Characterizing a Novel Coordinated Optimal Avoidance Maneuver Framework for Space Traffic Management (STM)

André Antunes de Sá, Matthew Shouppe, Shota Takahashi, Felix Portillo, Masafumi Isaji, Siamak Hesar
Kayhan Space
Sita Sonty
Boston Consulting Group

ABSTRACT

The continued increase of Resident Space Objects (RSOs) poses a serious space congestion problem, driving an aggressive growth of on-orbit collision likelihood and endangering the potential of tomorrow's space economy - one that is forecast to become a trillion-dollar industry by 2040. Yet, the methodologies currently employed by satellite operators for collision risk mitigation are often inadequate, non-uniform, uncoordinated, and are increasingly vulnerable to being overcome by events. Operators have not reached a consensus on collision avoidance risk metrics and actionable thresholds. There are no explicit norms of behavior for risk mitigation and coordination between operators, and space traffic management (STM) authority, regulation, and policy remain in their infancy. STM needs have grown to a global scale, involving multiple nation's assets in orbit. Such high entropy in the collision avoidance mitigation processes is often the main cause of costly inefficiencies at best and potential damage to in-orbit assets at worst. Spacecraft tracking uncertainties and inaccuracies associated with collision avoidance risk metrics have been connected to higher avoidance maneuver fuel usage and a greater number of avoidance maneuvers. Such increased resource usage bears the financial cost to each operator, not to mention the operators whose assets might be rendered vulnerable by others'. Operational uncertainty of the conjuncting ("secondary") object, due to uncoordinated avoidance or orbit maintenance maneuvers, can have a similar effect. Most concerning, sub-optimal and uncoordinated maneuvers between two oper-ator's assets can increase the risk of collisions, and they also tend to increase encounter rates and the need for further avoidance maneuvers during a spacecraft's lifetime. Lack of process automation significantly increases the response time of collision avoidance and further complicates coordination, which can grow quickly in a scenario involving mul-tiple organizations. The cost of these inefficiencies includes increased fuel expenditure, increased regulatory fees, the potential cost of litigation/mediation, decreased revenues from the utilization of vulnerable assets, increased mission downtime as more time is spent performing avoidance maneuvers, and decreased mission lifetime due to faster fuel depletion. Collision risk mitigation will soon become too costly, unless the risk metric uncertainties and deficiencies in avoidance maneuver selection, including lack of coordination, are simultaneously addressed. To help address the latter, we investigate a novel optimal avoidance maneuver framework.

Optimal avoidance maneuver planning is a complex problem, especially when automation and spacecraft coordination are required. The current diversity of collision avoidance processes employed by satellite operators is driven by their different mission requirements, long-time risk exposure, and corporate and cultural considerations, among others. Thus, any viable avoidance maneuver framework must be flexible and customizable, considering the mission and/or organization needs of each stakeholder. In this context, an optimal avoidance maneuver solution is one that best fulfills the interests of an organization, or of multiple organizations when coordination is considered. The framework must also level the playing field with respect to the scale of financial asset loss/risk encumbered by each respective party through means of pricing/financial risk assessment.

Kayhan Space has developed, and continues to update, an optimal avoidance maneuver framework to address the needs identified above. It combines a user-facing platform for asset automation configuration with an expansive maneuver suggestion engine. This maneuver suggestion engine is a critical part of the avoidance maneuver framework, and it is the main focus of this paper. Brief commentary on the Kayhan Space avoidance maneuver framework as part of a broader solution to the current Space Traffic Management (STM) needs of the space industry, including the

incorporation of rules of the road and data sharing, are also provided. The maneuver suggestion engine has a modular design built to be highly flexible, expandable, efficient, and parallelizable. It generates maneuver tradespaces for a conjunction event based on a variety of available settings and optimization algorithms. These algorithms all accept a polymorphic object-oriented optimization problem model as guiding input, featuring plug-and-play metrics that can be used as objective functions or as constraints, e.g., miss distance, probability of collision (PC), maneuver direction, shared burden between spacecraft, etc. The combination of these metrics allows for capturing the operational constraints and interests of spacecraft owners and producing optimal avoidance maneuver suggestions. A variety of optimization algorithms are supported, including nonlinear programming (NLP) using the Interior Point Optimizer (IPOPT) solver, global grid search, heuristic and metaheuristic methods such as ant colony optimization, gradient descent with Nesterov-momentum, and a trivial loop-back fixed solution. Some of these metrics and algorithms are described in this paper, and their performance for different optimization problem case studies are characterized both in terms of processing speed and optimality. The case studies include a combination of PC, maneuver magnitude and direction, and shared-burden minimization and constraint targeting. Two lower-fidelity modeling techniques commonly employed in avoidance maneuver generation are also characterized, specifically (1) the use of linearized dynamics to compute spacecraft relative position changes at Time of Closest Approach (TCA) from a preceding maneuver, as well as (2) the use of risk metrics computed at the original TCA as a proxy for metrics corresponding to the new TCA, which is affected by the avoidance maneuver. Planned future research efforts include: characterizing the entire avoidance maneuver framework, furthering its automation capabilities, supporting multi-objective functions in the maneuver suggestion engine, and expanding the supported risk metrics and algorithms.

1. INTRODUCTION

While the space industry continues to grow at an accelerating pace, the slow evolution of space debris mitigation practices and their adoption rate threatens the future sustainability of the space environment [1]. There is an overall consensus that effective STM will require reducing uncertainty in space situational awareness (SSA), also referred to as space domain awareness (SDA), standardizing actionable metrics and norms of behavior, and enabling coordination between operators [2, 3, 4]. Multiple sectors have been advocating for establishing STM regulations and processes, from congressional leaders [e.g., 5] to commercial operators [e.g., 6, 7, 8].

The first pillar of collision risk mitigation is conjunction risk assessment, under the umbrella of SSA. If properly quantified, the uncertainty associated with this assessment drives the achievable efficiency of collision risk mitigation, i.e., greater uncertainty is linked to increased maneuver rates and fuel usage. This uncertainty is tied to the collision risk metric and spacecraft trajectory data used by an operator to quantify collision risk and drive mitigation efforts. For a representative example of an operator's risk mitigation processes, Alfano et al. [2] investigated the Space Data Association [8]'s operator survey and found that operators use a diverse set of "Go/No-Go"" risk metrics and actionable thresholds without a clear consensus. After categorizing the risk metrics, they also found that the metrics that were the simplest to evaluate and required the least amount of input data negatively correlated with actionable conjunctions. The same correlation is observed with trajectory uncertainty.

