5], and offering support in reducing the workload of space operators [6].

In this paper, we present an automated conjunction assessment pipeline as a part of CASSANDRA (Computational Agent for Space Situational Awareness aNd Debris Remediation Automation), an intelligent system for providing robust decision-making support to operators in the context of STM. This system aims to close the loop of conjunction screening, risk assessment, manoeuvre screening, decision-making and manoeuvre execution, with the aim of allocating mitigation actions to avoid collisions in orbit in a robust and automated way.

An important element of such a system is its ability to cope efficiently with the scales associated with current, and future, space object populations. This is especially relevant in the so-called *All-vs-All* problem, which considers all possible pairs of catalogued objects, both active and debris, and is therefore crucial to space environment management (SEM), but a computational challenge owing to the vast and growing number of possible conjunction pairs. Another key element when dealing with the automation of high-risk scenarios is the treatment of uncertainty. In the case of STM, uncertainties in the state estimation and orbit propagation translate into an uncertainty on the probability of collision between two objects, which is critical for effective decision making. In this work, we propose a framework that combines a robust approach to uncertainty quantification, accounting for aleatory and epistemic uncertainty at each step of the conjunction assessment procedure, with the use of AI techniques to aid in efficient conjunction screening, manoeuvre screening, and decision making.

The paper is structured as follows: in Section 2, a overview of CASSANDRA and its modular architecture is presented. Section 3 includes an explanation of the Automated Conjunction Screening (ACS) module in two different modes: All-vs-All and One-vs-All (manoeuvre screening). In Section 4, the modules responsible for the uncertainty propagation, decision making and execution of the suggested action are explained: the Robust State Estimator (RSE), the Intelligent Decision Support System (IDSS) and the Collision Avoidance Manoeuvre (CAM) modules, respectively. Section 5 presents the proposed pipeline for the integration of these modules, and we illustrate the workflow of the system for analysing a set of potential encounters within a catalogue with an example. The paper concludes in Section 6 with the final remarks and future work.

2. CASSANDRA

CASSANDRA, Computational Agent for Space Situational Awareness aNd Debris Remediation Automation, is an intelligent system for SEM intended to assist operators with the management of space traffic by providing robust decision-making support. CASSANDRA has a modular architecture, as shown in Fig. 1. Each of the modules addresses a specific problem related to SEM: radar observation, catalogue screening, space weather forecasting, anomaly detection, robust decision-making, robust state estimation, collision avoidance manoeuvres, re-entry analysis and fragmentation analysis.

With this approach, each module can be developed independently and work as stand-alone tools, but at the same time can be integrated to work together in more complex problems. The modular architecture also enables the incorporation of new modules and functionalities to the system by just defining the interface (i.e., inputs and outputs) between the module and CASSANDRA, avoiding the complicated task of relating new modules with all pre-existing ones. In the following, a brief explanation is presented for each of the modules which are currently available in the CASSANDRA framework.

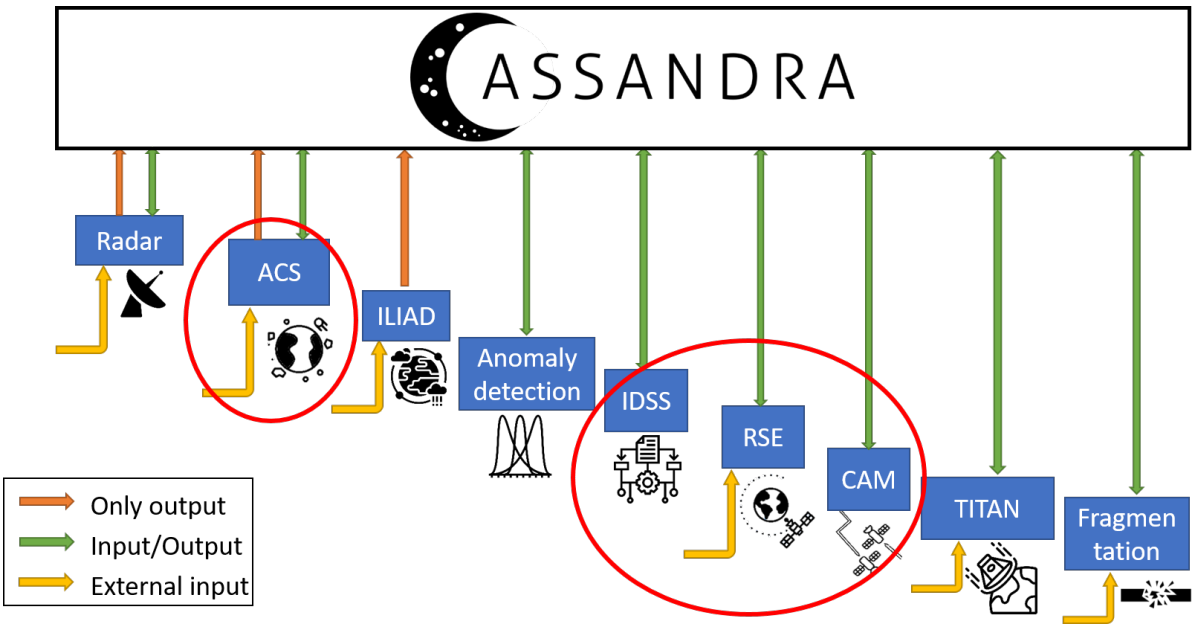

Figure 1: CASSANDRA and its modules. The modules developed and integrated in this work are circled in red.

- *Radar*: Responsible for acquiring radar measurements of space objects and their associated position uncertainty.
- *Automated Conjunction Screening* (ACS): AI-based module for predicting close encounters and generating conjunction alerts, both in the all-vs-all [2] and one-vs-all scenarios. More details will be given in Section 3.
- *InteLiIgent Atmospheric Density modelling for space operations* (ILIAD): AI-based system for space weather forecasting [7] and atmospheric density modelling [8].
- *Anomaly Detection*: AI-based module for predicting unmodelled terms on the dynamics [9]. Potential uses can be the detection of manoeuvres or alterations on perturbing forces.
- *Intelligent Decision Support System* (IDSS): AI-based decision making system for supporting operators in the event of a close approach [3]. Its output may suggest the execution of an avoidance manoeuvre or the acquisition of more observations. More details will be given in Section 4.2.

- *Robust State Estimator* (RSE): Module for robust estimation and propagation of an object's state and uncertainty (aleatory and epistemic), as well as functionalities for updating the state when receiving new observations [10]. More details will be given in Section 4.1.
- *Collision Avoidance Manoeuvre* (CAM): Module for computing the optimal CAM accounting for aleatory and epistemic uncertainty [4, 11]. More details will be given in Section 4.3.
- *TITAN*: Module responsible for the analysis of re-entry events, including the estimation of re-entry time and on-ground risk [12].
- *Fragmentation*: Module taking care of the fragmentation events of space objects.

In this work, we have focused on four of these modules: the ACS, RSE, IDSS and CAM modules. We have developed new capabilities for some of these, and combine them here to form an integrated system able to automatically screen a space object catalogue to detect possible encounters, which will be further analysed, and propose and execute remediation actions when required. In the two following sections (Sections 3 and 4), we provide a detailed description of these modules, and their new features. In Section 5, we then detail the pipeline of the integrated system, and present an example to illustrate the workflow.

3. Catalogue screening

In this section, we give details on the Automated Conjunction Screening (ACS) module, which is designed to predict close encounters between catalogued objects. This is typically the first stage of the conjunction assessment procedure, and is responsible for determining which (and when) object pairs fall within a given screening volume over a given screening period. These pairs are then subjected to a more detailed risk assessment to establish if mitigation actions are required to avoid a possible collision. These subsequent steps have dedicated modules within CASSANDRA, and are discussed in more detail in Section 4.

The ACS module has two actionable modes: *All-vs-All* and *One-vs-All*, which are described in more detail in Section 3.1 and Section 3.2, respectively.

3.1. All-vs-All

The All-vs-All mode was designed for fast inference in large-scale conjunction screening, specifically in the context of the All-vs-All problem, where conjunctions between all possible sets of catalogued objects, both active and debris, are considered. This scenario is computationally challenging, scaling quadratically with the growing number of catalogued space objects, and already today must cope with hundreds of millions of object pairs. To handle these scales efficiently, the ACS was developed using machine learning (ML) and deep learning (DL) AI techniques, which are promising in this context owing to their ability to process and exploit large datasets, infer hidden correlations and also reduce computational time during model prediction.

In line with the particular demands for efficiency of the All-vs-All mode, the task of automated conjunction screening was framed as a *tabular* machine learning *classification* task. This implies that, given a set of object pairs and their corresponding initial

states, the resulting model will predict whether or not these pairs will be involved in a conjunction over the given screening period (binary label).

