IAC–17–A6,7,1,x41479

# ARTIFICIAL INTELLIGENCE IN SUPPORT TO SPACE TRAFFIC MANAGEMENT

**Massimiliano Vasile**
University of Strathclyde, United Kingdom, massimiliano.vasile@strath.ac.uk

**Víctor Rodríguez-Fernández**
Universidad Autónoma de Madrid, Spain, victor.rodriguez@inv.uam.es

**Romain Serra**
University of Strathclyde, United Kingdom, romain.serra@strath.ac.uk

**David Camacho**
Universidad Autónoma de Madrid, Spain, david.camacho@uam.es

**Annalisa Riccardi**
University of Strathclyde, United Kingdom, annalisa.riccardi@strath.ac.uk

This paper presents an Artificial Intelligence-based decision support system to assist ground operators to plan and implement collision avoidance manoeuvres. When a new conjunction is expected, the system provides the operator with an optimal manoeuvre and an analysis of the possible outcomes. Machine learning techniques are combined with uncertainty quantification and orbital mechanics calculations to support an optimal and reliable management of space traffic. A dataset of collision avoidance manoeuvres has been created by simulating a range of scenarios in which optimal manoeuvres (in the sense of optimal control) are applied to reduce the collision probability between pairs of objects. The consequences of the execution of a manoeuvre are evaluated to assess its benefits against its cost. Consequences are quantified in terms of the need for additional manoeuvres to avoid subsequent collisions. By using this dataset, we train predictive models that forecast the risk of avoiding new collisions, and use them to recommend alternative manoeuvres that may be globally better for the space environment.

## I. INTRODUCTION

The IAA Cosmic Study on Space Traffic Management (2006) defined STM as "the set of technical and regulatory provisions for promoting safe access into outer space, operations in outer space and return from outer space to Earth free from physical or radio-frequency interference". One of the key operations is collision avoidance that follows the prediction and detection of a possible collision. With the increase of space traffic and number of objects to be operated, it is expected that ground operators might struggle and make sub-optimal decisions due to the less predictable consequences of each manoeuvre. More in general the increased traffic and associated long term evolution of the space environment can benefit from a number of existing techniques in support to operations that have been developed for air space traffic management. The motivation of this work is to explore the use of machine learning, and more in general artificial intelligence, to support ground operators when planning and implementing collision avoidance manoeuvres. By using a synthetic, and controlled, population, this work aims at showing how historical data, combined with, potentially expensive, simulations, can help supporting the decisions of ground operators.

The main contributions of this paper can be summarized as follows:

- The development of methodologies for generating databases of collision avoidance manoeuvres, from the perspectives of a real and a virtual scenario.
- The use of Artificial Intelligence techniques for exploiting the database and provide the operators with additional and valuable information when planning a manoeuvre.

The rest of the paper is structured as follows: The next section introduces the collision avoidance problem, and describes how the computation of the collision probability and the manoeuvres have been designed. Then, Section 3 proposes a methodology for generating a dataset of collision avoidance manoeuvres, for both a real and a virtual scenario. Using this methodology, a dataset of manoeuvres is created, and in Section 4 some possible uses

based on Artificial Intelligence are presented, including the prediction of the future consequences of a manoeuvre, and the recommendation of safer manoeuvres. Finally, Section 5 summarizes the results found in this work and presents some possible future research lines.

## II. Collision avoidance problem

The problem of collision avoidance arises each time a satellite operator is delivered a collision alert. Typical warnings such as the so-called Conjunction Data Messages (CDM) provided by the Joint Space Operations Center (JSpOC) contain information on a pair of objects susceptible to cross path in the near future, including their approximated size and a predicted time of encounter. Due to uncertainty in both orbit measurement and propagation in time, positions and velocities are stochastically described by their mean and covariance matrix. While it is possible that the active satellite, referred to as the primary $p$, is better known by the operator itself that can then correct the data, the orbital debris, referred to as the secondary $s$, is generally not. Nonetheless, updates on the alert can refine the knowledge on the states of both objects, calling for new assessments. In order to account for the random nature of the information contained in the conjunction message, the collision risk needs to be evaluated as a probability. When compared to a safety threshold e.g. one out of a thousand, it allows to judge if a thrusting strategy is needed or not for the primary to avoid the threat. Once this decision has been made, one or several orbital manoeuvres are designed. In the context of high-thrust propulsion, where a $\Delta$V can be considered as instantaneous, each of them is defined by its direction, magnitude and time of application. If performed sufficiently in advance, a single impulse, often small with respect to the original velocity, can be enough to mitigate the risk. It is assumed to be the case in this paper, lowering down the number of optimization parameters to four: date and Cartesian coordinates of the manoeuvre. In order to reduce the impact on mission lifetime, the cost function shall reflect fuel-consumption and simply writes as the Euclidean norm of the $\Delta$V. This section details the methods presently used to evaluate the collision probability and to design an avoidance manoeuvre.

### II.i Encounter model

While, in general, computing the collision probability is a complex problem, a simplified model exists for conjunctions with large relative speed i.e. superior to 1km/s. This is usually what happens in Low Earth Orbit (LEO), where orbital velocities are naturally high, and is assumed to be the case here. This type of encounter is referred to in the literature[Alf07, Cha08] as short-term and is briefly described thereafter.

Prerequisites for the computation of the collision probability are the knowledge, at a given time $t_C$ where the objects are predicted to be close by, of the state (position and velocity) of the primary and secondary, each one of them described by a multinormal distribution. It is worth noticing that such a law, for $p$, as given in a first alert, is based on the natural evolution of its original orbit as acquired from measurements. In order to know how a manoeuvre anterior to $t_C$ would affect it, one needs to propagate backward in time the trajectory, apply the impulse and propagate forward. In the end, it is still possible to sample the state of the two objects at $t_C$, with or without a manoeuvre taken into account. Note that if $t_C$ is actually the time of closest approach between the two mean trajectories when there is no mitigation measure, it is not any more when there is one, but the methodology described below to compute the collision probability remains valid as long as the $\Delta$V is relatively small.

The key assumption of short-term encounters, based on the high speeds involved, is that each object moves on a straight line. As a result, the relative trajectory is rectilinear as well. In order to conservatively account for all possible orientations of the objects, thus discarding attitude, one can model each object with a sphere containing it, with respective radius $R_p$ and $R_s$. Then, for a given value of the relative velocity at $t_C$, the set of relative positions at this date leading to a collision at any time (i.e. smaller, equal or greater than $t_C$) is an infinite cylinder. Its direction is aligned with the relative velocity and its radius equals $R = R_p + R_s$. This so-called tube of collision represents a simple geometric criterion, in the form of a single scalar inequality, to count colliding cases within a given set of samples. More precisely, let $(\mathbf{r}_p, \mathbf{v}_p)$ and $(\mathbf{r}_s, \mathbf{v}_s)$ be occurrences of the random vectors describing position and velocity of the two objects at $t_C$, then the event 'collision' writes:

$$\|\mathbf{r}_p - \mathbf{r}_s\|^2 - \left( (\mathbf{r}_p - \mathbf{r}_s)^T \cdot \frac{\mathbf{v}_p - \mathbf{v}_s}{\|\mathbf{v}_p - \mathbf{v}_s\|} \right)^2 < R^2 \quad [1]$$

Note that, strictly speaking, the original short-term model also assumes that uncertainty on velocity is negligible and that the latter can be considered as deterministic. This way, the number of uncertain variables is significantly lowered and the Gaussian assumption allows for a power series expansion of the collision probability.[SAJ+16] However, in theory, the addition of an avoidance manoeuvre degrades the properties of the distribution for the primary, hence the choice here of a Monte Carlo approach that does not require this additional assumption.