Different, but complementary, approaches for improving SSA data quality have been identified, from the constant technological advancements in ground sensor capabilities and deployments, to on-orbit tracking [9, 10, 11], data fusion [12], and higher fidelity propagation dynamics. Drag modeling is of significant importance, given it's one the largest uncertainty contributors [13, 14], especially with the increased solar activity expected in the near future. Data availability for the risk mitigation process dictates the need for collaborative sharing of authoritative data, including resident space object (RSO) ephemerides, maneuver plans, observations, dimensions, mass, and attitude profiles. Since SSA providers usually do not have access to much of this data owned by operators, a centralized, trustworthy, and practical repository of data is vital to the success of STM efforts and the space industry's future.

Ineffective risk mitigation has a huge impact on the operational cost of space assets, due to faster fuel depletion, shortened mission lifetimes, increased mission downtime, and overall loss of revenue. According to the European Space Agency (ESA), an average of three to four collision avoidance maneuvers are performed per year for each of their orbiting spacecraft [15]. Critical decision-making processes involving time-sensitive risk mitigation decisions are also adversely impacted by lack of automation and human-in-the-loop dependence, as seen extensively in many relevant industries [e.g., 16]. STM automated frameworks have been previously proposed and prototypes investigated [e.g., 17, 18, 19], but unifying industry interests and wide adoption remains a challenge. The relevance of maneuver sugges-

tions must address different mission requirements, long-time risk exposure, and corporate and cultural considerations, of each operator.

The future of the space industry depends on a practical space traffic management solution, which will require advancements in three key areas: (1) conjunction assessment uncertainty improvement, (2) practical and effective maneuver suggestion generation, and (3) automated data sharing and decision coordination. In this paper, further discussion of the space industry's need for STM is provided in Sec. 2, along with a partial description of a STM framework developed by Kayhan Space to address (2) and (3). The maneuver suggestion engine driving this framework is then further described and characterized in Sec. 3 and 4 respectively.

2. SPACE TRAFFIC AND ENVIRONMENT MANAGEMENT

STM has become an increasingly discussed area of need within the operational space community, especially in recent years with the growth of large constellations and debris-generating events in crowded orbital regimes. STM is only one part of a holistic orbital space safety plan. STM applies specifically to operational, maneuverable satellites, and can be thought of as the act of responsible collision avoidance coordinated at scale. From the perspective of the owner-operator and their day-to-day workflow, the term Space Traffic Coordination (STC) is perhaps more appropriate than STM.

Less frequently discussed than STM / STC, but equally important, is the concept of Space Environment Management (SEM) [20]. SEM consists of longer-term actions that promote debris mitigation (prevention of new debris) and debris remediation (cleaning up of existing debris).

Conjunction events that are between an operational satellite and a defunct RSO such as a debris fragment require conjunction risk assessment and risk mitigation maneuver planning. Most conjunctions dealt with by owner-operators currently fall into this category. However, conjunctions that are between two operational satellites not belonging to the same constellation require the two entities to coordinate maneuver responsibility - a process that so far has been painstakingly manual, fraught with potential for miscommunication, or simply non-existent.

Kayhan Space focuses its software-as-a-service (SaaS) Pathfinder platform on holistically addressing these three areas of operational space safety for satellite operators: conjunction risk assessment, collision avoidance planning, and STC. The focus of this paper is an in-depth technical discussion of Kayhan's cloud-based platform for computing optimal collision avoidance maneuvers. Kayhan is also presently unveiling the world's first machine-to-machine interface for autonomous STC, allowing satellite operators to pre-coordinate maneuver responsibility using industry-supported rules of the road. Further details on these new capabilities are available upon request.

2.1 STM industry and market trends

Other entities are both affected by and also help shape the growth of the STM market. Investors, fully cognizant of the increased threat posture of the *Great Power Competition*, have reviewed emerging space companies focused on the Space Situational Awareness (SSA), Space Domain Awareness (SDA), and STM segment of the value chain, as well as the more mature Tactical Space-based Intelligence, Surveillance, and Reconnaissance (TSISR). TSISR and SSA/SDA providers all have developed either the hardware into which the remote sensors are integrated, or have developed the software code designed to track, predict, and provide instructions for maneuverability for a diverse set of commercial, as well as national security applications and customers.

The data sets produced by these providers also have specific, targeted commercial applications for the financial services industry, insurance, and other asset-tracking business analytics providers. The macroeconomic basis and indicators to size the SSA, SDA, and STM marketplace is provided in [21].

A detailed analysis of USG customer bases, most critically the U.S. Space Force, indicates that LEO SDA data will yield constant demand over the next decade, rising from $157M in 2022 to $408M in 2032 (reflecting existing programs alone). Finally, an incredibly meaningful STM and SSA sub-market will grow from $12M in 2023 to $1.1B in 2032, as it accounts for 75% of the serviceable obtainable market for SSA data. This will be driven by commercial satellite operators, e.g., Kuiper, Starlink, OneWeb, plus all commercial Earth Observation and Position, Navigation, and Timing (PNT) providers.

These elements of viability for a SSA, SDA, and STM marketplace over the next ten years, combined with the specific focus and investment by the Departments of Defense and Commerce into fostering these capabilities for commercial

and national security application, makes clear that scaling SSA, SDA, and STM providers is a strong value creation opportunity.

3. AVOIDANCE MANEUVER SUGGESTION ENGINE

The main goal of collision avoidance maneuver (CAM) generation for conjunction risk mitigation is presenting the satellite operator with actionable solutions, i.e., solutions that fulfill the satellite's operational constraints and effectively balance risk tolerance and maneuver cost for the involved parties. This is usually cast as an optimization problem, where the decision variable(s), $\boldsymbol{x}$, affects the trajectory of one of the satellites (or both) in a way that some monitored risk/cost metrics are minimized, f, while others are constrained, g, to some desired range or target, l_i and u_i:

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & f(\boldsymbol{x}) \\ \text{s.t.} \quad & l_i \leq g_i(\boldsymbol{x}) \leq u_i \quad \forall_i. \end{aligned} \tag{1}$$

Many CAM advances have been made in recent years [22, and references therein]. Chan [23] summarized and compared publicly known avoidance maneuvers software systems with regard to model fidelity, computational efficiency, and accuracy. Of particular note is the work described by Bombardelli and Hernando-Ayuso [24], on the Polytechnical University of Madrid's comprehensive software tool "Optimal Computation of Collision Avoidance Maneuvers" (OCCAM), as the fastest and one of the most accurate systems analyzed. It features three optimization strategies for finding an avoidance impulse maneuver change of velocity vector, $\boldsymbol{\Delta v}$, as the decision variable, at a given maneuver epoch: maximum miss distance for a fixed $\boldsymbol{\Delta v}$ magnitude, minimum collision probability (P_c) for a fixed $\boldsymbol{\Delta v}$ magnitude, and minimum $\boldsymbol{\Delta v}$ magnitude while constraining P_c to be smaller than some threshold. These are formulated as quadratic programming by leveraging relative dynamics in the b-plane, linearized analytical dynamics through the use of a state transition matrix, Φ, and Chan's approach for P_c computation [25]. Gonzalo et al. [22] further developed this approach to use an linearized dynamics based on Gauss's planetary equations and linear relative motion instead of the Dromo orbital elements originally used.