In this work, we employ a feed-forward neural network for this task, which was trained using a realistic catalogue-wide conjunction database generated by the CNES BAS3E space surveillance simulation framework [13] and input two-line element set (TLE) catalogue. Based on the initial state vectors of the object pairs, the model is trained to predict which of the pairs will undergo a close encounter (as defined by a 20 km spherical safety volume) over the next 7 days. These predictions are designed to serve as an initial first filter for the catalogue, and thus particular attention is given to the number of misdiagnosed conjunction cases which are wrongly, and potentially critically, discarded by the screening process. Various methods were employed for forcing the importance of these cases including class rebalancing, weighted loss functions, and the tuning of the class probability threshold to ensure that only cases for which the model has high confidence are rejected. This threshold can be translated into a constraint on the number of acceptable missed conjunctions by the model, and therefore to an operator decision. More details on the model, underlying database and training procedure can be found in [2].

The primary output of the All-vs-All mode is thus a list of risky object pairs requiring further consideration. Following from this, the Time of Closest Approach (TCA) can be found, and the list of objects further refined through the inclusion of uncertainty data and evaluation of the collision risk in subsequent modules (Section 4). At this stage, information on the properties of the objects involved in the conjunctions, for example whether they are operational, can also be retrieved from catalogues such as ESA's DISCOS [14], which can be used alongside the TCA to prioritise the analysis of certain object pairs. This information can also be used to ultimately determine an appropriate course of mitigation action, such as a CAM (as considered in this work), or even techniques such as "just-in-time" collision avoidance for non-manoeuvrable debris-debris conjunctions [15].

3.2. One-vs-All

A second One-vs-All mode for the ACS was additionally developed for this work. This mode was designed to provide more detailed information on possible conjunctions, including the TCA and miss distance of individual events. As such, this mode could be employed as a second higher-fidelity filter following the All-vs-All mode, or, as explored in this work, for manoeuvre screening. In this latter scenario, the new orbit of an object undertaking a CAM to avoid a specific conjunction (the "one"), should be screened against the space object catalogue (the "all") to ensure that no additional conjunctions will be incurred. This screening should be used to decide whether the proposed CAM, which may be optimal under given constraints to avoid the original conjunction, should go ahead, or whether a different solution should be found. This scenario is explored in Section 5.

For this task, we build on previous works by the authors [16] to predict the future behaviour of the B-parameter. The B-parameter is the norm of the relative distance vector between two objects in the so-called B-plane or Body plane [17]. Under the assumptions of a short-term encounter, the B-plane can be defined as the plane perpendicular to the relative velocity of the primary object at the encounter time, such that it contains both objects. This definition of the B-parameter is therefore equivalent to the miss distance used in the calculation of the probability of collision [18], and thus

this quantity gives direct information on the proximity of two objects, with low values indicating a close encounter (assuming positional uncertainties are not considered).

As the B-parameter for a given object pair is dependent on the relative position and velocity vectors of both bodies, and therefore varies in time, we phrase the prediction of this quantity as a *time series* machine learning *regression* task. Regression tasks in machine learning consist of predicting a continuous outcome y based on the value of one or more predictor variables x. In our case, both x and y are time series structured data, with x consisting of the 6-channel ephemeris data of each object (12 variables in total), and y the future evolution of the B-parameter. In machine learning terms, this could be viewed as a univariate forecasting task with auxiliary variables, but this case is unusual in that the history of the forecast variable does not feature as an input to the predictive model. Nevertheless, to describe this problem, we make use of the forecasting terms *horizon*, the number of time steps to predict in y, and *lookback*, the number of time steps used to create the prediction from x.

In this work, we trained our predictive model using a synthetic database which comprised the ephemeris data of 200 objects over 7 days, with a timestep of 180 seconds, and the corresponding B-parameters for each pair (19,900 in total), calculated at each timestep. The initial parameters of these objects were restricted to the orbit regime given in Table 1. This range of orbits was restricted due to computational considerations, but enables us to present a proof of concept of this new mode, and demonstrate the pipeline of the integrated system described in this work (Section 5.2).

Table 1: Bounds on the initial orbital parameters of the orbits used to train the *One-vs-All* mode of the ACS. Units in km and rads.

	Lower bound	Upper bound
SMA	6900	7200
Eccentricity	10^{-5}	$5 \cdot 10^{-3}$
Inclination	1.22173	1.91986
RAAN	0	2π
Arg. Perigee	0	2π
True Anomaly	0	2π

The choice of the length and configuration of the lookback and horizon windows is application dependent. For the case of manoeuvre screening, it is the new (future) orbit of an object following a CAM that needs to be considered, and not the past history of the object which is typically associated with the forecast lookback. As such, our lookback needs to be based on the post-manoeuvre propagated state. We choose the model input to be the 12 ephemeris variables of each object pair over a 1-day period post-CAM. We then seek to predict the evolution of the B-parameter for each pair over a 3-day period post-CAM. This allows us to increase the quantity of training data by using a sliding window over the 7 available days in the database, generating 20 training examples for each object pair. An example is illustrated in Fig. 2, with truncated sections of the lookback and horizon windows for readability.

This dataset was then divided following an 80% to 20% splitting strategy into training and validation sets, ensuring that overlapping windows were not present in both to prevent data leakage. While the training set is used to train the model directly, the validation set is used to evaluate the performance of the model during training to en-

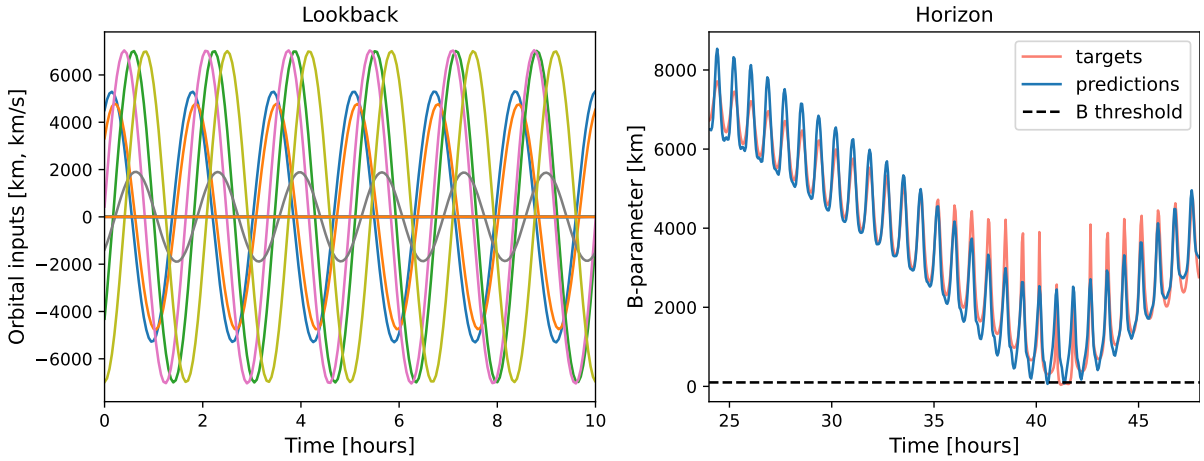

Figure 2: Example of the input (left, lookback) and output (right, horizon) of the ACS One-vs-All mode for truncated sections of the 1-day lookback and 3-day horizon.

sure that the model is generalising well to unseen data, and not merely memorising the training data, as well as to tune higher level hyperparameters that are pre-set by the user and not learnt by the model.

In this work, we employ the InceptionTime architecture [19] for training our predictive model. This is, to the best of our knowledge, the best deep neural architecture for time series tasks among the family of 1-dimensional Convolutional Neural Networks (CNNs). We chose this family of networks as they are most suitable for capturing short-term patterns in the data, such as orbit oscillations, since they compute features using sliding convolutional filters. Deep learning architectures such as this have a higher complexity and representation capacity compared to simpler tabular architectures, but come at the cost of increased computational expense as a result, motivating the two-mode nature of the ACS.

The training configuration used was as follows, for which we employed the `tsai` [20] library, which implements deep learning techniques and architectures for time series data. First, we build our neural network using six sequential inception modules to accept 12-channel inputs and a 1-channel output. The input data is standardised, transforming each variable to have zero mean and unit variance to account for differences in units and scales. For the loss function (the objective to be minimised while training), we use the nominal regression metric, the Root Mean Square Error (RMSE). For the optimiser (the component that updates the parameters of the neural network), we employed Ranger, an extension of the popular Adam optimiser that combines both RAdam and Lookahead optimisation algorithms. A dynamic learning rate (which determines the factor by which the weights of the neural network are updated in each training iteration) was chosen following a one-cycle schedule with cosine annealing. The model was then trained subject to an early stopping criteria, whereby training is terminated once over-fitting becomes evident on the validation set, and the best performing model (based on the validation loss) taken.

This model can then be applied to new data to obtain the predicted evolution of the B-parameter for different possible conjunction pairs. Once this has been obtained for a given pair, the TCA associated with possible conjunction events can then be identified by considering crossing points of the B-parameter with an operator defined threshold

(e.g. 20 km), as illustrated in Fig. 2. The output of the One-vs-All mode is thus a list of conjunction events, the objects involved, their TCA and miss distance.