II.ii Maneuver design

In this paper, the effects of a $\Delta$V on the trajectory of the primary are computed neglecting orbital perturbations i.e. assuming Keplerian dynamics, which is non-linear but can be propagated analytically and rather fast. Due to the probabilistic formulation of the problem, the minimization of fuel-consumption for collision avoidance belongs to the class of chance-constrained optimization,[Pre70] that are hard to solve in general. However, heuristics have been proposed by operators[KAS05] as a work-around. One possibility for that is to arbitrarily lower the number of decision variables. For instance, on a circular orbit, for a given velocity jump, the manoeuvre maximizing the separation with respect to the original trajectory is a tangential impulse, performed half an orbit before the point of interest. Building on this result, one can set in advance both the date and direction of the $\Delta$V, based on the mean state of the primary. A simple one-dimensional search is then enough to find the minimum magnitude satisfying the probabilistic constraint on collision. However, this particular method fails for non-negligible eccentricities, since the theoretical result on separation becomes invalid in general. As a more universal approach, it is proposed here, for any given time of application, to precompute the direction of the impulse in the way it was suggested for asteroid deflection.[VC08] It is worth noting that the only simplification with this technique is that the effects of the $\Delta$V are linearized, making the obtained direction pseudo-optimal for separation. Nevertheless, the magnitude is then computed with a bisection method using the full, nonlinear model. This procedure, described in more details below, is repeated for a list of dates within a given interval, and the less expansive one is selected.

Supposing that the burn occurs at a date $t_B$, the approach consists in two steps: first, getting a pseudo-optimal direction of thrust; and second, iterating on the Euclidean norm of the impulse to converge towards the minimum value such that the collision risk is mitigated according to the chosen threshold. The latter is achieved by dichotomy on the magnitude of the $\Delta$V, with the stochastic constraint computed via a Monte Carlo approach through Eq.[1]. On the other hand, finding a priori an optimal thrust direction requires some analysis of the orbital mechanics. In order to evaluate the effects of the past manoeuvre on the mean state of the primary at the expected conjunction, one needs to estimate these effects at the time of application $t_B$, and then propagated them to $t_C$.

$$\mathbf{G} = \begin{bmatrix} \frac{2a^2 v_B}{\mu} & 0 & 0 \\ \frac{2(e+\cos\theta_B)}{v_B} & -\frac{r_B}{a v_B}\sin\theta_B & 0 \\ 0 & 0 & \frac{r_B\cos(\omega+\theta_B)}{h} \\ 0 & 0 & \frac{r_B\sin(\omega+\theta_B)}{h\sin i} \\ \frac{2\sin\theta_B}{e v_B} & \frac{2e+(r_B/a)\cos\theta_B}{e v_B} & -\frac{r_B\sin(\omega+\theta_B)\cos i}{h\sin i} \\ -\frac{b}{e a v_B}\left(1+\frac{e^2 r_B}{p}\right)2\sin\theta_B & -\frac{b}{e a v_B}\frac{r_B}{a}\cos\theta_B & 0 \end{bmatrix} \quad [2]$$

$$\mathbf{A}^T = \begin{bmatrix} \frac{r_C}{a} - \frac{3}{2}\frac{e\sin\theta_C}{\sqrt{1-e^2}}\frac{\sqrt{\mu}}{a^{3/2}}\Delta t & -\frac{3}{2}\frac{r_C}{(1-e^2)^{3/2}}(1+e\cos\theta_C)^2\frac{\sqrt{\mu}}{a^{5/2}}\Delta t & 0 \\ -a\cos\theta_C & \frac{r_C\sin\theta_C}{1-e^2}(2+e\cos\theta_C) & 0 \\ 0 & 0 & r_C\sin(\omega+\theta_C) \\ 0 & r_C\cos i & -r_C\cos(\omega+\theta_C)\sin i \\ 0 & r_C & 0 \\ \frac{a e\sin\theta_C}{\sqrt{1-e^2}} & \frac{r_C}{(1-e^2)^{3/2}}(1+e\cos\theta_C)^2 & 0 \end{bmatrix} \quad [3]$$

At first order, the instantaneous impact of the velocity jump on $p$ can be obtained from the Gauss planetary equations.[Bat99] Let $\delta\mathbf{v}$ be the Cartesian coordinates of the impulse in the so-called $\{\hat{\mathrm{t}}, \hat{\mathrm{n}}, \hat{\mathrm{h}}\}$ frame associated to the mean state of the primary. Let $\delta\mathbf{x}$ be the vector of induced differences in orbital elements w.r.t. the original set $(a, e, i, \Omega, \omega, M_B)$, where the subscript $B$ refers to $t_B$ and the mean anomaly $M$ replaces the true anomaly $\theta$ among

the Keplerian coordinates. Then, one has $\delta\mathbf{x} = \mathbf{G}\delta\mathbf{v}$ with the matrix $\mathbf{G}$ given in Eq.[2], where $\mu$ is the Earth gravitational constant, $p$ the latus rectum and $h$ the norm of the momentum.

Without orbital perturbations, the time evolution of this initial deviation is analytically known. Using local coordinates with the unaltered trajectory as a reference, it can be expressed linearly with a matrix $\mathbf{A}$ derived from the so-called proximal motion,[SJ03] the advantage of this set of equations being that it works also for highly eccentric cases. Note that $\mathbf{A}$, given in Eq.[3], includes some terms proportional to the elapsed time $\Delta t = t_C - t_B$ due to the drift in mean anomaly induced by the difference in semi-major axis. The subscript $C$ refers to values at date $t_C$.

In the end, the relative position at time $t_C$ of the manoeuvred object w.r.t. its unaltered trajectory writes $\delta\mathbf{r} = \mathbf{T}\delta\mathbf{v}$, where $\mathbf{T} = \mathbf{AG}$. Thus maximizing the separation $\|\delta\mathbf{r}\|$ is equivalent to maximizing the quadratic form $\mathbf{T}^T\mathbf{T}$, which is achieved, at constant input magnitude, by choosing a direction parallel to the eigenvector associated to its largest eigenvalue. Therefore, computing this particular vector gives the orientation of the thruster necessary to initiate the dichotomic search on the cost. Recall that the latter is performed with the nonlinear dynamics, so that no simplifying assumption is made at this stage. More precisely, in order to know the effects of the manoeuvre with an arbitrary magnitude, the trajectory of the primary is propagated backward from $t_C$ to $t_B$ using the Lagrange $F$ and $G$ functions.[Bat99] Then the $\Delta$V, parallel to the precomputed direction, is added to its velocity and finally, the state is propagated forward using the same technique, to $t_C$ again so that the collision probability can be computed as explained in 2.1.

## III. Creating a dataset of manoeuvres

Since the system to be created must help the operator by providing him with information extracted from previously applied manoeuvres, one of the keys of this work lies in the quantity and quality of the past-manoeuvre data we have available. To the best of our knowledge, there is no public databases with this type of data. Thus, a methodology has been developed to generate this database, for both a real and a virtual scenario.