Although computationally effective, the analytical solutions commonly described in the literature are tied to specific optimization problem formulations. They leverage, but also rely on, assumptions for the decision variable(s), objective, and constraint functions. Thus, these methods are unsuitable for optimization problem customization, each requiring a dedicated analytical approach and substantially increasing the development and maintenance of these solutions. Instead, the Kayhan Space CAM Suggestion Engine is set up to be as generic as possible, allowing for flexibility and customization to better support satellite operators. The engine strives to be an expandable compendium of plug-and-play avoidance algorithms, decision variables, and risk metrics that can serve as either objective or constraint functions. The top-level assumptions of this system only include:

- Conjunction information through a populated CCSDS conjunction data message (CDM) [26]. The CDM input provides nominal collision risk to be mitigated, as well as the spacecraft properties and states. The ability to regenerate the conjunction information when avoidance maneuvers are considered is also required.
- A CAM algorithm must take the CDM input and produce a suggested mitigation as a *maneuver plan*. These plans may contain maneuvers for each primary and/or secondary object, and the maneuvers can be of any type supported by the Kayhan Dynamics Library, including impulse and constant-thrust maneuvers.
- A CAM algorithm is directed by an optimization problem, which must be compatible with the formulation in Eq. 1.

The major strength of this system is that it allows for numerous customizations of an optimization problem at a relatively cheap development and maintenance cost. This is because most of the physics-domain modeling can be highly modular and polymorphic, enabling the use of NLP solvers with minimal extra inputs or parsing. Note that custom avoidance algorithms that use analytical solutions to specific optimization problems are still compatible with the above, and they could still be leveraged for performance, albeit with a higher development cost.

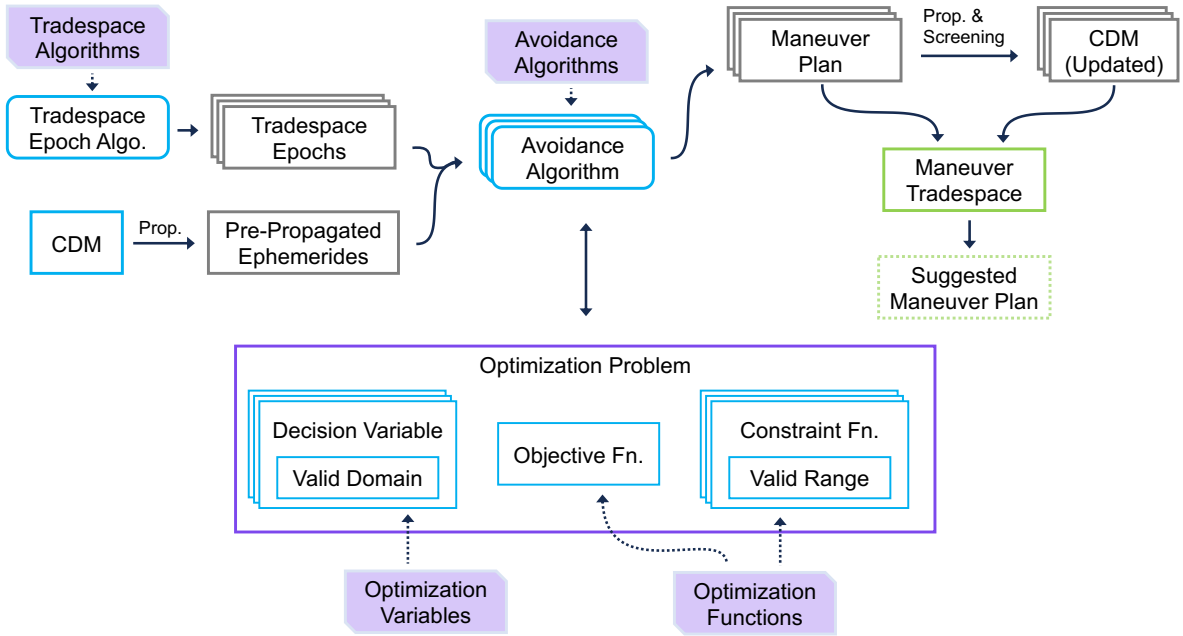


Fig. 1: Kayhan Space CAM engine diagram. Blue shapes indicate engine inputs/selections. Purple-shaded shapes indicate a collection of possible options supported, with their blue-shaped counterparts being a realization of those supported types.

3.1 Architecture

The Kayhan Space CAM engine architecture is illustrated in Fig. 1. It is built to be modular and efficient, following object-oriented paradigms. It expects four main external inputs: a CDM defining the collision risk to be mitigated, an algorithm to generate tradespace epochs (dependent variable), an optimization problem to guide the mitigation strategy, and an avoidance algorithm to generate mitigation *maneuver plan*s given the previously listed inputs. The *maneuver plan*s are validated through high-fidelity propagation of the objects, taking each maneuver plan into account around the original closest approach time, t_{ca}. With the updated object trajectories, conjunction screening is performed to generate updated CDMs corresponding to the result of each *maneuver plan*, and they comprise the output maneuver trade space. Automation of trade space selection and even iteration of other inputs is beyond the scope of this paper, although they are also part of Kayhan's planned STM framework.

The high-fidelity propagation and screenings are performed using Kayhan Space's own internal dynamics library. Several methods of collision probability are available, including those described in [27], referred to as Foster's 2D P_c, and [28], reffered to as Hall's 3D P_c. Screening for secondary and tertiary conjunctions for the maneuvered object is performed against an RSO catalog in the Kayhan Space STM framework, and it's also beyond the scope of this paper. Currently, the CAM engine does not take into account secondary/tertiary conjunctions directly in the optimization of $\boldsymbol{\Delta v}$.