4. Uncertainty propagation and decision making

In this section, the Robust State Estimation (RSE), Intelligent Decision-Support System (IDSS) and Collision Avoidance Manoeuvre (CAM) modules are explained. These modules constitute the core of the precise conjunction analysis and decision making section of CASSANDRA. In [11], a successfully integrated and simplified version of CASSANDRA, comprising only these three modules, was presented, including an illustrative example of the pipeline for a Collision and a No-collision scenario.

4.1. Robust State Estimation

The RSE is the module in charge of the precise state propagation, but more importantly, of the uncertainty propagation and the state updating when observations are received. This module is able to account both for aleatory and epistemic uncertainty. The RSE provides, in addition to the propagated state, an estimation of the confidence interval of the probability of collision and other parameters related with the encounter. These intervals of confidence will be used later by the IDSS to make the decision. A complete explanation of the RSE can be found in [10].

4.1.1. Robust formulation

For providing robust decision-making, the system should account for epistemic uncertainty. In the next lines, a brief explanation on how the filtering problem is reformulated to account for this uncertainty is included.

The problem can be formulated as:

$$\dot{\mathbf{x}} = f(t, \mathbf{x}, \mathbf{d}), \tag{1}$$

$$\mathbf{y}_k = h(t_k, \mathbf{x}_k, \varepsilon_k), \tag{2}$$

where Eq. (1) is the equation of motion, with t the time, $\mathbf{x}$ the state vector and $\mathbf{d} \in \mathbb{R}$ model parameters, with initial conditions $\mathbf{x}_0 = \mathbf{x}(t_0)$, and where Eq. (2) represents noisy observations of the state vector at certain times $t_k, k = 1, .., M$, where $t_k < t_{k+1}$ and ε_k the measurement noise. The initial condition, the model parameters and the observations are modelled as random variables: $\mathbf{X}_0$, $\mathbf{D}$, $\mathbf{E}_k$, being $\mathbf{x}_0$, $\mathbf{d}$ and ε_k specific realisation within their respective sample spaces: $\Omega_{\mathbf{x}_0}$, $\Omega_{\mathbf{d}}$ and Ω_{ε}. The probability density function (pdf) of the initial state and the dynamic model parameters would be represented by $p(\mathbf{x}_0)$ and $p(\mathbf{d})$, and the likelihood of the observations is expressed as $p(\mathbf{y}_k, |\mathbf{x}_k)$. The pdf of the state at time t_k given the observations acquired up to this time is $p(\mathbf{x}_k|\mathbf{y}_{1:k})$.

If epistemic uncertainty is introduced, the pdfs are no longer precisely known, but defined within imprecise sets, which can be expressed as:

$$\begin{aligned} \mathcal{P}_{x_0} &= \{p(\mathbf{x}_0; \boldsymbol{\lambda}_0) \,|\, \boldsymbol{\lambda}_0 \in \Omega_{\lambda_0}\}, \\ \mathcal{P}_{x_k|x_{k-1}} &= \{p(\mathbf{x}_k|\mathbf{x}_{k-1}; \boldsymbol{\lambda}_{x_k}) \,|\, \boldsymbol{\lambda}_{x_k} \in \Omega_{\lambda_{x_k}}\}, \\ \mathcal{P}_{y_k|x_k} &= \{p(\mathbf{y}_k|\mathbf{x}_k; \boldsymbol{\lambda}_{y_k}) \,|\, \boldsymbol{\lambda}_{y_k} \in \Omega_{\lambda_{y_k}}\}, \end{aligned} \tag{3}$$

so that:

$$\begin{aligned} \mathbf{X}_0 &\sim p(\mathbf{x}_0; \boldsymbol{\lambda}_0) && \in \mathcal{P}_{x_0} \\ \mathbf{X}_k &\sim p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}; \boldsymbol{\lambda}_{x_k}) && \in \mathcal{P}_{x_k|x_{k-1}} \\ \mathbf{Y}_k &\sim p(\mathbf{y}_k \,|\, \mathbf{x}_k; \boldsymbol{\lambda}_{y_k}) && \in \mathcal{P}_{y_k|x_k}, \end{aligned} \tag{4}$$

for $k = 1, \ldots, M$, where $\boldsymbol{\lambda}_k = [\boldsymbol{\lambda}_0, \boldsymbol{\lambda}_{x_{0:k}}, \boldsymbol{\lambda}_{y_{0:k}}] \in \Omega_{\lambda_k}$ are the epistemic parameters for the initial position, model parameters and observations, respectively. Similarly, the epistemic distribution of the state at time t_k given the observation up to that time can be expressed as:

$$\mathcal{P}_{x_k} = \{p(\mathbf{x}_k \,|\, \mathbf{y}_{1:k}; \boldsymbol{\lambda}_k) | \boldsymbol{\lambda}_k \in \Omega_{\lambda_k}\}. \tag{5}$$

4.1.2. Uncertainty propagation

To compute the generic posterior Eq. (5), two steps are executed iteratively from t_0 to t_k:

- *uncertainty propagation*: the posterior distribution at time t_{k-1} is propagated to time t_k through mapping

$$p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1}) \rightarrow p(\mathbf{x}_k|\mathbf{y}_{1:k-1}) \ . \tag{6}$$

  The resulting distribution is called prior at time t_k. This propagation is performed in two steps, detailed here below: the propagation of the polynomial representation of all possibles states at time t_k as a function of the sates at time t_{k-1}, followed by the propagation of particles by inexpensive polynomial evaluations. In this way, we reduce the cost of the particle propagation, which is the most expensive step [10, 21].

- *observation update*: at discrete instances, the prior distribution is updated to incorporate the last observation according to the unnormalised sequential Bayes' rule

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) \propto p(\mathbf{y}_k|\mathbf{x}_k)\, p(\mathbf{x}_k|\mathbf{y}_{1:k-1}) \ . \tag{7}$$

  The equation is evaluated for each particle to compute the corresponding weight.

4.1.3. Optimisation and estimation

In addition to uncertainty propagation, the RSE can provide the expectation on any variable of interest, $\phi(\mathbf{x}_{0:k})$, function of the propagated state (i.e. probability of collision).

$$\mathbb{E}\left[\phi(\mathbf{x}_k)\right] = \int_{\mathbf{\Omega}_{x_k}} \phi(\mathbf{x}_k)\, p(\mathbf{x}_k|\mathbf{y}_{1:k})\, d\mathbf{x}_k \ . \tag{8}$$

This expectation depends on the on the posterior distribution. Due to the epistemic uncertainty, the expectation is not single-valued, but interval-valued. The RSE is then able to provide the upper and lower bound of the expectation as a function of the epistemic parameters: $\underline{\mathbb{E}} = \min_{\lambda \in \Omega_\lambda}(\mathbb{E}\left[\phi(\mathbf{x_k})\right]), \overline{\mathbb{E}} = \max_{\lambda \in \Omega_\lambda}(\mathbb{E}\left[\phi(\mathbf{x_k})\right])$.

The integral involving the expectation computation has no closed form solution due to the non-parametric nature of the posterior pdf for generic distributions. In addition, it is not practical to draw samples from non-parametric distributions to obtain a numerical approximation of the integral. The chosen alternative is to use importance sampling [10] to construct an estimator of the expectation:

$$\mathbb{E}\left[\phi(\mathbf{X}_{0:k}) \,|\, \mathbf{y}_{1:k}; \boldsymbol{\lambda}\right] \approx \hat{\theta}(\mathbf{X}_{0:k}; \boldsymbol{\lambda}_k) = \sum_{i=1}^{N} \hat{w}_k^{(i)}(\boldsymbol{\lambda}_k)\, \phi(\mathbf{x}^{(i)}), \tag{9}$$

This estimator, which exploits the sequential nature of the problem, is a weighted sum of $\phi(\mathbf{x}^{(i)})$ evaluated on fixed samples, and importance weights that depends on

the epistemic parameters. Thus, the function of interest can be pre-computed with a fixed number of samples drawn for a proposal posterior distribution, $\pi(\mathbf{x}_k \,|\, \mathbf{x}_{0:k-1}, \mathbf{y}_{1:k})$, which does not depend on the epistemic parameters. Only the weights depend on the epistemic parameter:

$$\begin{aligned} w_k^{(i)}(\boldsymbol{\lambda}) &= \frac{p(\mathbf{y}_k \,|\, \mathbf{x}_k^{(i)}; \boldsymbol{\lambda}_{y_k})\; p(\mathbf{x}_k^{(i)} \,|\, \mathbf{x}_{k-1}^{(i)}; \boldsymbol{\lambda}_x)}{\pi(\mathbf{x}_k^{(i)} \,|\, \mathbf{x}_{0:k-1}^{(i)}, \mathbf{y}_{1:k})}\; \hat{w}_{k-1}^{(i)}(\boldsymbol{\lambda}) \\ \hat{w}_k^{(i)}(\boldsymbol{\lambda}) &= \frac{w_k^{(i)}(\boldsymbol{\lambda})}{\sum_{j=1}^{N} w_k^{(j)}(\boldsymbol{\lambda})}, \end{aligned} \tag{10}$$

In this way, it is possible to split the problem, particles can be drawn and propagated once, while the optimisation over the epistemic parameters is executed only on the weights of the pre-computed. Thus, the upper and lower bound of the expectation estimator can be computed:

$$\begin{aligned} \underline{\hat{\theta}}(\mathbf{X}_{0:k}) &= \min_{\boldsymbol{\lambda}_k} \hat{\theta}(\mathbf{X}_{0:k}; \boldsymbol{\lambda}_k) \\ \overline{\hat{\theta}}(\mathbf{X}_{0:k}) &= \max_{\boldsymbol{\lambda}_k} \hat{\theta}(\mathbf{X}_{0:k}; \boldsymbol{\lambda}_k). \end{aligned} \tag{11}$$

These optimisation problems are generally nonlinear and multimodal. Therefore, a Brunch and Bound (B&B) approach over simplicial subdomains exploiting the estimator Lipschitz continuity is developed in [11]. This algorithm ensures asymptotic convergence to the global bound and a measure of the distance from it at each iteration.