### III.i Real scenario

Based on real tracking data of space objects, the goal here is to look for high probability collisions between operational satellites and orbital debris, and to generate avoidance manoeuvres to avoid those collisions. Furthermore, we will analyze the subsequent high probability collisions that may arise after applying the manoeuvre, and thus, the possible subsequent manoeuvres needed, which are computed recursively. Figure 1 shows a graphical outline of this process, whose steps are detailed below:

A. *Read TLE database*: The input data consists in a set of JSON * files containing a list of measurements of space objects at a specific epoch. The orbital elements are encoded using the Two-line element set (TLE) data format, and each file contains measurements for about one day. Based on the name of each tracked object (contained in the title line of the TLE), we can separate primary and secondary objects (or debris), and thus, cross them to create the set of all possible pairs (primary,secondary). Assuming that $[t_0, t_f]$ is the interval of interest for analyzing these data, we only take into account the first measurement found for each object along this interval.

B. *Compute minimum critical points*: For each pair of objects, the critical points of the squared Keplerian distance between their orbits are computed, according to an algebraic method introduced in.[Gro05] Only the minimum critical points whose miss distance is under a threshold $D$ (set to 10 meters in this work) are accepted.

C. *Compute times at critical point in* $[t_0, t_f]$: For each accepted critical point, we compute the times when each object in the pair arrives to that true anomaly in the interval of interest $[t_0, t_f]$. Those times ($\{t_i^p\}$ for the primary object and $\left\{t_j^s\right\}$ for the secondary) are then compared, and any situation such that the difference between $t_i^p$ and $t_j^s$ is closer than a threshold $T$ (set to 1s in this work) is considered as a potential high probability collision.

D. *Compute collision probability*: The time of possible conjunction, $t_C$, is set to the half point between the times when primary and secondary arrive to the critical point, i.e., $t_C = \left(t_i^p + t_j^s\right)/2$. As it was described in Section 2.1, here we assume the case of short-terms encounters, and compute the collision probability by counting the occurrences of a colliding event (See Eq. 1) within a set of one million samples of the state of primary and secondary at $t_C$. To perform the sampling (a 6-dimensional multinormal distribution), the mean state is set to the result of propagating each object to $t_C$ from the TLE. As for the covariance matrix, it is fixed as a diagonal matrix expressed in the $\{\hat{\mathrm{t}}, \hat{\mathrm{n}}, \hat{\mathrm{h}}\}$ reference frame, with ratios for standard deviations used by Chan:[Cha08] cross-track and in-track errors respectively 2 and 10 times larger than radial ones. Following this rule, the first three elements of the diagonal, associated to the position

* JSON: JavaScript Object Notation: `http://www.json.org/`

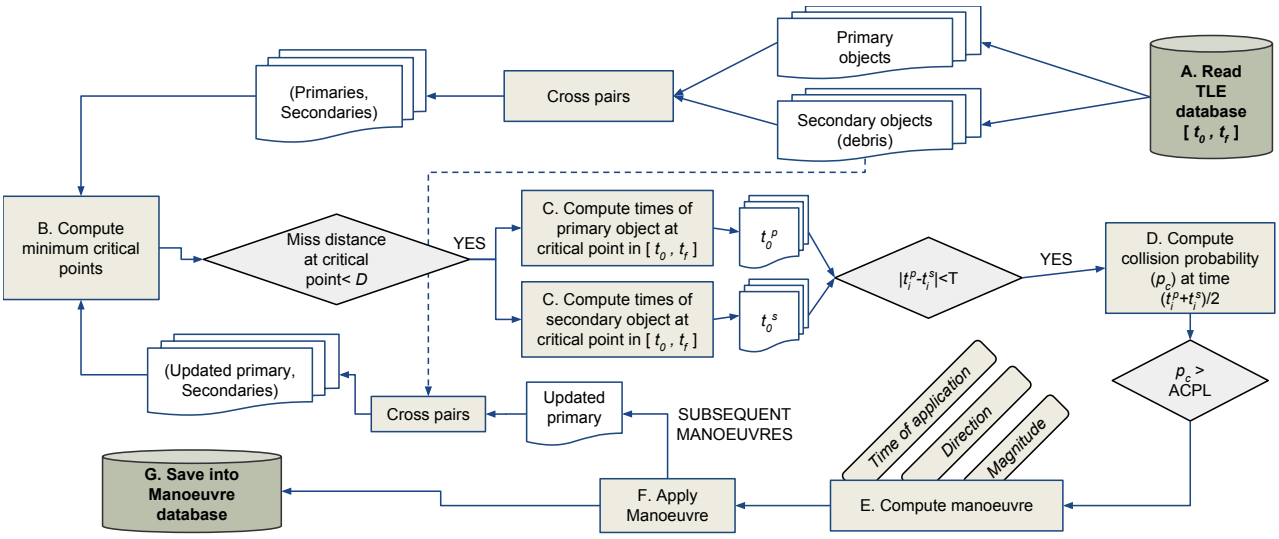


Fig. 1: Process flow for generating real scenario-based collision avoidance manoeuvres.

of the primary object, were set to $\left[10^4, 10^2, 4\times10^2\right]$ $\mathrm{m}^2$, and for the last three elements, related to the velocity, $\left[10^{-2}, 10^{-4}, 4\times10^{-4}\right]$ $\mathrm{m}^2\,\mathrm{s}^{-2}$ i.e. negligible values. On the other hand, diagonal values for secondary objects are fixed to $\left[10^6, 10^4, 4\times10^4\right]$ $\mathrm{m}^2$ and $\left[10^{-2}, 10^{-4}, 4\times10^{-4}\right]$ $\mathrm{m}^2\,\mathrm{s}^{-2}$ respectively. This difference between the two objects comes from the fact that it is assumed that the primary, as an operational satellite, is usually larger and as such easier to track than a debris. Since a TLE does not give information about the size of an object, it is fixed here by setting a radius of 9 meters for primaries and 1 meters for secondaries, always summing up to a combined radius of $10$ m. The Accepted Collision Probability Level (ACPL) is chosen as $0.0001$. If the calculated collision probability is above this threshold, the conjunction is considered as high-risk and requires an avoidance manoeuvre.

E. *Compute collision avoidance manoeuvre*: The direction and magnitude of the collision avoidance manoeuvre are computed following the procedure presented in Section 2.2. The dichotomy on the magnitude of the $\Delta$V is applied separately on the intervals $[-0.01, 0]\,\mathrm{m\,s}^{-1}$ and $[0, 0.01]\,\mathrm{m\,s}^{-1}$, keeping the lowest magnitude found between the two processes. Furthermore, these intervals are extended automatically in case that the boundary magnitudes are not enough to reduce the collision probability below the ACPL. Regarding the time of application, we simply try different possibilities from three periods (of the primary object) before $t_C$ to two periods before $t_C$, and select the one that minimizes the magnitude of the $\Delta$V found.

F. *Apply manoeuvre*: The computed $\Delta$V is applied to the primary object in order to check the impact of the mitigation in terms of future encounters and subsequent manoeuvres needed. What we do is to update the state of the primary object by applying the $\Delta$V, and cross it to the set of secondaries to study again the possibility of collision with each of them, which consists in restarting the process described here from point B. The interval of analysis $[t_0, t_f]$ is also relocated to a future time after the avoided collision. This process of computing subsequent manoeuvres can be performed recursively. In addition, we will also keep a copy of the primary object without the effect of the manoeuvre, and analyse its future subsequent collisions in the same way as described above. This allows to compare the future consequences of a manoeuvre in two different future situations.