The output tradespace *maneuver plan*s are each populated by an independent execution of an avoidance maneuver algorithm at a specific algorithm epoch t_{alg}. These epoch times are dictated by the tradespace epoch algorithm, which outputs a list of times t_{alg} prior to t_{ca}. The list of t_{alg} generated by the tradespace algorithm is also used during the pre-propagation of the reference ephemerides for the primary and secondary objects ahead of the maneuver optimization, efficiently providing the high-fidelity state and state transition matrix, Φ, for the object between the earliest t_{alg} and t_{ca}. Three tradespace algorithms have been implemented: a uniform grid, an antipodal grid that computes the times when the primary satellite is at the antipode point relative to the conjunction in its orbit, and a custom grid. The antipodal grid search algorithm is useful for finding the most effective maneuver times, as numerical analyses suggest that tangential maneuvers at antipodal locations tend to result in global minima for common optimization problems.

Note that the algorithm epoch t_{alg}, the reference epoch for the avoidance algorithm, is decoupled from the maneuver times, t_{cam}, generated by the algorithm, i.e., the algorithm is allowed to generate maneuvers outside the algorithm epoch, t_{alg}, as required by the mitigation strategy chosen. For the analyses in this paper, all generated maneuvers are impulsive with $t_{\text{cam}} = t_{\text{alg}}$.

The next two sections explain the optimization problem and avoidance algorithm concepts and their implementations.

3.2 Optimization Problem

Following the design philosophy of the engine, the optimization problem allows for a modular plug-and-play configuration. It is defined by a list of decision variables, an objective function, and a list of constraint functions, illustrated in Fig. 1 and following the generic formulation in Eq. 1. Note that within the context of this section, *optimization variables* and *optimization functions* refer interchangeably to their implementation as an object, as defined in object-oriented programming.

The decision variable selection dictates the solution space dimensionality for the optimization problem. The domain of the decision variables must also be defined, bounding the solution space. Currently, only two decision variables are supported: an impulse maneuver's ignition epoch and delta-V vector. Only, the impulsive maneuver delta-V vector decision variable is characterized in this paper. Support for additional variables, e.g., variables used to define a non-impulsive maneuver, is relatively easy by virtue of the modular programming paradigms used, i.e., it can be concisely implemented without any architectural changes. An initial guess can also be given to initialize the decision variable for some classes of optimization problem solvers, which could be computed by leveraging the analytical solution or empirical knowledge for specific problems.

The objective and constraint functions can be populated by a common list of optimization functions described in the next section. Each function is required to, at a minimum, support its evaluation at a given realization of the decision variable list, returning a single float value, i.e. the output has a single dimension.

The objective function is usually treated as a cost function by the avoidance algorithms and minimized. Objective functions can also be flipped by having their evaluations multiplied by -1, i.e., so their minima become maxima and vice-versa. Only a single objective function is currently supported, and future expansion to multi-objective support is planned.

Constraint functions are set up by also providing a valid range, i.e., a minimum and maximum value, for which a realization of the decision variable will be considered valid if the constraint is within the desired range.

3.2.1 Optimization Functions

As described in the previous section, the optimization functions (programming construct) that can be realized as objective or constraint functions for an optimization problem must, at a minimum, implement the evaluation of that function at a given decision variable coordinate of the optimization problem domain. Additionally, implementation of first and second-order partials, i.e., the gradient and hessian, may also be implemented and can be leveraged by the avoidance algorithms being used if supported or required.

Because propagation is usually the most expensive computation in avoidance maneuver modeling, the computation cost of an optimization function evaluation is usually tied to the epoch to which the risk metric corresponds and the earlier maneuver performed. As such, metrics related to maneuver properties are usually cheap to compute, whereas conjunction metrics corresponding to state changes at t_{ca} are more expensive. Lower fidelity propagation or surrogate models may be used where appropriate to improve performance. Linearized dynamics from the pre-propagation of the object's trajectory, described in Sec. 3.1, are computationally free during optimization function evaluations. Additionally, because t_{ca} is the time when

$$\left[\boldsymbol{r}(t)_{\text{primary}} - \boldsymbol{r}(t)_{\text{secondary}}\right] \cdot \left[\boldsymbol{v}(t)_{\text{primary}} - \boldsymbol{v}(t)_{\text{secondary}}\right] = 0, \tag{2}$$

t_{ca} is sensitive to trajectory changes. Thus, risk metrics that depend on t_{ca} require additional screening to update the original t_{ca} using updated trajectories. Alternatively, first-order approximation of t_{ca} shift can be computed by linearizing Eq. 2, or risk metrics can be evaluated at the original t_{ca} instead, if accuracy loss is acceptable. The accuracy and performance of these model simplifications are investigated in the Sec. 4.1.

Table 1 presents the list of supported optimization functions and their features. The second and third columns indicate whether first and second-order derivatives with respect to the relevant decision variables are supported or not; the fourth column indicates whether the metric is computed at the original or new t_{ca}. The last column shows the propagation model used to compute the metric. Each of the functions is described in the following subsections.

Table 1: Optimization functions considered in this work and their feature list.

Metric	Gradient	Hessian	Orig./New t_{ca}	Prop. Model
Delta-V Magnitude Squared	Yes	Yes	N/A	N/A
Delta-V Axis Projection (Squared)	Yes	Yes	N/A	N/A
Delta-V Azimuth	Yes	Yes	N/A	N/A
Delta-V Elevation	Yes	Yes	N/A	N/A
Delta-V Magnitude Squared Combination	Yes	Yes	N/A	N/A
Miss Distance (Numeric)	No	No	Both Supported	Linearized and Full Dynamics
Miss Distance (Analytic)	Yes	Yes	Orig. t_{ca} Only	Linearized Only
Probability of Collision (Numeric)	No	No	Both Supported	Linearized and Full Dynamics
Chan's P_c Depth of Intrusion	Yes	Yes	Orig. t_{ca} Only	Linearized Only

3.2.1.1 Delta-V Magnitude Squared, $\|\boldsymbol{\Delta v}\|^2$ The delta-V magnitude squared is used as an alternative to computing the magnitude of the maneuver $\boldsymbol{\Delta v}$ and as a proxy for maneuver fuel usage since their extrema overlap and the square is cheaper to compute. It is given by

$$f(\boldsymbol{\Delta v}) = \boldsymbol{\Delta v} \cdot \boldsymbol{\Delta v}, \tag{3}$$

and the gradient and hessian can be easily derived for both an impulse and constant thrust maneuver. The metric is associated with a particular maneuver decision variable, so for example, several $\|\boldsymbol{\Delta v}\|^2$ constraints can be associated with multiple maneuvers of the optimization problem for either primary or secondary objects.