4.2. IDSS

The IDSS automatically provides support to operators in decision-making when a close encounter is detected. Using ML techniques, the IDSS uses the uncertainty encounter geometry and the reliability on the sources of information to make a decision on which is the most suitable action to be taken. The decision is based on a classification criterion of the encounter based on the probability of collision and the confidence on this value given the reliability of the information [3].

4.2.1. Confidence on the variables of interest

The IDSS provides a decision based on the confidence on the correctness of the PoC, which is quantified as the confidence of the PoC in being above or below a certain safety threshold, where PoC is the probability of collision computed without any assumption or simplification on the objects' position distribution. This confidence can be expressed as the expectation, $\mathbb{E}_p$, function of the epistemic parameters $\boldsymbol{\lambda}_k$, of the PoC being above a certain threshold, PoC_0, with uncertain space Ω_{λ_k}. Details on the computation of PoC and the limits on its expectation, $\underline{\mathbb{E}}_p$ and $\overline{\mathbb{E}}_p$, can be found in [11].

The IDSS assumes that the fast encounter hypothesis holds [22]. Under these circumstances, the distribution of the position in the impact plane can be approximated as Gaussian distributions and the computation of the probability of collision can be simplified as:

$$\text{PoC} \approx P_C = \frac{1}{2\pi\sqrt{||\boldsymbol{\Sigma}_b||}} \int_{\mathcal{B}((0,0),R)} e^{\left(-\frac{1}{2}\boldsymbol{\mu}_b^T \boldsymbol{\Sigma}_b^{-1} \boldsymbol{\mu}_b\right)} d\xi d\zeta, \tag{12}$$

where $\boldsymbol{\mu}_b(\boldsymbol{\lambda}_k)$ and $\boldsymbol{\Sigma}_b(\boldsymbol{\lambda}_k)$ are the mean and covariance matrix and are a function of the uncertain vector $\boldsymbol{\lambda}_k$, and $\mathcal{B}((0,0),R)$ is the Hard Body Radius (HBR) of the combined objects, with $\mathcal{B} : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}^2$ the transformation from the two object's state to the coordinates in the impact plane.

Thus, the uncertain space, Ω_U, can be now defined in the impact plane, and the variable of interest whose upper and lower limits with respect to the $\boldsymbol{\lambda}_k$ are to be provided by the RSE are the component of the mean and covariance of the miss distance: $\Omega_U = [\underline{\mu}_{b_\xi}, \overline{\mu}_{b_\xi}] \times [\underline{\mu}_{b_\zeta}, \overline{\mu}_{b_\zeta}] \times [\underline{\sigma}^2_{b_\xi}, \overline{\sigma}^2_{b_\xi}] \times [\underline{\sigma}^2_{b_\zeta}, \overline{\sigma}^2_{b_\zeta}] \times [\underline{\sigma}_{b_{\xi\zeta}}, \overline{\sigma}_{b_{\xi\zeta}}]$.

Given the set $\Phi = \{P_C | P_C \geq P_{C0}\}$ and the subset in the epistemic uncertainty space $\Omega = \{\mathbf{u} \in \Omega_U | P_C(\mathbf{u}) \in \Phi\}$, with $\mathbf{u} = [\mu_{b_\xi}, \mu_{b_\zeta}, \sigma^2_{b_\xi}, \sigma^2_{b_\zeta}, \sigma_{b_{\xi\zeta}}]^T$, we can compute the upper and lower probability on the values of P_C:

$$\begin{aligned} \underline{\mathbb{P}} &= \sum_{U_i \subset \Omega} m(U_i) \\ \overline{\mathbb{P}} &= \sum_{U_i \cap \Omega \neq \emptyset} m(U_i). \end{aligned} \tag{13}$$

where $\Omega_U = \bigcup_i U_i$ and m is a function that assigns a value between 0 and 1 to each subset U_i so that $\sum_{U_i \subset \Omega} m(U_i) = 1$. This is analogous to what was proposed in [3] with Dempster-Shafer theory of evidence. Note that since, by construction, $\underline{U}$ is an outer approximation of the set defined by $\mathcal{B}$, it is possible state that: $\underline{\mathbb{P}} \leq \underline{\mathbb{E}}_p \leq \overline{\mathbb{E}}_p \leq \overline{\mathbb{P}}$.

The advantage of using P_C instead of PoC in the IDSS is mainly computational: it is possible to use a more reduced uncertainty space, independently of the number of observations. In the following, we will assume the fast encounters hypothesis holds, so the IDSS will use P_C and $\mathbb{P}$. Nevertheless, an analogous system would work in a similar way using the more general variables PoC, $\boldsymbol{\lambda}_k$, and $\underline{\mathbb{E}}_p$, $\overline{\mathbb{E}}_p$, and Ω_{λ_k}, instead of P_C, $\mathbf{u}$, $\underline{\mathbb{P}}$, $\overline{\mathbb{P}}$ and Ω_U, respectively.

4.2.2. Classification criteria

The IDSS suggests the best action to an operator according to a 5 field classification criterion, which takes into account whether the confidence on the probability of collision is above a certain safety threshold (lower expectation on the P_C and the Degree of Uncertainty), and the time to the encounter (time to TCA). Table 2 summarises the criterion, where: TCA_1 and TCA_2 divides the events in short/mid/long term; P_{C_e} differentiates events according to the risk level, being $P_{Ce} \mid \mathbb{P}(P_C) < \underline{\mathbb{P}}_0 \; \forall \; P_C > P_{Ce}$, with $\mathbb{P}_0$ a minimum confidence acceptable value; and Δ is the degree of uncertainty at P_{C_0} indicating the level of uncertainty affecting the event.

The decision is encoded in these 5 classes:

- Class 1: Execute CAM. For high risk and short-term encounters or short-term uncertain events.
- Class 2: Design CAM. For high-risk events where there is time before the encounter to collect new measurements.
- Class 3: Collect more observations. Events affected by such a degree of uncertainty a decision on whether or not cannot be confidently made based on the available information, but there is time to collect more information.

Table 2: Epistemic classification criterion.

Time to TCA	P_{Ce} for $\underline{\mathbb{P}} = \underline{\mathbb{P}}_0$	Degree of uncertainty at P_{C0}	Class
$t_{TCA} < T_1$	$P_{Ce} \geq P_{C0}$	-	1
	$P_{Ce} < P_{C0}$	$\overline{\mathbb{P}} - \underline{\mathbb{P}} \leq \Delta$	5
		$\overline{\mathbb{P}} - \underline{\mathbb{P}} > \Delta$	1
$T_1 \leq t_{TCA}$ $t_{TCA} < T_2$	$P_{Ce} \geq P_{C0}$	-	2
	$P_{Ce} < P_{C0}$	$\overline{\mathbb{P}} - \underline{\mathbb{P}} \leq \Delta$	5
		$\overline{\mathbb{P}} - \underline{\mathbb{P}} > \Delta$	3
$T_2 \leq t_{TCA}$	$P_{Ce} \geq P_{C0}$	-	2
	$P_{Ce} < P_{C0}$	$\overline{\mathbb{P}} - \underline{\mathbb{P}} \leq \Delta$	4
		$\overline{\mathbb{P}} - \underline{\mathbb{P}} > \Delta$	3

- Class 4. Low risk events occurring in the mid/long-term. The available information confidently indicates the event presents low risk, but the event can be followed up if the operators considers so.
- Class 5. No need to perform a CAM. Low risk short-term events where a CAM is not required.

More details on the classification criterion can be found in [3].

4.2.3. Intelligent Classification System

The classification system would require the computation of P_C and the confidence bounds $\underline{\mathbb{P}}$ and $\overline{\mathbb{P}}$ for every new conjunction. Instead, we train an ML algorithm to classify an event simply starting from $\mathbf{u}$ and the TCA.

In this work, we use a Random Forest (RF) classifier trained on 28,800 synthetic conjunction events with different geometries and different values of $\mathbf{u}$. In previous works [3], we tested and compared different machine learning techniques (Random Forests, Artificial Neural Networks, Support Vector Machine and K-Nearest Neighbours) on a large data set of different scenarios and RF was found to be the best performing algorithm. For all the cases in this paper, we used the same settings presented in [3].