G. *Save into manoeuvre database*: The information about the collision, the manoeuvre and the subsequent ones are stored into a MongoDB [†] database. A complete description of a recorded manoeuvre is given in Section 3.3.

### III.ii Virtual scenario

The real scenario described in the previous section has the disadvantage of being computationally intensive, due

[†] MongoDB: `https://www.mongodb.com/`

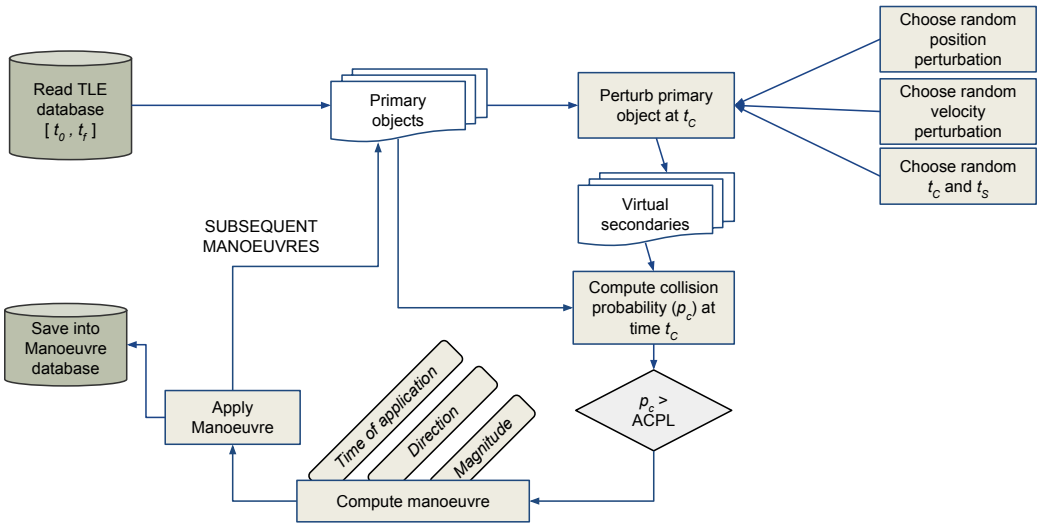


Fig. 2: Process flow for generating virtual scenario-based collision avoidance manoeuvres.

to it needs to move along every pair of tracked objects recursively. Additionally, the probability of finding high probability collisions in this scenario is low, and thus one can analyse thousands of pairs and find only a few situations where a manoeuvre is needed. For that reason, a simplified scenario for generating high probability collisions and avoidance manoeuvres is proposed here, simulating a plausible future scenario where lot of collisions will happen. A graphical outline of this process is shown in Figure 2.

The main difference with respect to the real scenario is that here the TLE database is read only to extract information about primary objects, and the secondary objects (debris) are created virtually by perturbing the state of the primaries. The conjunction time $t_C$ is chosen randomly within the boundaries of the interval of analysis $[t_0, t_f]$, and a screening time $t_S$ is also generated to represent the moment when the collision was detected by the operators. Regarding the state perturbation, we shift the position of the object at $t_C$ by a random value in the interval $[50, 150]$ m and rotate its velocity by a random angle between 0 and $2\pi$. Since this perturbation is small, the virtual object created from it will be close to the primary at $t_C$, increasing the chances of having a high-probability collision between them. If the collision probability with this new virtual object appears to be above the ACPL, we compute a collision avoidance manoeuvre for it in the same way as described in the previous section.

Regarding the creation of subsequent manoeuvres in this virtual scenario, the process must be repeated by generating multiple secondary objects in the future. Then, the collisions and manoeuvres against those secondary objects are computed with respect to two primary objects: the one resulted from applying the first-stage manoeuvre and the one without the effect of the manoeuvre. This allows a comparison of the cost of applying or not applying the manoeuvre in terms of future encounters.

### III.iii Description of a manoeuvre record

The computed manoeuvres, for both the real and virtual scenarios, are stored into a database. Not only the $\Delta$V is saved, but also every piece of data associated to it. By storing this additional data we allow for a deeper analysis of the contents of the database, and also the possibility to reproduce these results.

Below are briefly described every field comprising one record in the database:

- *_id*: A unique identifier of the manoeuvre record.
- *collision*: This field contains information about the collision avoided by the manoeuvre. Although in this work it is always referred to primary and secondary object, when talking about a collision we will refer to the objects taking part of it as object 1 and object 2, for the sake of generality. The collision field is described by:
  - *covar_o1_ct*: Covariance matrix of object 1 at conjunction time (a $6 \times 6$ matrix).
  - *covar_o2_ct*: Covariance matrix of object 2 at conjunction time (a $6 \times 6$ matrix).
  - *mean_o1_ct*: Mean state of object 1 at conjunction time, given as a 6-dimensional vector

of Cartesian coordinates, where the 3 first elements represent the position of the object (in meters) and the last 3 the velocity (in $\mathrm{m\,s^{-1}}$).

- *mean_o2_ct*: Mean state of object 2 at conjunction time.
- *miss_distance*: Distance between the mean states at conjunction time, the square root of the left hand-side in Eq. 1.
- *o1*: This field contains general information about object 1 in the collision:
  - *mass*: Mass of the object in kilograms. This value is fixed to $1\ \mathrm{k}$, since it is not part of the collision probability computation.
  - *measurements*: An array of the measurements of the object, only available for real objects read from the TLE database. Every measurement defined with the following fields:
    - *time*: Epoch of the measurement, as a Julian date in seconds.
    - *value*: Vector or orbital elements $(a, e, i, \Omega, \omega, M)$.
  - *name*: Name of the object, as given from the TLE database. In case it is a virtual object, the name is fixed to `"AUTOGENERATED-OBJECT"`.
  - *radius*: Radius of the object $(\mathrm{m})$. Recall that we assign a radius of $9\ \mathrm{m}$ to primaries and $1\ \mathrm{m}$ to secondaries.
- *o2*: This field contains general information about object 2 in the collision (same fields as o1).
- *probability*: Probability of collision. Note that this value will always be greater than the Accepted Level of Collision Probability (ACPL), which is set to $0.0001$ in this work.
- *screening_time*: Indicates the moment when the collision was detected (or screened), as a Julian date converted in seconds.
- *state_o1_st*: Mean state of object 1 at screening time.
- *state_o2_st*: Mean state of object 2 at screening time.
- *time*: Conjunction time $(t_C)$, as a Julian date converted in seconds.

- *deltaV*: The computed manoeuvre, expressed in $\mathrm{m\,s^{-1}}$ as a 3-dimensional vector in the Tangent-Normal-h reference frame $(\{\hat{\mathrm{t}}, \hat{\mathrm{n}}, \hat{\mathrm{h}}\})$. This vector contains both the direction and the magnitude of the collision avoidance manoeuvre.
- *object_to_manoeuvre*: Information about the object on which the manoeuvre will be applied, i.e, the primary object. Note that this information is also recorded in the collision field, either as object 1 or object 2.
- *posterior_collision_probability*: Probability of collision after applying the collision avoidance manoeuvre. Note that this value will always be lower than the ACPL.
- *scenery_type*: Whether the manoeuvre has been computed in a real scenario or in a virtual scenario.
- *subsequent_manoeuvres_after_application*: Array of collision avoidance manoeuvres needed in the future (after $t_C$) in case we apply this manoeuvre to the primary object. Note that every element of this array contains a whole record of a collision avoidance manoeuvre.
- *subsequent_manoeuvres_no_application*: Array of collision avoidance manoeuvres needed in the future (after $t_C$) in case that we decide not to apply this manoeuvre to the primary object.
- *time_of_application*: Indicates the time when this manoeuvre must be applied, as a Julian date in seconds.