3.2.1.2 Delta-V Axis Projection (Squared), $\boldsymbol{\Delta \hat{v}} \cdot \boldsymbol{\hat{u}}$ The projection of the delta-V vector onto a user-defined direction can be used to align a maneuver closest to a given direction or set up a cone constraint. The square of the projection can also be used, which is useful in collapsing the projection sign and allowing for a mirrored projection, e.g., a posi/retro-grade in-track maneuver. It is given by

$$f(\boldsymbol{\Delta v}) = \boldsymbol{\Delta \hat{v}} \cdot \boldsymbol{\hat{u}}, \tag{4}$$

where $\boldsymbol{\hat{a}}$ denotes a unit vector of $\boldsymbol{a}$, and the user-defined direction, $\boldsymbol{\hat{u}}$, is given as a pair of elevation and azimuth angles in the Radial In-track Cross-track (RIC) frame (See Fig. 2). When set as a constraint, range bounds can be inputted as an angle measure instead of its cosine for greater convenience in setting up a cone constraint. Again, the gradient and hessian can be easily derived, and the metric can be set to track any of the maneuvers in the optimization problem.

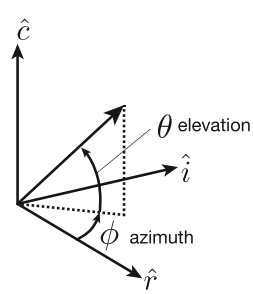


Fig. 2: Diagram of azimuth and elevation angles describing a vector in the RIC frame.

3.2.1.3 Delta-V Azimuth $\mathrm{az}_{\boldsymbol{\Delta v}}$ and Elevation $\mathrm{el}_{\boldsymbol{\Delta v}}$ These two independent functions return the azimuth and elevation angles of a maneuver delta-V in the RIC local frame (See Fig. 2). The azimuth function is given by

$$f(\boldsymbol{\Delta v}) = \arctan\left(\frac{\boldsymbol{\Delta v}\cdot\hat{\boldsymbol{i}}}{\boldsymbol{\Delta v}\cdot\hat{\boldsymbol{r}}}\right), \tag{5}$$

where $\hat{\boldsymbol{r}}$ and $\hat{\boldsymbol{i}}$ are the radial and in-track unit vectors. The elevation is given by

$$f(\boldsymbol{\Delta v}) = \arctan\left(\frac{\boldsymbol{\Delta v}\cdot\hat{\boldsymbol{c}}}{|\boldsymbol{M_c}\cdot\boldsymbol{\Delta v}|}\right), \tag{6}$$

where the $\boldsymbol{M_c}$ operator projects a vector onto the cross-track plane, i.e., $\boldsymbol{M_c} = -\tilde{\hat{\boldsymbol{c}}}\cdot\tilde{\hat{\boldsymbol{c}}}$, and $\tilde{\hat{\boldsymbol{c}}}$ is the skew-symmetric matrix for cross-product operation ($\hat{\boldsymbol{c}}\times\boldsymbol{a} = \tilde{\hat{\boldsymbol{c}}}\cdot\boldsymbol{a}$).

Note that when azimuth is set as a constraint, the original constraint bounds are internally transformed, and $f(\boldsymbol{\Delta v})$ is adjusted to fit completely within the unit circle to avoid wrap-around discontinuities. This allows the original constraints to describe an interval outside the nominal $-\pi < \mathrm{az}_{\boldsymbol{\Delta v}} < \pi$ domain, e.g., $3/4\pi < \mathrm{az}_{\boldsymbol{\Delta v}} < 5/4\pi$ is supported. The range of these two function is thus: $\mathrm{az}_{\boldsymbol{\Delta v}} \in \mathbb{R}$ and $-\pi/2 < \mathrm{el}_{\boldsymbol{\Delta v}} < \pi/2$ in rad. The gradient and hessian can be easily derived, and the metrics can be set to track any of the maneuvers in the optimization problem.

3.2.1.4 Delta-V Magnitude Squared Combination, $\|\boldsymbol{\Delta v}\|^2$ Delta-V magnitude squared of two maneuvers in the optimization problem, from either spacecraft, are combined together by either a sum or subtraction depending on user choice. It is useful in constraining a shared-burden optimization problem, where the maneuver delta-V burden is shared between primary and secondary objects. It is given by

$$f(\boldsymbol{\Delta v}) = \boldsymbol{\Delta v}_{\mathrm{a}}\cdot\boldsymbol{\Delta v}_{\mathrm{a}} \pm \boldsymbol{\Delta v}_{\mathrm{b}}\cdot\boldsymbol{\Delta v}_{\mathrm{b}}, \tag{7}$$

and the gradient and hessian is a trivial derivation from Sec 3.2.1.1. This function could easily be modified to a weighted difference scaled by total delta-V, where a user-defined maneuver cost can be input for each maneuvering spacecraft, enabling greater flexibility for a cost-shared coordinated avoidance maneuver.

3.2.1.5 Miss Distance (Numeric), $\|\boldsymbol{r}\|$ This miss distance metric is the Euclidean distance between the two conjuncting objects at t_{ca}, and requires knowledge of the object's relative states at t_{ca}. This is usually computed through the screening process described in Sec. 3.1. In the simplest approach, computing miss distance for a given maneuver plan requires propagating the maneuvering spacecraft forward from the start of the maneuver to a window of time around the original CDM's t_{ca} and screening for a conjunction within that time. The propagation and, to a lesser extent, screening are expensive computations, but this optimization function also offers cheaper alternatives to the propagation and screening.

As already discussed in Sec. 3.1, states and state transition matrix, Φ, are computed for a window covering all algorithm epochs. So expensive propagation can be avoided by employing linearized dynamics to compute state changes at the original t_{ca} due to an earlier maneuver.

For avoidance maneuvers with a relatively small impact on relative position, the miss distance at the original t_{ca}, with updated states for objects that maneuvered, may be used as a reasonable approximation for the miss distance at the updated t_{ca}. This is particularly useful when the metric is used as an objective function instead of as a narrow constraint target, since extrema locations are similar albeit at different depths.