Given a conjunction event, the outcome of the IDSS is the risk class indicating the most suitable action to be taken. The output of the IDSS can then be used to schedule a new observation and update the P_C, or execute a CAM and re-assess the P_C.

The same structure would be followed if the approach using PoC, $\boldsymbol{\lambda}_k$, $\underline{\mathbb{E}}_p$ and $\overline{\mathbb{E}}_p$ were selected.

4.3. CAM

This model computes the optimal robust Collision Avoidance Manoeuvre (CAM) to avoid a potential encounter, accounting for aleatory and epistemic uncertainty on the

encounter, when the IDSS indicates an action should be taken. There are two different models: the first model is a high-efficiency version based on a linear model relating the manoeuvre with the change of position in the impact plane; the second model is a high-fidelity version, that also accounts for uncertainty on the manoeuvre itself. Both models can be combined, so the high-efficiency model quickly provides candidates to the high-fidelity model to compute a more accurate manoeuvre.

4.3.1. Linear model

This approach is based in the following linear model:

$$\delta\mathbf{x} = \mathbf{T}\delta\mathbf{v} = \mathbf{B}\mathbf{A_t}\mathbf{G}\delta\mathbf{v} \tag{14}$$

where $\delta\mathbf{x}$ is the change in position in the B plane, $\delta\mathbf{v}$ is the change of velocity due to the manoeuvre, $\mathbf{B}$ is the rotation matrix from a object's centred $< R, T, H >$ reference frame to the B plane reference frame, $\mathbf{A_t}$ is the matrix relating the change in keplerian elements at the manoeuvre positions with the change in position at the encounter, expressed in the $< R, T, H >$ reference frame, and $\mathbf{G}$ relates the change in velocity with the change in orbit parameters at the manoeuvre positions [4].

While this model presents the advantage of being faster, it assumes the fast encounter hypothesis, the rigid displacement of the uncertainty ellipse due to the manoeuvre, and Keplerian motion[4].

Under these assumptions, the optimal robust CAM can be obtained by solving the following min-max problem:

$$\begin{aligned} &\min_{\delta\mathbf{v}} \max_{\mathbf{u}\in\Omega_u} P_C \\ &s.t. \\ &\mathbf{r_e}\cdot\delta\mathbf{v} > 0 \end{aligned} \tag{15}$$

where Ω_u is the set of uncertain variables $\mathbf{u}$, and $\mathbf{r_e}$ is the objects' relative distance.

The minimisation of P_C is equivalent to finding the eigenvector associated with the maximum eigenvalue of the matrix $\mathbf{T}^T\mathbf{\Sigma}^{-1}\mathbf{T}$, where $\mathbf{\Sigma}$ is the covariance matrix in the impact plane. Thus, the minimisation step in the min-max algorithm is performed analytically. During the iteration (see Algorithm in [23] for more details), the matrix $\mathbf{T}^T\mathbf{\Sigma}^{-1}\mathbf{T}$ has been replaced by $\mathbf{T^T A T}$, where $\mathbf{A}$ is the sum of the inverse covariance matrices of the worst-case ellipses found so far: $\mathbf{A} = \mathbf{\Sigma_1^{-1}} + \mathbf{\Sigma_2^{-1}} + ...$

The maximisation step to find the worst-case ellipse given the $\delta\mathbf{v_{opt}}$ from the minimisation step has been carried out using a numerical optimiser applied directly to Eq. (12). Given the nature of the linear model and the fact that the manoeuvre does not change the shape of the uncertainty region on the B-plane but simply translates it in the same plane, the convergence is quite fast and requires only a handful of iterations. More details on the algorithm can be found in [4].

4.3.2. High-fidelity model

This model computes a more accurate CAM, including also uncertainty on the manoeuvre itself. The optimal CAM is generated by solving the following bilevel optimisation problem

$$\begin{aligned} &\min_{\Delta\mathbf{v}} \ ||\Delta\mathbf{v}|| \\ &\text{s.t. } \overline{PoC} < \varepsilon \ , \end{aligned} \tag{16}$$

Therefore, the optimal impulsive manoeuvre minimises the propellant expenditure while realising a PoC below the safe threshold ε even in the epistemic worst case scenario. The state at the encounter after the manoeuvre used to compute the PoC is propagated using the aforementioned polynomial representation of all possible propagated states.

The solution provided by the linear model approach can be used as initial guess for this model to speed up the computation.

This model accounts for error in the execution, $\delta\mathbf{V}$, quantified with the Gate's model [24]. According to this model, the realisation $\delta\mathbf{v}$ of the execution error is decomposed in modulus and angular variations on the nominal manoeuvre $\Delta\mathbf{v}$.

In a frame centred in the spacecraft with the z-axis aligned with $\Delta\mathbf{v}$, the y-axis perpendicular to both the $\Delta\mathbf{v}$ and the ecliptic normal, and the x-axis completing the right-handed frame, the execution error components are taken from zero-mean normal distributions

$$\begin{aligned} \delta v_x &\sim \mathcal{N}(0, \sigma_{pf}^2 + \sigma_{pp}^2 \|\Delta\mathbf{v}\|) \\ \delta v_y &\sim \mathcal{N}(0, \sigma_{pf}^2 + \sigma_{pp}^2 \|\Delta\mathbf{v}\|) \\ \delta v_z &\sim \mathcal{N}(0, \sigma_{mf}^2 + \sigma_{mp}^2 \|\Delta\mathbf{v}\|) \ . \end{aligned} \tag{17}$$

The parameters σ_{mf} and σ_{mp} indicate the standard deviations of the fixed and proportional magnitude components respectively, whereas σ_{pf} and σ_{pp} define the corresponding pointing components. The error in the local frame is then rotated into the inertial one where $\Delta\mathbf{v}$ is defined to achieve the error $\delta\mathbf{v}$. The executed control is the sum of the nominal manoeuvre and the error. Eq. (17) defines the aleatory uncertainty on the CAM and the execution error $\delta\mathbf{V}$ is incorporated in the uncertain model parameters $\mathbf{D}$.

5. Pipeline

In this section, the integrated pipeline of the modules explained above is presented. First, the generic workflow and the different paths the decision making process can follow is described. Second, we present an illustrative example for a (virtual) close encounter between two space objects.

5.1. System workflow

The pipeline is illustrated in Fig. 3. The workflow starts in the left bottom corner of the figure with the automatic screening of the catalogue (*Catalogue* box in Fig. 3) by the ACS (*ACS: All-vs-All* block). The catalogue includes information on the states of a number of objects at a certain epoch and information on the type of object (debris, operational satellite, manoeuvrable, etc.). Using the All-vs-All mode, it detects NE potential close encounters. After the analysis, the ACS provides an output file with a reduced set of potential conjunction pairs, indicating the objects requiring further screening. For this work, we compute the time to TCA with respect to the epoch of the catalogue for individual encounters by propagation, although this could by replaced using the One-vs-All mode as a second filter in future work.

For each of the NE conjunction pairs identified by the ACS, a more detailed conjunction analysis is automatically activated, where information about the uncertainty on the state, both epistemic and aleatory, is included in the analysis. Thus, the RSE propagates the state and uncertainty of the objects to the TCA (*RSE. Orbit & Uncertainty Propagation* box in the Figure). In case there are observations available for any