## IV. Exploiting the dataset of manoeuvres

In this section, different possibilities are discussed to exploit a collision avoidance manoeuvre dataset, assuming that it has been created following the processes described in the previous section. There is a focus on the fields "subsequent_manoeuvres_after_application" and "subsequent_manoeuvres_no_application", as this information represents the future consequences of a manoeuvre, which is not available at real time. The dataset used in the following experimentation comprises a total of 529 collision avoidance manoeuvres, most of them based on a virtual scenario, due to the low computational time required to compute them, in comparison with a real one (See Section 3.2).

### IV.i Predicting the future consequences of a collision avoidance manoeuvre

As it was introduced above, the most valuable information in our dataset is related to the consequences of the manoeuvre, which are stored in terms of the need to avoid

subsequent collisions. In order to quantify these consequences, two quantities are defined, namely the *future risk* and the *future cost*, as follows:

$$future\_risk = \frac{\sum_{SM} \|PoC\|}{\sum_{\widehat{SM}} \|PoC\|}. \quad [4]$$

$$future\_cost = \frac{\sum_{SM} \|\Delta \mathrm{V}\|}{\sum_{\widehat{SM}} \|\Delta \mathrm{V}\|}. \quad [5]$$

Here, PoC refers to the probability of collision, and *SM* and $\widehat{SM}$ denote the set of subsequent manoeuvres needed after applying or not applying a collision avoidance manoeuvre, respectively. If any of these ratios is above 1, it can be considered that the application of the manoeuvre involves a certain amount of risk for the future situation of the spacecraft. To avoid null denominators in these measures, the database is filtered to ensure that every manoeuvre has at least one element in both the fields "subsequent_manoeuvres_after_application" and "subsequent_manoeuvres_no_application".

Once the consequences of a manoeuvre have been quantified, our goal here is to **create predictive models of those quantities** based on the information stored for a collision avoidance manoeuvre. Since the two quantities defined in Eqs. 4 and 1 are continuous, we will achieve this goal by performing a Multiple Regression Analysis (MMR).[JWHT13] MMR look at causal relations between a set of predictors (independent variables) and a response (dependent) variable.

Table 1 shows a summary of all the variables extracted from every collision avoidance manoeuvre in the database, divided into predictors and responses. Although the name of most of the predictors is self explanatory, below is a short description of the feature set:

1. *collision_probability*: Probability of collision at conjunction time.
2. *DV_magnitude*: Magnitude of the $\Delta \mathrm{V}$ applied to avoid the collision, in $\mathrm{m\,s^{-1}}$.
3. *miss_distance_ct*: Miss distance at conjunction time, in $\mathrm{m}$.
4. *norm_covar_primary*: One norm (maximum absolute column sum) of the covariance matrix of the primary object ($\mathrm{m^2}$).
5. *norm_covar_primary*: One norm (maximum absolute column sum) of the covariance matrix of the secondary object ($\mathrm{m^2}$).
6. *norm_rel_pos_ct*: Norm of the relative position ($\|r_p - r_s\|$) at conjunction time ($\mathrm{m}$).
7. *norm_rel_vel_ct*: Norm of the relative velocity ($\|v_p - v_s\|$) at conjunction time ($\mathrm{m\,s^{-1}}$).
8. *period_ratio*: Ratio between the periods of the primary and the secondary, in $\mathrm{s}$ ($T_p/T_s$).
9. *pos_angle_ct*: Angle between the position vectors of primary and secondary, in radians.
10. *time_difference_ct_ta*: Time difference between the conjunction and the application of the manoeuvre, expressed in periods of the primary object.
11. *vel_angle_ct*: Angle between the velocities of primary and secondary, in radians.

The fact that the working database is comprised mainly of virtual scenario-based manoeuvres introduces a clear bias in the distribution of some of the predictors. Since the secondary object in a virtual encounter is derived created a soft perturbation in the primary, the periods of the two objects are almost the same, and thus the variable `period_ratio` is constantly around 1. Similarly, the angle between position vectors (variable `pos_angle_ct` in Table 1) is approximately 0 across the whole database, due to the position of the virtual secondary object is defined just by shifting the position of the primary. Thus, the information of these variables will not be used in here, although they might be considered for future experiments with richer databases.

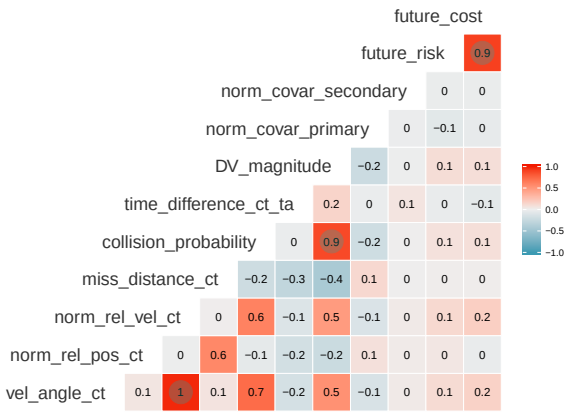


Fig. 3: Within features correlation (Pearson coefficient). Values above $0.75$ are marked with a circle and considered as highly correlated variables.

Before going into the model building, looking at data relation is a sensible step to understand how the different variables interact together. Correlations look at trends shared between two variables and can be calculated in different ways. The most usual measure is the Pearson

Table 1: Summary of the variables extracted from the manoeuvre database for analysis purposes. The horizontal line marks which variables will be used as predictors (independent variables) and which are considered as responses (dependent variables)

| variable | mean | sd | min | max | skewness | kurtosis |
|---|---|---|---|---|---|---|
| collision_probability | 6.308E-04 | 3.809E-04 | 1.080E-04 | 2.129E-03 | 2.327 | 8.511 |
| DV_magnitude | 2.520E-02 | 8.172E-03 | 1.270E-03 | 5.527E-02 | 0.698 | 5.150 |
| miss_distance_ct | 1.039E+02 | 3.987E+01 | 1.809E+01 | 2.423E+02 | 0.252 | 2.983 |
| norm_covar_primary | 1.224E+04 | 8.102E+02 | 1.015E+04 | 1.361E+04 | -0.431 | 2.448 |
| norm_covar_secondary | 1.222E+06 | 7.814E+04 | 1.044E+06 | 1.361E+06 | -0.171 | 2.151 |
| norm_rel_pos_ct | 1.337E+02 | 2.842E+01 | 8.769E+01 | 2.424E+02 | 0.741 | 3.910 |
| norm_rel_vel_ct | 8.953E+03 | 3.939E+03 | 2.153E+03 | 1.531E+04 | -0.278 | 1.844 |
| period_ratio | 1.000E+00 | 3.009E-05 | 9.999E-01 | 1.000E+00 | 0.099 | 2.443 |
| pos_angle_ct | 1.473E-05 | 5.074E-06 | 1.883E-06 | 3.386E-05 | -0.243 | 3.507 |
| time_difference_ct_ta | 2.537E+00 | 6.286E-02 | 2.200E+00 | 2.700E+00 | -0.287 | 5.598 |
| vel_angle_ct | 1.443E+00 | 7.819E-01 | 3.038E-01 | 3.127E+00 | 0.339 | 2.295 |
| future_cost | 8.357E-01 | 4.683E-01 | 3.125E-03 | 2.184E+00 | 0.603 | 2.872 |
| future_risk | 8.474E-01 | 5.244E-01 | 6.052E-02 | 2.530E+00 | 0.982 | 3.544 |

coefficient, which is the covariance of the two variables divided by the product of their variance, scaled between 1 (for a perfect positive correlation) to -1 (for a perfect negative correlation), being 0 a complete randomness. In Figure 3 ah correlation heat-map is shown for the set of features enumerated above. Regarding the within correlation between predictors, only two of pairs of variables, namely (`norm_rel_vel_ct`,`vel_angle_ct`) and (`DV_magnitude`,`collision_probability`), can be considered as highly correlated since they present values of the Pearson coefficient above $0.8$. They will impart nearly exactly the same information to the model, but including both will possible infuse the model with noise. Thus, one of them will be removed. Concretely, variables `norm_rel_vel_ct` and `DV_magnitude` are removed from the set of predictors.