Additionally, a closed-form expression of the gradient and hessian for this metric is non-trivial to compute when propagation and screening are used since they both involve complex mathematical operations. The first-order approximation of t_{ca} shift can be computed by linearizing Eq. 2 and using linearized dynamics, which makes it possible to compute the gradient of the miss distance. However, the analytical computation of the Hessian requires the inclusion of the second-order dynamical effects, which are not easily accessible. Therefore, this metric is not supported by optimization problem solvers that require direct first- and second-order partials, though they could still be numerically computed, e.g., through finite differencing. When the dynamics are linearized and higher-order effects are ignored, an analytical formulation is possible, as discussed in the next function.

3.2.1.6 Miss Distance (Analytic), $\|\boldsymbol{r}\|$ By leveraging the state transition matrix, Φ, from a single propagation of the primary with no avoidance maneuver to the desired maneuver time, a relationship between maneuver $\boldsymbol{\Delta v}$ and change in relative position can be found

$$\boldsymbol{r} = \boldsymbol{r}_0 + \boldsymbol{T} \cdot \boldsymbol{\Delta v}, \tag{8}$$

where $\boldsymbol{r}_0$ is the original relative position, and $\boldsymbol{r}$ is the new relative position. T can include the first-order effect of t_{ca} shift, but in the simplest case, $\boldsymbol{T}$ is the 3×3 upper-right corner of the $\Phi(t_{\text{ca}}, t_{\text{cam}})$ with t_{ca} being the original TCA. This simplification is employed throughout the paper unless otherwise mentioned.

For ease of implementation, the system uses the miss distance squared, namely $\|\boldsymbol{r}\|^2 = \boldsymbol{r} \cdot \boldsymbol{r}$. This metric is a special case of a more generic quadratic form $f(\boldsymbol{\Delta v}) = \boldsymbol{r} \cdot \boldsymbol{Q} \cdot \boldsymbol{r}$. Given the linear mapping of the maneuver to the change in the relative position at t_{ca} in Eq. 8, the general quadratic form becomes

$$\begin{aligned} f(\boldsymbol{\Delta v}) = \boldsymbol{r} \cdot \boldsymbol{Q} \cdot \boldsymbol{r} &= (\boldsymbol{r}_0 + \boldsymbol{T} \cdot \boldsymbol{\Delta v}) \cdot \boldsymbol{Q} \cdot (\boldsymbol{r}_0 + \boldsymbol{T} \cdot \boldsymbol{\Delta v}) \\ &= \boldsymbol{\Delta v} \cdot \underbrace{\boldsymbol{T}^T \cdot \boldsymbol{Q} \cdot \boldsymbol{T}}_{\boldsymbol{A}} \cdot \boldsymbol{\Delta v} + 2 \underbrace{\boldsymbol{r}_0 \cdot \boldsymbol{Q} \cdot \boldsymbol{T}}_{\boldsymbol{c}} \cdot \boldsymbol{\Delta v} + \underbrace{\boldsymbol{r}_0 \cdot \boldsymbol{Q} \cdot \boldsymbol{r}_0}_{d} \\ &= \boldsymbol{\Delta v} \cdot \boldsymbol{A} \cdot \boldsymbol{\Delta v} + 2\boldsymbol{c} \cdot \boldsymbol{\Delta v} + d. \end{aligned} \tag{9}$$

This quadratic form can be used to construct a special class of optimization problems such as quadratically constrained quadratic programming (QCQP) where analytical or more efficient solution methods are available as demonstrated in [22, 24]. The miss distance squared can be computed from Eq. 9 with $\boldsymbol{Q} = \boldsymbol{I}_3$.

3.2.1.7 Probability of Collision (Numeric), P_c The probability of collision measure depends directly on the conjunction, i.e., closest approach, between the two objects, specifically on their relative position, uncertainties, and spacecraft shape. There are different approaches to computing P_c with varying assumptions in the literature.

The widely accepted measure of P_c is given by Foster's formulation [27], also known as 2D P_c. This work uses the same implementation as the publicly available NASA CARA Analysis Tool [29]. Similar to the discussion in the numeric miss distance in section 3.2.1.5, the computation of its gradient and Hessian is non-trivial if we assume full nonlinear dynamics. The 2D P_c itself involves a complex and nonlinear integral over the hard body radius of the encounter plane. In addition, since any avoidance maneuver will cause a change to t_{ca}, P_c would have to be computed with respect to the new t_{ca}.

While the global search algorithm (Sec. 3.3.2) can be used to optimize Foster's P_c directly without access to its partial derivatives, the faster optimal algorithm (Sec. 3.3.4) cannot. Instead, Chan's formulation for computing P_c [25] can provide a useful approximation for optimization through the use of its intermediate depth of intrusion parameter, v, and it's described next in Sec. 3.2.1.8.

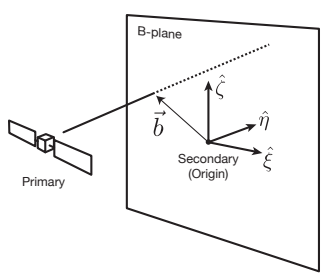


Fig. 3: Schematic of B-plane coordinate frame.

3.2.1.8 Depth of Intrusion, v In Chan's method, the 2D integral of a circular area with an elliptic Gaussian is converted to a 2D integral of an elliptic area with a circular Gaussian as discussed in Chap. 4 and 5 in [25]. As a result, P_c can be given as the following Rician distribution

$$P_c(u,v) = e^{-v/2} \sum_{m=0}^{\infty} \frac{v^m}{2^m m!} \left(1 - e^{-u/2} \sum_{k=0}^{m} \frac{u^k}{2^k k!} \right), \tag{10}$$

where

$$u = \frac{r_A^2}{\sigma_\xi \sigma_\zeta \sqrt{1-\rho^2}}, \tag{11}$$

$$v = \left[\left(\frac{\xi}{\sigma_\xi}\right)^2 + \left(\frac{\zeta}{\sigma_\zeta}\right)^2 - 2\rho \frac{\xi}{\sigma_\xi}\frac{\zeta}{\sigma_\zeta} \right] / (1-\rho^2). \tag{12}$$

The parameter u depends only on the covariance and hard body radius, whereas the depth of intrusion v is a function of the relative position in the B-plane. The B-plane coordinate frame is shown in Fig. 3, and the same definition as [22] is used; η-axis is in the relative velocity direction; ξ-axis is in $\boldsymbol{v}_{\text{secondary}} \times \hat{\boldsymbol{\eta}}$ direction; ζ-axis is in $\hat{\boldsymbol{\xi}} \times \hat{\boldsymbol{\eta}}$ direction. The parameters u and v are defined in terms of the combined covariance at t_{ca}, i.e., $\boldsymbol{C}_{\text{prim}} + \boldsymbol{C}_{\text{sec}}$, expressed in the B-plane coordinate frame and marginalized along η-axis as per the assumption of 2D P_c:

$$^{B}\boldsymbol{C}_{\text{marginal}} = \begin{bmatrix} \sigma_\xi^2 & \rho\sigma_\xi\sigma_\zeta \\ \rho\sigma_\xi\sigma_\zeta & \sigma_\zeta^2 \end{bmatrix}. \tag{13}$$

$-1 < \rho < 1$ is the correlation coefficient, and r_A is the hard body radius (radius of the combined cross-section of the primary and secondary).