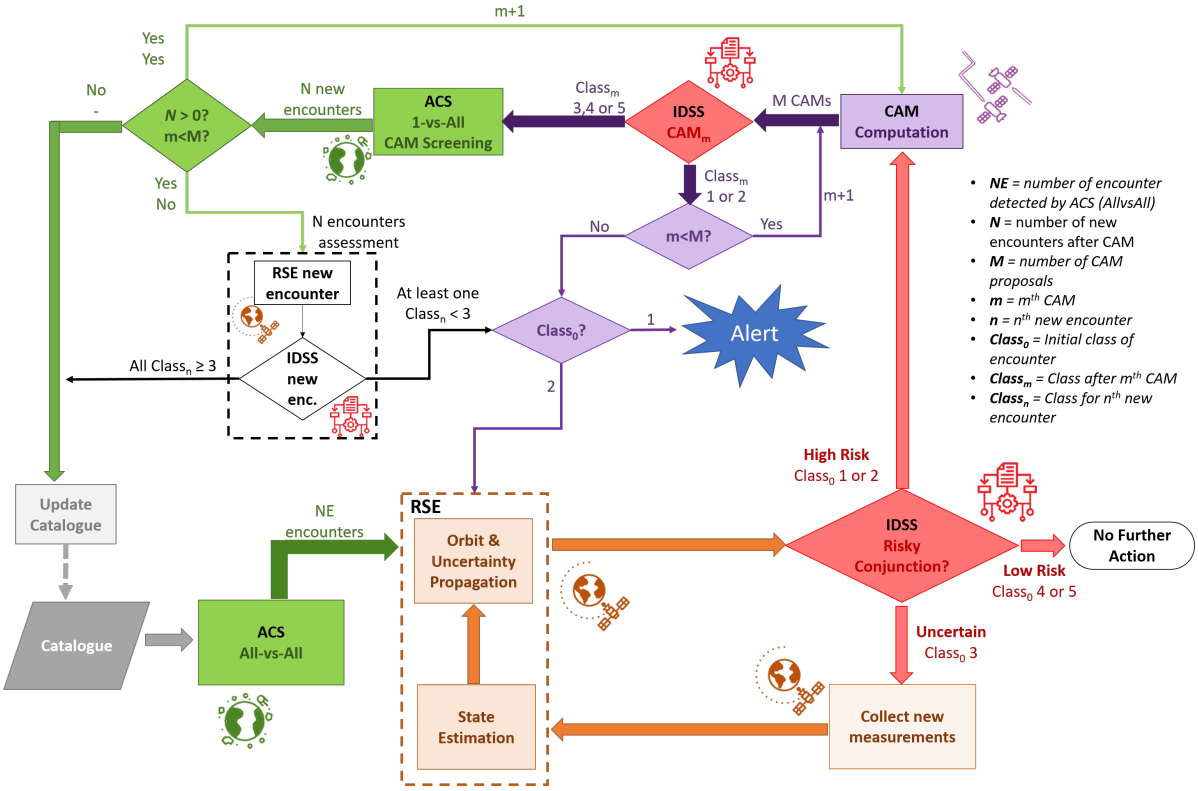

Figure 3: Pipeline of the modules to be integrated in this work.

of the objects involved, they can be included in the uncertainty propagation process (*RSE: State Estimation* box).

Once the uncertainty states of both objects are propagated to the TCA, and the uncertain geometry of the encounter is defined, the IDSS will autonomously propose the most suitable action to be made to address the event, according to the risk, the available information and the confidence on this information (*IDSS. Risky Conjunction?* decision box). This information is encoded in 5 classes (see Section 4.2). Three possible actions can be suggested:

i) *Low Risk*, related with *Class*$_0$ 4 or 5. No further action.
ii) *Uncertain*, corresponding to *Class*$_0$ 3. Allocate new observations.
iii) *High Risk*, associated with *Class*$_0$ 1 or 2. Compute CAMs.

If the event returns low risk, the situation is safe enough not to require mitigation actions (*No further action* box, right side of Fig. 3). In the case the event is classified as *Uncertain*, new simulations will be allocated (*Collect new observations* box). In this work, the new observations are simulated by the RSE (see Section 5.2). These observations may introduce new uncertainty on the system (i.e. noise). The new state and uncertainty are fed again to the RSE (represented by the *RSE* dashed box). If the event is classified as *High Risk*, it is necessary to compute at least one CAM to avoid the encounter or reduce its risk (*CAM Computation* block, right-top corner of Fig. 3). A more comprehensive explanation of these steps around the decision-making, involving the RSE, IDSS and CAM for a single encounter, can be found in [11].

In this last scenario, where the event is classified as *Class*$_0$ 1 or *Class*$_0$ 2, a new iterative process starts. The CAM module will provide M robust optimal CAM candi-

dates, sorted according to certain criteria and constraints, which can be selected by the operator [25]: time to the encounter, cost of the manoeuvres, reduction of the risk, disruption of the operational orbit, etc. As indicated in Section 4.3, to compute the CAMs, the module can opt between the fast but less accurate linear model, the high fidelity model accounting for uncertainty in the execution, or a combination of both.

The first step after obtaining the list of candidates is to evaluate the safety of the new orbit, and whether it reduces the risk satisfactorily for the considered conjunction: i.e. the IDSS should output a $Class_m$ greater than or equal to 3 for the encounter after the CAM is performed (*IDSS, CAM*$_m$ and "$m < M$?" decision boxes). In case this is not achieved by any of the proposed options ("$Class_0$*?*" decision box):

- If the original $Class_0$ before CAM was 2 (meaning a long/mid-term event), new observations are allocated, and we return to the RSE state estimation and uncertainty propagation.
- If the original class was $Class_0 = 1$ (short-term high-risk event), there is no time to allocate new measurements and an alert is raised to the operator (*Alert* box) to perform a manual detailed analysis: relax constraints on the computation of the CAM, widen the search of alternatives, etc.

For the CAM options that do reduce the risk of the event, the ephemerides associated with the new post-CAM orbit are subsequently computed, starting with the top-ranked alternative (as ranked by $Class_m$). These ephemerides are then use to screen the new orbit against all the objects in the catalogue to identify any new possible conjunctions. This operation is also performed by the ACS, but under the One-vs-All mode (*ACS: 1-vs-All CAM Screening* module). If new encounters are predicted, the ACS generates a file containing the objects involved, predicted TCAs and the predicted B-parameters for the N new encounters. This can result in three different situations ("$N > 0$?, $m < M$?" decision box):

a) No new encounters are found, in which case the execution of the manoeuvre is encouraged and the catalogue updated accordingly (*Update Catalogue* box).
b) New encounters are detected, but the analysed CAM is not the last of the proposed options. In this case, the system loops back to the CAM module and the next manoeuvre on the sorted list is selected. If the number of future encounters is lower than for previous CAMs, this alternative is saved as the most promising solution thus far.
c) Future encounters are anticipated and there are no more alternative manoeuvres remaining. This possibility is described below.

In case none of the proposed options avoid the generation of new encounters (alternative *c* above), the CAM resulting in the minimum number of new encounters is selected for further analysis.

These N new encounters are fed back to the RSE-IDSS loop (*N encounters assessment* dashed box): both orbits, the new post-CAM and that of the object in the catalogue are propagated, along with their uncertainties, to the TCA using the RSE (*RSE new encounters* module); then, the event is assessed with the IDSS (*IDSS new encounters* decision box). Depending on the $Class_n$ output by the IDSS for these N events, the following actions can be taken:

I) All events present $Class_n \geq 3$. This means there is no evidence that the new predicted events present a high risk. The best action is to perform the CAM to avoid the initial encounter and update the catalogue.
II) At least one event presents $Class_n < 3$ (high risk event) and the original event without CAM was classified as $Class_0 = 2$. Since there is enough time, new measurements are allocated, returning to the RSE state estimation and uncertainty propagation stage.
III) At least one event presents $Class_n < 3$ (high risk event) and the original event was classified as $Class_0 = 1$ (i.e. short-term high-risk event): since there is no more time to acquire measurements and all the CAM alternatives involve new encounters, an alert is raised to the operator (*Alert* box). Possible alternatives the operator can analyse are: to relax the constraints employed to obtain the list of CAM alternatives, or to consider the possibility of performing a multi-encounter CAM [25].

In summary, there are 5 possible exit points of the process, detailed below, for each of the NE potential encounters detected by the ACS module in the All-vs-All mode. Once any of these are reached, the system will restart the analysis for the next encounter.

1. The encounter does not present an actual risk ($Class_0 \geq 3$), and no further actions are required.
2. The encounter requires a CAM ($Class_0 < 3$), and the proposed CAMs do not create new encounters ($N = 0$). This implies the execution of the manoeuvre and updating of the catalogue.
3. The encounter requires a new CAM ($Class_0 < 3$), which generates new encounters ($N \neq 0$), but none of them present, at the moment of the decision, evidence of being high risk ($Class_n \geq 3 \forall n = 1, .., N$). This also implies the execution of the manoeuvre and updating of the catalogue.
4. The encounter presents a high risk, occurring in the short term ($Class_0 = 1$), and none of the proposed CAMs sufficiently reduce the risk ($Class_m \geq 3 \forall m = 1, .., M$). This scenario will raise an alert to the operators, who should decide what steps to take (e.g. obtain new CAMs, relax constraints, change criteria to compute alternatives, etc.).
5. The encounter presents a high risk, occurs in the short term ($Class_0 = 1$), and some of the new encounters generated by the new post-CAM orbit are also high risk ($\exists n \mid Class_n = 1$ or $Class_n = 2$, for $n = 1, ..., N$). In this situation, another alert will be raised, indicating that the operator should start a multi-encounter CAM analysis.

5.2. Study case

In the rest of this section, we present an example of the workflow with a synthetic catalogue containing a potential encounter.

For this study case, we use a synthetic catalogue containing 20 objects including their state vector and associated uncertainty at CNESJD 25718.999594907 (31-05-2020, 23:59:25.00). The catalogue also includes information on the type of object: operational with manoeuvre capabilities, non-manoeuvrable, debris or unknown. For the scope of this work, and due to the characteristics of the database used to train the

One-vs-All models of the ACS, a reduced catalogue including only orbits falling within the bounds indicated in Table 1 has been used.

The catalogue is automatically screened by the ACS in the *All-vs-All* mode. This mode takes the initial state vectors of each of the object pairs from the catalogue, and returns a binary label as to whether the pairs will undergo a close encounter (as defined by a 20 km screening volume) over a screening period of 7 days. Further details of this mode are given in Section 3.1.