On the other hand, if we check the within correlation between the two dependent variables (future risk and future cost), the value of the Pearson coefficient is above $0.9$, which is almost a perfect positive correlation. This correlation can be seen graphically in Figure 4, along with the result of computing a linear regression model between them (F-statistic: 1074 on 1 and 248 DF, p-value: $\leq 2.2e-16$). There is no need to predict both dependent variables, and thus, from now on, we will focus exclusively in the prediction of the **future risk**.

The basis for the regression analysis used here is the popular least squares method, which tries that the overall solution minimizes the sum of the squares of the errors. An important assumption of this method is the normality of the error distribution, and thus, the normality of the dependent variable. According to the literature,[GW16] a quick

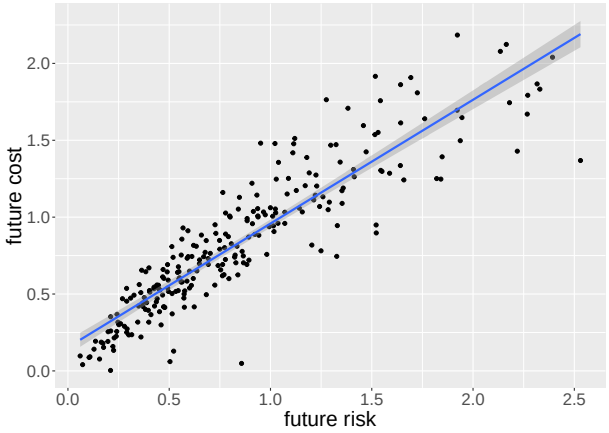


Fig. 4: Within correlation between dependent variables. The blue line represents the fitted linear model that regresses the future cost in terms of the future risk.

test for testing the normality assumption can be done by checking whether the skewness of the distribution is close to 0 and the kurtosis is close to 3. Focusing on the information of the future risk shown in Table 1, it can be appreciated that the distribution is moderately normal, but a bit skewed to the right. In order to improve the normality of the dependent variable, a Box-Cox transformation[Osb10] will be applied on every experiment described below.

In order to make the modelling process more efficient, it is useful to check whether there is any notorious global movement shared between any of the features and the response, an if so, perform the regression analysis hierarchi-

cally by using first the predictors which make a significant statistical contribution to the model.[CCWA13] However, as it can be seen in Figure 3, none of the features show a high correlation with respect to the dependent variable. Thus, an automatic way to search for the best model blindly is needed. In this work, we make use of a popular regularization regression method, the Elastic Net.

The Elastic Net is a model that builds over the classical least squares regression, but uses a penalization on the size of the coefficients in the objective function to try to regularize ("shrink") them. This means that the estimated coefficients are pushed towards 0, to make them work better on new datasets (optimizing the model for prediction). This allows to use complex models and avoid over-fitting at the same time, excluding irrelevant variables.[FHT10] More specifically, considering a sample of N cases, each of which consists of p features and a single response. Let $y_i$ be the response and $x_i = (x_1, x_2, \ldots, x_p)^T$ be the feature vector for the ith case. Then the objective of Elastic Net is to solve:

$$\min_{\beta_0,\beta} \frac{1}{N}\sum_{i=1}^{N}(y_i - \beta_0 - x_i^T\beta_i)^2 + \lambda\left[\frac{(1-\alpha)}{2}||\beta||_2^2 + \alpha||\beta||_1\right]. \quad [6]$$

Here $\beta = \{\beta_i\}_{i=1}^{p}$ are the regression coefficients ($\beta_0$ for the intercept) and $\lambda$ is a pre-specified penalty term that determines the amount of regularization. Depending on the size of the penalty term, Elastic Net shrinks less relevant predictors to (possibly) zero. It is an hybrid approach that blends penalization of the L2 and L1 norms of the coefficients. The $\alpha$ hyper-parameter is between 0 and 1 and controls how much L2 or L1 penalization is used. When $\alpha = 0$ this method is known as Ridge Regression and when $\alpha = 1$ we have the LASSO (Least Absolute Shrinkage and Selection), popular for variable selection and extremely effective with high-dimensional data. Elastic-net is a compromise between the two that attempts to shrink and do a sparse selection simultaneously.

Since normally the Elastic Net will select less variables than Ordinary Least Squares (OLS), its forecast will have much less variance at the cost of a small amount of bias in sample. Elastic Net regression puts constraints on the size of the coefficients associated to each variable. However, this value will depend on the magnitude of each variable. It is therefore necessary to centre and reduce, or standardize, the variables. By doing so, comparing the relative importance of the coefficients of the final model becomes also much easier.

One of the most important features of the Elastic Net is that it can deal with lot of variables, and thus, we can take advantage of it to model not only linear relationships between variables, but also combined predictors and non linear relationships. In concrete, here we will experiment with polynomial regressions up to order 3, introducing also interaction terms to allow the exploration of synergistic effects between predictors. The size of the different set of predictors used in this experiment is given in Table 2.

Table 2: Size of the set of predictors used in each of the experiments run for the Elastic Net cross validation.

| | No interaction terms | With interaction terms |
|---|---|---|
| Order 1 | 7 | 28 |
| Order 2 | 14 | 98 |
| Order 3 | 21 | 210 |

Table 3: Cross validation results for finding the value of the Elastic Net hyperparameters ($\alpha$ and $\lambda$) that minimizes the Mean Squared prediction Error.

| Order | I.terms | Min. MSE | $\alpha$ | $\lambda$ |
|---|---|---|---|---|
| **1** | **No** | $\mathbf{0.357 \pm 0.026}$ | **0.20** | **0.16** |
| 2 | No | $0.358 \pm 0.031$ | 0.45 | 0.08 |
| 3 | No | $0.359 \pm 0.029$ | 0.50 | 0.08 |
| 1 | Yes | $0.373 \pm 0.03$ | 0.45 | 0.10 |
| 2 | Yes | $0.374 \pm 0.03$ | 0.80 | 0.09 |
| 3 | Yes | $0.359 \pm 0.026$ | 1.00 | 0.08 |

In order to select the best model, this work takes a K-fold cross validation approach, with $K = 10$, that uses grid search to find the optimal $\alpha$ hyper-parameter while also optimizing the $\lambda$ hyper-parameter, by minimizing the cross-validated Mean Squared prediction Error (MSE). The procedure will be applied for the 6 different sets of predictors showed in Table 2. Possible values of $\alpha$ go from 0 (Ridge) to 1 (LASSO) with a step size of 0.05, and for the penalty term, $\lambda$, a set of 1000 values is used, in a range that is adaptively chosen by the implementation of the algorithm. The execution of the Elastic Net is performed via the coordinate descent algorithm,[FHT10] and it will be run 20 times for every fold and every combination of the hyper-parameters $\alpha$ and $\lambda$, in order to ensure the stability of the results. All the experiments have been implemented in the R Statistical Environment [‡], making use of the package *glmnet*.