By definition, $u > 0$. Thus, in Eq. (10) the second term in the parenthesis is bounded

$$0 < e^{-u/2} \sum_{k=0}^{m} \frac{u^k}{2^k k!} < e^{-u/2} \sum_{k=0}^{\infty} \frac{u^k}{2^k k!} = 1. \tag{14}$$

Therefore,

$$0 < \left(1 - e^{-u/2} \sum_{k=0}^{m} \frac{u^k}{2^k k!}\right) < 1, \tag{15}$$

where the above converges to 0 as $m \to \infty$. This result, along with the fact that $v > 0$, means that there is some constant $0 < \alpha < 1$ such that

$$\sum_{m=0}^{\infty} \frac{v^m}{2^m m!} \left(1 - e^{-u/2} \sum_{k=0}^{m} \frac{u^k}{2^k k!}\right) < \sum_{m=0}^{\infty} \frac{(\alpha v)^m}{2^m m!} = e^{\alpha v/2}. \tag{16}$$

Thus an inequality for Chan's P_c can be derived

$$P_c(u,v) < e^{-(1-\alpha)v/2}, \tag{17}$$

demonstrating that *minimizing P_c is equivalent to maximizing the depth of intrusion v.*

Note that, more generally, v can be rewritten in terms of the covariance $\boldsymbol{C}$ of the 3D joint distributions of the relative position,

$$v = \boldsymbol{r} \cdot \boldsymbol{M}_\eta \cdot \boldsymbol{C}^{-1} \cdot \boldsymbol{M}_\eta \cdot \boldsymbol{r}, \tag{18}$$

where $\boldsymbol{M}_\eta = -\tilde{\hat{\boldsymbol{\eta}}} \cdot \tilde{\hat{\boldsymbol{\eta}}}$, and $\boldsymbol{r}$, $\hat{\boldsymbol{\eta}}$, and $\boldsymbol{C}$ are in the same frame. Eq. 18 shows that v is the squared Mahalanobis distance of the marginalized 2D Gaussian distribution on the B-plane. Assuming the linear mapping in Eq. 8, Eq. 18 is equivalent to the quadratic form in Eq. 9 with

$$\boldsymbol{Q} = \boldsymbol{M}_\eta \cdot \boldsymbol{C}^{-1} \cdot \boldsymbol{M}_\eta \tag{19}$$

Thus, Chan's 2D probability of collision can be used to define a quadratic programming problem.

Given, u and v, Chan's P_c can be computed following Eq. (10) using the recursive formulation in [25]. The zeroth-order approximation, $m = 0$, which might be appropriate for small objects, simplifies to

$$P_c \simeq e^{-v/2}(1 - e^{-u/2}). \tag{20}$$

By inverting this relationship, an approximation of the depth of intrusion, v, that corresponds to a given P_c can be found

$$v = 2\ln\left(1 - e^{-u/2}/P_c\right). \tag{21}$$

This equation is used to convert P_c bounds given to the avoidance maneuver engine into the depth of intrusion, v, bounds.

3.3 Avoidance Algorithms

The avoidance algorithms that sit at the core of the Kayhan Space CAM engine are responsible for outputting a suitable avoidance maneuver plan given an optimization problem setup. Currently supported avoidance algorithms are described later in this section.

The only common requirement across all avoidance algorithms is that optimization problems are populated with objective functions and constraint functions that can be evaluated at a given decision variable realization. However, each avoidance algorithm is allowed to have further requirements on an optimization problem setup, i.e., avoidance algorithms are not required to be compatible with all optimization problem configurations. As such, some optimization problems, variables, or functions may be implemented specifically to support certain avoidance algorithms. For example, a family of avoidance algorithms may require that optimization function objects can also return the function's gradient at evaluation.

3.3.1 Fixed Maneuver Plan

The simplest avoidance algorithm simply returns a user-defined *maneuver plan*, regardless of the optimization problem. A *maneuver plan* template can also be provided, and dynamically modified to use the algorithm epoch, t_{alg}, as the maneuver time. Paired maneuvers are supported in this fashion, where the second maneuver in the pair is set to the next t_{alg} in the tradespace. This is the cheapest algorithm to compute, and in many geometric scenarios, it can be tuned to return the optimal solution for certain optimization problems, e.g., at conjunction antipodes, miss distance is the most sensitive to maneuvers in the in-track direction for common conjunction geometries.

3.3.2 Global Grid Search

The global grid search algorithm evaluates an optimization problem's objective and constraint functions over the whole domain of the optimization decision variables, discretized to a uniform grid. The grid size is an algorithm input, and the minimum inside the solution space is guaranteed to be global over the discretized space, assuming appropriate choice of grid size to match the smoothness of the optimization functions. For each grid point, the constraint function is first evaluated, followed by the objective function, but only if the constraint is met. The gradient and hessian for the metrics are not needed, but they can also be computed if desired. This algorithm suffers from the "curse of dimensionality" with respect to the number of solution space dimensions, i.e., the number of decision variables and the grid resolution. However, since the evaluations are independent, the problem is "embarrassingly parallel", and parallelization options are part of the algorithm inputs. This is the most flexible algorithm, capable not only of finding the globally optimal solution to an optimization problem but also of illustrating the evaluated metrics and their gradients in the solution space for research (e.g. Figs 7, 8 and 9 in Sec. 4)

3.3.3 Gradient descent with Nesterov Momentum

A more efficient traversal of the solution than the global grid search, gradient descent uses the objective function's gradient to indicate the local direction of a minimum. The gradient used can be an approximation, thus not imposing any requirement on the optimization functions used in the optimization problem. As with any gradient descent, only local minimization is guaranteed, which is sensitive to the initial guess. The gradient descent is extended with Nesterov momentum [30] to improve the convergence rate and decrease convergence to shallower local minima. Currently, this algorithm is only partially implemented, and only minimization of an in-track impulse maneuver delta-V with a single constraint risk metric, e.g., target P_c, using a gradient approximation, is supported.