From the screening process, 10 pairs of potential encounters are detected. Two output files are provided by the ACS. One includes the potential conjunction pairs found by the system (Table 3 displays some of these), as well as the time to the encounter and the type of object involved. The other file contains the state vector at the initial epoch used for the screening, and the associated uncertainty (if any).

Table 3: 5 of the potential encounters detected by the ACS: All-vs-All.

Obj. 1 ID	Obj. 2 ID	Time to TCA [days]	Manoeuvre Capab.	Num. encs.
11111	22222	6.956327887	OP-DEB	1
36345	41138	6.956327887	DEB-DEB	1
42127	36337	1.929933804	DEB-DEB	1
7959	10520	1.210088713	DEB-OP	1
6843	44547	4.574278349	DEB-OP	1

For the rest of this example, we will focus the study on the first pair of satellites (IDs: 11111 and 22222), whose initial conditions are included in Table 4. Note that in this work, only the secondary object of the encounter is assumed to be affected by uncertainty, and the uncertainty is given in the object's $< T, N, H >$ reference frame. The primary object is assumed to be perfectly known.

Table 4: Initial state vector and uncertainty of objects involved in the first encounter detected by the ACS. Epoch CNESJD 25718.999594907. Units: km and km/s

Obj. ID	X	Y	Z	Vx	Vy	Vz
11111	618.143	7143.340	7.183	1.119	-0.112	7.372
22222	-7183.237	-331.032	96.552	0.035	1.176	7.348
Obj. ID	σ_t	σ_n	σ_h	σ_{vt}	σ_{vn}	σ_{vh}
22222	0.104	0.556	0.139	$5.59{\cdot}10^{-6}$	$1.10{\cdot}10^{-6}$	$1.48{\cdot}10^{-6}$

Each of the encounters detected during the screening stage should undergo a detailed conjunction analysis through the RSE and IDSS modules. For this analysis, the first step is the propagation of the orbit and the uncertainty to the TCA using the RSE. However, the uncertainties in Table 4 are merely presumed [26], and therefore far from being well-characterised. More in general, the values of uncertainty associated to the other objects in the catalogue lack details on how they have been obtained. Hence, epistemic uncertainty is introduced on the covariance to account for this lack of information by parameterising $\mathbf{\Sigma}_0$ using two epistemic parameters $\boldsymbol{\lambda}_0 = [\lambda_{0-1}, \lambda_{0-2}]$.

The covariance parameterisation reads as follows

$$\boldsymbol{\Sigma}_0(\boldsymbol{\lambda}_0) = \begin{bmatrix} \lambda_{0-1}\overline{\boldsymbol{\Sigma}}_0(1:3,1:3) & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \lambda_{0-2}\overline{\boldsymbol{\Sigma}}_0(4:6,4:6) \end{bmatrix}. \tag{18}$$

The epistemic parameters scale respectively the position and velocity blocks of the reference covariance $\overline{\boldsymbol{\Sigma}}_0$ computed as above. The bounds considered for these multipliers are $\lambda_{0-1}, \lambda_{0-2} \in [1/5^2, 5^2]$, that is, they can change the reference 1σ uncertainties by roughly shrinking them up to $1/5$, or expanding them by a factor of 5.

In general, immediately after the ACS catalogue screening, no more observations are available for the objects involved in the encounter. Thus, a single prediction step is performed by the RSE. The outputs of this module are the bounds on the mean and covariance elements of the miss distance in the B-plane. This information, along with the time to the TCA, represents the inputs the IDSS uses to make the decision. The threshold values used by the IDSS are included in Table 5. More information in the RF model used in the IDSS for this example can be found in Section 4.2 and in [3].

Table 5: IDSS thresholds.

Variable	Value
TCA_1 [days]	2
TCA_2 [days]	4
P_{C_0}	10^{-6}
$\underline{\mathbb{P}}$	0.5
Δ	0.3
HBR [m]	0.3

Due to the large propagation time interval (more than 6 days) and the initial uncertainty (both, aleatory and epistemic), the miss distance intervals in the impact plane are very large (Table 6) and, as expected, the IDSS outputs a Class 3: the event is affected by such a degree of uncertainty at this step that no confident decision can be made. Thus, new measurements are allocated.

Table 6: Bounds output by the RSE if no observations available. Units in km and km^2.

Variable	Lower bound	Upper bound
μ_ξ	-1.830	-0.7759
μ_ξ	-23.939	36.722
σ^2_ξ	3.203	5.698
σ^2_ζ	$1.915{\cdot}10^3$	$5.9448{\cdot}10^4$
$\sigma_{\xi\zeta}$	-289.981	15.682

In this work, the observations are simulated within the RSE. The measurements are simulated using the debris reference trajectory and then adding the sensor errors, drawn from a zero-mean normal distribution with diagonal covariance $\boldsymbol{\Sigma}_y$, whose non-zero elements are: $1\sigma_{y_{az}} = 10$ and $\sigma_{y_{el}} = 10$ arcsec. Similarly to the uncertainty in the initial position, epistemic uncertainty on the observations is added to quantified the

lack of detail on the observation sensors. The epistemic uncertainty is considered on the noise covariance by means of two parameters $\boldsymbol{\lambda}_y = \left[\lambda_{y-az}, \lambda_{y-el}\right]$, which range in the interval $\lambda_{y-az}, \lambda_{y-el} \in [1/5^2, 5^2]$, in line with diverse 1σ values found in literature [27, 28].

As mentioned above, the primary satellite is assumed to be perfectly known, so the observations refer only to the secondary object. 10 observations are simulated, evenly distributed between the initial epoch and one day before the TCA. Again, the initial state and uncertainty of the objects are propagated to the TCA, although in this case, the position is updated sequentially with the information provided by the measurements. The robust bounds of the new B-plane variables are included in Table 7.

Table 7: Bounds output by the RSE when 10 measurements. The time to TCA after the last observation is 1 day. Units in km and km^2.

Variable	Lower bound	Upper bound
μ_ξ	-2.0408$\cdot 10^{-2}$	-1.1596$\cdot 10^{-2}$2
μ_ξ	0.6783	1.20140
σ^2_ξ	2.0571$\cdot 10^{-11}$	2.1102$\cdot 10^{-4}$
σ^2_ζ	9.7038$\cdot 10^{-4}$	5.81921$\cdot 10^{-2}$
$\sigma_{\xi\zeta}$	-2.0504$\cdot 10^{-4}$	1.2473$\cdot 10^{-3}$

These bounds along with the new time to the encounter (1 day) are inserted in the IDSS, which classifies the event, accounting for the new information, as Class 1. This means that a CAM should be performed.

This decision activates the CAM module, which computes a list of optimal manoeuvres. In this work, we have only employed the linear model-based CAM (Section 4.3.1). We have computed 7 possible CAMs, executed half an orbit before the encounter position, for the 9 revolutions before the encounter, but the last two revolutions, $\theta_m \in \Theta_m \,|\, \Theta_m = \{k\pi\}, k = 5, 7, 9, ..., 17$. The maximum capacity of the thruster has been set to 10 cm/s. The only criterion considered to sort the proposed CAM options was the disruption of the orbit (i.e. time away of the nominal orbit before the encounter), thus the later manoeuvres are ranked first. For more information on other criteria, see [25]. Table 8 includes the list of manoeuvres, expressed in the manoeuvrable satellite's $< T, N, H >$ reference frame.

Table 8: List of ranked possible manoeuvres, expressed in the satellite's $< T, N, H >$ reference frame. The last column includes the $Class_m$ of the event after the manoeuvre. Units in rad. and m/s.

Alternative	θ_m	δv_t	δv_n	δv_n	$Class_m$
CAM 1	5π	9.9994$\cdot 10^{-2}$	-1.0527$\cdot 10^{-4}$	0	5
CAM 2	7π	9.9997$\cdot 10^{-2}$	-7.4706$\cdot 10^{-5}$	0	5
CAM 3	9π	9.9998$\cdot 10^{-2}$	-5.7745$\cdot 10^{-5}$	0	5
CAM 4	11π	9.9998$\cdot 10^{-2}$	-4.6958$\cdot 10^{-5}$	0	5
CAM 5	13π	9.9999$\cdot 10^{-2}$	-3.9492$\cdot 10^{-5}$	0	5
CAM 6	15π	9.9999$\cdot 10^{-2}$	-3.4018$\cdot 10^{-5}$	0	5
CAM 7	17π	9.9999$\cdot 10^{-2}$	-2.9830$\cdot 10^{-5}$	0	5

The different CAM alternatives are evaluated in order of priority. First, the risk associated to the encounter after implementing the manoeuvre is computed. The class of the encounter after the first manoeuvre is 5 (Table 8), meaning the new orbit is safe. Then, the new ephemerides are computed and a further analysis performed using the *One-vs-All* mode of the ACS to detect possible future conjunctions with other catalogued objects. Note, that in case that the first alternative would not have reduced the risk of the event to Class 4 or 5, the next alternative would have been chosen.