The results of the experimentation are shown in Table 3 as a cross validation table that represents when the mean squared cross validation error varies is minimized in terms of the values of $\alpha$ and $\lambda$. As it can be seen, the cross validation error is very similar for every set of predictors

[‡] The R project for Statistical Computing: `https://www.r-project.org/`

tested, which give a sign that adding high order terms and interaction terms does not help to find better relationships with the dependent variable. Also, it is remarkable how the value of $\alpha$ increases along with the size of the predictor set, which is indicating that the Elastic Net needs to put more emphasis in selecting variables rather than shrinking them in order to keep the quality of the model. Regarding the interpretation of the MSE column itself, we have that the corresponding R-squared coefficient for these results is around $0.01$, which means that the resulting models explain barely a $1\%$ of the variance contained in the dependant variable.

Table 4: Top resulting regularized coefficients for the best model and best $\lambda$ found after the experiment with Elastic Net regression.

| feature | beta |
|---|---|
| vel_angle_ct | 2.995E-02 |

The value of $\lambda$ that minimizes the mean cross validation error in Table 3 is 0.16 for the model with order 1 and no interaction terms, trained with a value of 0.2 for the hyper-parameter $\alpha$. Using this configuration, we get the estimated matrix of coefficients ($\beta$-matrix), to analyse which predictors are the most important in explaining the variance of the response. Table 4 shows the top coefficients resulted from this regression process. As it can be seen, there is only one non-zero coefficient after applying Elastic Net with this configuration, the variable `vel_angle_ct`, which results as the most important feature for predicting the future risk here. However, although the resulting model is fair in terms of statistical significance (F-statistic: 3.523 on 1 and 248 DF, p-value: 0.06171), the coefficient is too low, which means that they it contribute to produce a better model than the one achieved by a null model (horizontal line, no slope). In fact, as it was introduced above, the R-squared coefficient of determination marks $0.0093$, an extremely low value for a regression model.

Thus, we cannot affirm that, using this database, a significant relationship between the future risk and the predictors has been found. Nonetheless, it must be remarked that these results are biased mainly by the fact that the database used is completely virtual, and the variance of some randomly generated quantities such as the angle between the velocity vectors is bounded in the current data generation process. Hence, more experiments are needed across wider databases to extract more robust conclusions. Nonetheless, the methodology of analysis proposed in this section is valuable for the rest of subsequent experiments that one wants to make in this context.

### IV.ii Recommending safer manoeuvres

Implementing the prediction process described above into a real system could lead to situations where an operator is planning a collision avoidance manoeuvre that is optimal in terms of magnitude and time of application, but in turn is risky in terms of future consequences. In this section, a novel methodology is presented to solve this issue, aimed at providing the operator with alternative safer manoeuvres that may be globally better for the space environment.

The approach taken here is based on the database of manoeuvres and the prediction of their future consequences. Assuming that an operator faces a new encounter where the computed optimal manoeuvre presents a high value of the predicted future risk, the problem consists in finding "close" past encounters in the database and retrieve the $\Delta$V applied to avoid that encounter and the real future consequences resulted from that avoidance. Based on that, the operator will be provided with a *balanced* ranking of alternative manoeuvres that constitutes a fair trade-off between the similarity of the encounters, the magnitude of the $\Delta$V to apply and the future risk associated to that $\Delta$V.

Two issues arise when formulating this problem. On the one hand, there is a need of quantifying the proximity of two encounters. In this work, the encounter distance (ED) is computed as an Euclidean distance between three geometric elements: the relative speed, the angle between velocities and the miss distance. To avoid scale variance, all those elements in the database are standardized to have 0 mean and unit standard deviation. On the other hand, it is also necessary to balance the final ranking in terms of encounter similarity, $\| \Delta \mathrm{V} \|$ and future consequences. To achieve this, we will make use of Rank Aggregation techniques.

Rank Aggregation can be defined as the task of combining several individual sorted lists (also called base rankers) in order to generate a better one which aggregates information from each of the input lists.[LLQ+07] In recent years, Rank Aggregation techniques have become more sophisticated and have been used in different applications.[RFGPC17]

Following the notation of,[PDD09] Rank Aggregation can be expressed as an optimization problem, where one would like to find an optimal ranking which would be as "close" as possible to all the base rankers simultaneously. In terms of an objective function, this can be formulated as follows:

$$\Phi(\delta) = \sum_{i=1}^{m} \omega_i d(\delta, L_i), \qquad [7]$$

where $L_i$ is the $i^{th}$ base ranker, $\delta$ is a proposed aggregated ranking of length $|L_i|$, $\omega_i$ is the importance associated to

the base ranker $L_i$, and $d$ is a distance function between $\delta$ and $L_i$. Using previous formulation, the goal of any Rank Aggregation method can be defined as: finding a ranking $\delta^*$ that minimizes the function $\Phi$.

The distance used here to measure the differences between rankings is the *Weighted Spearman footrule distance*, defined as:

$$WS(\delta, L_i) = \sum_{t \in L_i \cup \delta} |W(r^{\delta}(t) - W(r^{L_i}(t))| \times |r^{\delta}(t) - r^{L_i}(t)|, \quad [8]$$

where $r^{L_i}(t)$ is the position of element $t$ in the rank $L_i$. This metric can be understood in terms of sum of penalties for moving an arbitrary element $t$ from the position $r^{\delta}(t)$ to the position $r^{L_i}(t)$, adjusted by the difference in scores between the two positions.

The Rank Aggregation obtained by optimizing the Spearman distance is also called *footrule optimal aggregation* and it is identified as an NP-hard problem for partial lists and rankings.[BA03] Thus, different optimization algorithms have tried to tackle this problem, such Genetic Algorithms (GA), which will be applied here, following the guidelines of the literature.[PDD09]

Table 5: Summary of the most relevant information related to the test manoeuvre used to show the recommendation of safer manoeuvres.

| Variable | Value |
|---|---|
| Relative speed ($\mathrm{m\,s^{-1}}$) | 7287.340 |
| Angle between velocities (rad) | 1.073 |
| Miss distance ($\mathrm{m}$) | 105.765 |
| $\parallel \Delta \mathrm{V} \parallel$ ($\mathrm{m\,s^{-1}}$) | 0.022 |
| time of application (periods) | 2.600 |
| Predicted future risk | 0.726 |
| Predicted future cost | 0.738 |

In order to better illustrate the ideas presented above, next is detailed an example that applies this AI-based decision support system to recommend alternative manoeuvres to a "dangerous" one, extracted randomly from the database. Table 5 shows the most relevant information of this manoeuvre, along with the forecast of its future consequences, computed using the prediction model selected in the previous section. As it can be seen, the value of the predicted future risk is much larger than 1, which is a sign that the impact of this decision is not safe for the future environment. The first three variables of the table define the encounter quantitatively, and are used to compute the distance of this collision against the rest of encounters stored in the database. Since the goal here is to find safer decisions, only the encounters whose associated future risk is lesser than 0.726 (the predicted risk of the dangerous manoeuvre) are analysed. In this case, a total of 121 manoeuvres are considered as possible alternatives. From them we create 4 independent base rankers, sorting the values of the encounter distance, the magnitude of $\Delta$V, the future risk and the future cost respectively.

Table 6: Ranking of the top 10 recommended manoeuvres as a safer alternative to the manoeuvre showed in Table 5. ED refers to the Encounter distance, t to the time of application in periods before the conjunction time, F.risk to the future risk and F.cost to the future cost.

| | ED | $\parallel \Delta \mathrm{V} \parallel$ | t | F.risk | F.cost |
|---|---|---|---|---|---|
| **1** | **1.447** | **0.007** | **2.400** | **0.439** | **0.559** |
| 2 | 1.949 | 0.025 | 2.500 | 0.061 | 0.097 |
| 3 | 0.056 | 0.025 | 2.500 | 0.364 | 0.655 |
| 4 | 3.039 | 0.043 | 2.500 | 0.073 | 0.041 |
| 5 | 0.072 | 0.018 | 2.400 | 0.559 | 0.754 |
| 6 | 0.074 | 0.027 | 2.500 | 0.575 | 0.414 |
| 7 | 1.411 | 0.018 | 2.500 | 0.105 | 0.087 |
| 8 | 0.969 | 0.010 | 2.400 | 0.290 | 0.537 |
| 9 | 0.171 | 0.031 | 2.600 | 0.473 | 0.602 |
| 10 | 0.931 | 0.026 | 2.500 | 0.109 | 0.092 |

Finally, the rank aggregation process is executed over the 4 base rankers, resulting in the final aggregated ranking that will be provided to the operator. The underlying genetic algorithm executed for this purpose is run with the default parameters set in[PDD09] (Population size = 100; Cross-over probability = 0.4; Mutation probability = 0.01; Maximum number of iterations = 2000). Regarding the stopping criteria, the algorithm will be stopped when the fitness of the best solution does not change in 15 iterations. The whole process is run 20 times, and the execution with the minimum fitness value is chosen as final solution.

The top 10 alternative manoeuvres obtained from the final aggregated ranking are shown Table 6. The time of application associated to each manoeuvre has been included as additional information for the operator, although it does not take part of the rank aggregation. As it can be seen, the first recommended manoeuvre is not the closest the nominal one, but it manages to reduce the risk by applying a small burn. Based on the priorities of the operator in the context of the manoeuvre, he/she would be able to choose one of these recommendations. In fact, the values of $\omega_i$ in Eq. 7 can be used to give more or less importance to the variables of the ranking, making this approach adaptive to the needs of the operation.

## V. Conclusions and future work

This paper has presented new approaches to support space traffic management with additional information about collision avoidance manoeuvres. Based on a database of past encounters and manoeuvres, different techniques from the field of Artificial Intelligence are applied with two main goals. On the one hand, the prediction of the consequences of a manoeuvre, quantified as ratios of subsequent collision probabilities subsequent $\Delta$Vs. On the other hand, the recommendation of safer manoeuvres in cases where the predicted future risk is too high. Since there are no public databases of past manoeuvres to support this type of research, one of the keys of this work is the development of a methodology for creating synthetic encounters and manoeuvres, from both a realistic and a virtual point of view. However, the limitation in terms of computational time has restricted the experimentation of this paper to work with a database comprised mainly of virtual-scenario based manoeuvres.

Although some of the assumptions and formulations made in this work are simplistic, the approaches presented pave the way to a new approach for space traffic management, taking advantage of the data obtained from previous operations. As future work, several issues would need to be extended and improved, including:

- The optimization of the time of application during the data generation process.
- The upgrade to a more accurate dynamical model including orbital perturbations to compute the effects of a manoeuvre,
- The analysis of a dataset fully comprised of real scenario-based manoeuvres, to study whether the processes described here are valid also in a fully realistic environment.
- A formal test that studies whether the top manoeuvres recommended as alternatives to a dangerous one achieve a real reduction of the future risk when applied to the actual encounter.
- The redefinition of the impact collision probability (future risk) and future cost taking into account a lot more factors than the collision probability. The formulations of Eqs. 4 and 5 are still too superficial to be used in a realistic environment as a risk quantity.

## References

[Alf07] Salvatore Alfano. Review of conjunction probability methods for short-term encounters (aas 07-148). *Advances in the Astronautical Sciences*, 127(1):719, 2007.

[BA03] MM Sufyan Beg and Nesar Ahmad. Soft computing techniques for rank aggregation on the world wide web. *World Wide Web*, 6(1):5–22, 2003.

[Bat99] Richard H Battin. *An introduction to the mathematics and methods of astrodynamics*. Aiaa, 1999.

[CCWA13] Jacob Cohen, Patricia Cohen, Stephen G West, and Leona S Aiken. *Applied multiple regression/correlation analysis for the behavioral sciences*. Routledge, 2013.

[Cha08] F Kenneth Chan. *Spacecraft collision probability*. Aerospace Press El Segundo, CA, 2008.

[FHT10] Jerome Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33(1):1, 2010.

[Gro05] Giovanni F Gronchi. An algebraic method to compute the critical points of the distance function between two keplerian orbits. *Celestial Mechanics and Dynamical Astronomy*, 93(1):295–329, 2005.

[GW16] Frederick J Gravetter and Larry B Wallnau. *Statistics for the behavioral sciences*. Cengage Learning, 2016.

[JWHT13] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An introduction to statistical learning*, volume 112. Springer, 2013.

[KAS05] H Klinkrad, JR Alarcon, and N Sanchez. Collision avoidance for operational esa satellites. In *4th European Conference on Space Debris*, volume 587, page 509, 2005.

[LLQ+07] Yu-Ting Liu, Tie-Yan Liu, Tao Qin, Zhi-Ming Ma, and Hang Li. Supervised rank aggregation. In *Proceedings of the 16th international conference on World Wide Web*, pages 481–490. ACM, 2007.

[Osb10] Jason W Osborne. Improving your data transformations: Applying the box-cox transformation. *Practical Assessment, Research & Evaluation*, 15(12):1–9, 2010.

[PDD09] Vasyl Pihur, Susmita Datta, and Somnath Datta. Rankaggreg, an r package for weighted rank aggregation. *BMC bioinformatics*, 10(1):62, 2009.

[Pre70] Andras Prekopa. On probabilistic constrained programming. In *Proceedings of the Princeton symposium on mathematical programming*, pages 113–138, 1970.

[RFGPC17] Víctor Rodríguez-Fernández, Antonio Gonzalez-Pardo, and David Camacho. Modelling Behaviour in UAV Operations using Higher Order Double Chain Markov Models. *IEEE Computational Intelligence Magazine*, In press, Nov 2017.

[SAJ+16] Romain Serra, Denis Arzelier, Mioara Joldes, Jean-Bernard Lasserre, Aude Rondepierre, and Bruno Salvy. Fast and accurate computation of orbital collision probability for short-term encounters. *Journal of Guidance, Control, and Dynamics*, 2016.

[SJ03] Hanspeter Schaub and John L Junkins. *Analytical mechanics of space systems*. Aiaa, 2003.

[VC08] Massimiliano Vasile and Camilla Colombo. Optimal impact strategies for asteroid deflection. *Journal of guidance, control, and dynamics*, 31(4):858, 2008.