The model use to predict future encounters in the *One-vs-All* ACS mode is described in more detail in Section 3.2. This mode takes the newly computed ephemerids of the manoeuvred object, and the ephemerids of the other catalogued objects over a 1-day period after the CAM as input, and predicts the evolution of the B-parameter for each possible conjunction pair over a 3-day screening period. The threshold on the B-parameter chosen to identify encounter events was set to 20 km.

The first CAM implemented already reduces the risk of the current encounter and, according to the *One-vs-All*, does not generate further encounters. This means that the system proposes the manoeuvre to the operators. In case the operators decides to execute it, the catalogue has to be updated accordingly.

Once this manoeuvre has been analysed, the system automatically start the analysis of the next pairs found by the *All-vs-All* module.

6. Conclusion

In this paper we have presented new methods and functionalities of CASSANDRA, a tool for supporting space operators in SEM problems.

Firstly, we have extended the capabilities of the ACS module, responsible for the automatic screening of the catalogue to detect potential conjunctions. In addition to the *All-vs-All* approach, we have included a higher-fidelity *One-vs-All* mode. Thus, the ACS can be used in several kinds of scenarios, being activated periodically to screen the whole catalogue (*All-vs-All*), or on demand to analyse specific scenarios (*One-vs-All*).

Secondly, we have successfully established a framework where the ACS, RSE, IDSS and CAM modules of CASSANDRA can work together. This framework enables the scope of each individual module to be extended to address more complex problems, exemplified by the decision-making process that follows the automatic screening of a space object catalogue. The proposed framework allows us to include uncertainty at each single step, both epistemic and aleatory, enabling the system to propose robust decisions to the operators.

Thirdly, we have illustrated the operation of this framework with a representative study case, with which we have highlighted the main characteristics of the pipeline. This example shows the automatic nature of the whole process for analysing a set of potential encounters within a catalogue, and highlights the relevance of the robust decision-making system to handle uncertain or incomplete information.

Finally, we have identified some promising avenues for future work. Multi-criteria decision making algorithms are expected to be implemented throughout the pipeline to classify the potential encounters identified by the *All-vs-All* mode of the ACS module, or to sort the CAM alternatives, to avoid high-risk encounters based on more complex criteria. Similarly, the CAM module will be improved to combine the fast linear model with the high fidelity method. Lastly, we expect to extend the pipeline to integrate the multi-encounter CAM method for those scenarios that require it.

Acknowledgments

This work was supported by the European Space Agency, through the Open Space Innovation Platform (OSIP), Idea I-2019-01650: “Artificial Intelligence for Space Traffic Management”, and partially by the European Commission’s Stardust-R programme, through the by the EU H2020 MSCA ITN Stardust-R, grant agreement 813644.

References

[1] Sanchez, L., Vasile, M. and Minisci, E., ”AI and Space Safety: Collision Risk Assessment”, in: Handbook of Space Security: Policies, Applications and Programs, Springer International Publishing, Cham, 2020, pp. 1–19.

[2] Stevenson, E. and Rodriguez-Fernandez, V. and Urrutxa H. and Morand, V. and D. Camacho, ”Artificial Intelligence for All vs All Conjunction Screening”, in: Proceedings of the 8th European Conference on Space Debris, ESA/ESOC. 20-23 April 2021.

[3] Sanchez, L. and Vasile, M, ”On the Use of Machine Learning and Evidence Theory to Improve Collision Risk Management”, Acta Astronautica 181 (2021) 694–706.

[4] Sanchez, L. and Vasile, M., ”AI for Autonomous CAM Execution”, in: Proceedings 71st International Aeronautical Congress (IAC), The Cyber Space Edition. 12-14 October 2020.

[5] Vasile, M., Rodriguez-Fernandez, V., Serra, R., Camacho, D. and Riccardi, A., Artificial Intelligence in Support to Space Traffic Management, in: Proceedings 68th International Astronautical Congress (IAC), Adelaide, Australia. 25-29 September 2017.

[6] Flohrer, T., Krag, H., Merz, K. and Lemmens, S., ”CREAM - ESA’s Proposal for Collision Risk Estimation and Automated Mitigation”, in: Proceedings of the Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS). 2019.

[7] Stevenson, E. and Rodriguez-Fernandez, V. and Minisci, E. and Camacho, D., ”A deep learning approach to solar radio flux forecasting”, Acta Astronautica 193 (2022) 595–606.

[8] Manzi, M. and Vasile, M., ”Autoencoder-based Thermospheric Density Model for Uncertainty Quantification and Real-Time Calibration”, in: Proceedings of the 8th European Conference on Space Debris, ESA/ESOC. 20-23 April 2021.

[9] Manzi, M. and Vasile, M., ”Discovering Unmodeled Components in Astrodynamics with Symbolic Regression”, in: Proceedings 2020 IEEE Congress on Evolutionary Computation (CEC). 19-24 July 2020.

[10] Greco, C. and Vasile, M, ”Robust Bayesian Particle Filter for Space Object Tracking Under Severe Uncertainty”, Journal of Guidance, Control, and Dynamics (2021) 1–18.

[11] Greco, C. and Sanchez, L. and Vasile, M., ”A Robust Bayesian Agent for Optimal Collision Avoidance Manoeuvre Planning”, in: Proceedings 8th European Conference in Space Debris, ESA/ESOC. 20-23 April 2021.

[12] Falchi, A. and Renato, V. and Minisci, M. and Vasile, M., ”FOSTRAD: An advanced open source tool for re-entry analysis”, in: Proceedings 15th Reinventing Space Conference, Glasgow, United Kingdom. 24-26 October 2017.

[13] Morand, V., Yanez, C., Dolado Perez, J. C., and Fernandez, C. and Roussel, S. and Pucel, X. and Vidal, V., ”BAS3E: A framework to Conceive, Design, and Validate Present and Future SST Architectures”, in: Proceedings of the 1st NASA International Orbital Debris Conference, p. 10.

[14] European Space Agency, ”Database and information system characterising objects in space”, 2022. `https://discosweb.esoc.esa.int/`.

[15] Bonnal, C., McKnight, D., Phipps, C., Dupont, C., Missonnier, S., Lequette, L., Merle, M. and Rommelaere, S., ”Just in time collision avoidance – A review”, Acta Astronautica 170 (2020) 637–651.

[16] Sanchez, L., Vasile, M. and Minisci, E., ”AI to Support Decision Making in Collision Risk Assessment”, in: Proceedings 70th International Aeronautical Congress (IAC), Washington D.C., USA. 21-25 October 2019.

[17] Jah, M., ”Derivation of the B-plane (Body Plane) and its Associated Parameters”, in: A Lecture Series on Orbit Mechanics and Interplanetary Mission Design, University of Colorado. 2002.

[18] Alfano, S., ”Review of conjunction probability methods for short-term encounters”, Advances in the Astronautical Sciences 127 (2007) 719–746.

[19] Fawaz, H. I., Lucas, B., Forestier, G., Pelletier, C., Schmidt, D. F., Weber, J., Webb, G. I., Idoumghar, L., Muller, P. and Petitjean, F., "InceptionTime: Finding AlexNet for time series classification", Data Mining and Knowledge Discovery 34 (2020) 1936–1962.
[20] Oguiza, I., "tsai - A state-of-the-art deep learning library for time series and sequential data", Github, 2022. https://github.com/timeseriesAI/tsai.
[21] Greco, C. and Campagnola, S. and Vasile, M, "Robust Space Trajectory Design using Belief Optimal Control", Journal of Guidance, Control, and Dynamics (Under review).
[22] Serra, R. and Arzelier, D. and Joldes, M and Lasserre, J. B. and Rondepierre, A. and Salvy, B., "Fast and Accurate Computation of Orbital Collision Probability for Short-Term Encounters", Journal of Guidance, Control, and Dynamics 39 (2016) 1–13.
[23] Vasile M., "On the Solution of Min-Max Problems in Robust Optimization", in: Proceedings of the EVOLVE of Guidance, Control and Dynamics International Conference, Beijing, China. 2014.
[24] Gates, C.R., "A Simplified Model of Midcourse Maneuver Execution Errors", JPL technical report, Jet Propulsion Laboratory, California Institute of Technology, 1963.
[25] Sanchez, L. and Vasile, M., "Constrained Optimal Collision Avoidance Manoeuvre Allocation Under Uncertainty for Subsequent Conjunction Events ", in: Proceedings 72nd International Aeronautical Congress (IAC), Dubai, UAE. 25-29 October 2021.
[26] Klinkrad, H. and Alarcon, J. R. and Sanchez, N., "Collision avoidance for operational ESA satellites", in: Proceedings of the 4th European Conference on Space Debris, ESA/ESOC, Darmstad, Germany. 18-20 April 2005.
[27] Bennett, J. C. and Sang, J. and Smith, C. and Zhang, K., "An analysis of very short-arc orbit determination for low-Earth objects using sparse optical and laser tracking data", Advances in Space Research 55(2) (2015) 617–629.
[28] Li, B. and Sang, J. and Zhang, Z., "A real-time orbit determination method for smooth transition from optical tracking to laser ranging of debris", Sensors 16(7) (2016).