3.3.4 Interior Point Optimazation

Another gradient-based method and more efficient traversal of the solution space, this algorithm wraps around IPOPT developed by COIN-OR [31] for solving NLP problems. The NLP solver requires the gradient and Hessian of the objective and constraint functions. Similar to the gradient descent, only local minimization is guaranteed, which can be sensitive to the initial guess.

3.3.5 Ant Colony Optimization (ACO)

This algorithm wraps around pagmo's implementation of Ant Colony Optimization (ACO) [32]. ACO is a metaheuristic global optimization algorithm, categorized as a stochastic, gradient-free (i.e., zeroth order), and population-based evolutionary algorithm. The algorithm is inspired by ants' behavior to find the shortest path to food sources from their nest via indirect communication with their pheromones [33]. Pagmo's implementation is mainly based on the extended ACO, which is designed to solve non-convex mixed-integer nonlinear optimization problems [34]. Out of many metaheuristics offered by pagmo, ACO was chosen due to its ability to handle constrained optimization problems without gradient information, and potentially support future expansion to mixed-integer domain problems. Other pagmo algorithms could also be leveraged in the future. As a new addition to the Kayhan CAM engine, this paper does not include any characterization of ACO as an avoidance algorithm.

4. AVOIDANCE MANEUVER SUGGESTION ENGINE PERFORMANCE

The performance, for both optimality and computational efficiency, of the maneuver suggestion engine is presented in this section for a handful of optimization problem and avoidance algorithm combinations, followed by some discussion on the results. A collection of seven randomly picked low-earth orbit (LEO) CDMs are used as the collision risk input to be mitigated by the engine. These CDMs describe conjunctions between about 400 to 600 km, with miss distances ranging from 20 to 20 000 m and P_c from 1.5×10^{-2} to 5.5×10^{-5}. An impulsive maneuver delta-V for the primary spacecraft is setup as the sole decision variable, with the exception of the shared-burden case where an impulsive maneuver for the secondary spacecraft is also considered. The requested tradespace epochs, which equal the impulse maneuver epoch, is composed of five distinct times between roughly 15 to 12 hours before t_{ca}, aligned to start at a conjunction antipode.

The summary of the performance results for several runs of the avoidance maneuver engine are presented in Table 2. Six categories of mitigation strategies are investigated:

1. A simple fixed 1 cm/s maneuver, either posigrade or retrograde based on sign of the relative state radial separation (pulled from a CDM), i.e., retrograde if primary is lower at t_{ca};
2. Maximization of miss distance with a delta-V magnitude constraint (Sec. 3.2.1.1) of less than or equal to 1 cm/s;
3. Minimization of probability of collision with a delta-V magnitude constraint of less than or equal to 1 cm/s;
4. Same as (3) but with the added delta-V projection constraint (Sec. 3.2.1.2) set to only allow maneuvers along the in-track axis (posi- or retro-grade);
5. Minimization of probability of collision, with primary and secondary impulsive maneuver delta-V set as a decision variables and constrained to have equal magnitude and combined sum less than or equal to 1 cm/s (Sec. 3.2.1.4);
6. Minimization of delta-V magnitude with P_c constrained to be less than or equal to 1×10^{-6}.

Each category holds highlighted combinations of avoidance algorithms and optimization problems, including:

- Fixed maneuver (Sec. 3.3.1), global grid search (Sec. 3.3.2), IPOPT (Sec. 3.3.4), and Nesterov gradient descent (Sec. 3.3.3) algorithms,
- Numeric miss distance (Sec. 3.2.1.5) and probability of collision (Sec. 3.2.1.7), with non-linear vs. linear dynamics at original vs. update t_{ca}, analytic miss distance (Sec. 3.2.1.6), and Chan's depth of intrusion (Sec. 3.2.1.8).

For global grid searches, a cell linear size of 0.2 cm/s is used for total of 1331 points in the ± 1 cm/s domain.

The computation time, delta-V magnitude, miss distance (numeric), and probability of collision (numeric) for each tradespace epoch of every CDM for a given row in Table 2 (an avoidance algorithm and optimization problem combination) are compared to the corresponding results from the fixed maneuver strategy at each tradespace epoch. The

results are averaged across all CDM tradespaces (five epochs per CDM, for seven CDMs, yielding a total of 35 *maneuver plans*) and displayed on the main columns of the table. The average $\log_{10} P_c$ and overall convergence failure rate are also displayed. Note that the miss distance and probability of collision used for this comparison are the validated metrics from high-fidelity propagation and screening of each *maneuver plan* as described in Sec. 3.1, independent of optimization problem used. Fig. 4 showcases an example tradespace for one of the studied CDMs, with an extended number of tradespace epochs and a limited number of the mitigation strategies in Table 2.

Table 2: Summary of the performance of several *maneuver plans* output by the engine, for different mitigation strategies and combination of avoidance algorithm and optimization problem. The columns shows the average improvement (blue) or worsening (red) of a particular metric with respect to the corresponding fixed maneuver algorithm, for any given *maneuver plan* across all CDM tradespaces investigated.

Algorithm (Opt. Problem Notes)	Average Across CDM Tradespaces					Failure Rate
	Δ Elapsed Time	Δ Delta-V Mag.	Δ Miss Distance	Δ 2D P_c	$\log_{10} P_c$	
1) **Fixed Posi-/Retro-Grade Impulsive Maneuver for Primary Spacecraft**						
Fixed Imp. Man.	0%	0%	0%	0%	-4.4	0%
2) **Maximize Miss Distance** with Delta-V Magnitude Constraint						
Grid Search (^{F,N})	8071%	0%	5%	9%	-4.4	0%
Grid Search (^{L,N})	173%	0%	0%	0%	-4.4	0%
Grid Search (^{L,O})	136%	0%	0%	0%	-4.4	0%
Grid Search (Analytic)	5%	0%	0%	0%	-4.4	0%
IPOPT (Analytic)	4%	0%	0%	0%	-4.4	0%
3) **Minimize** P_c with Delta-V Magnitude Constraint						
Grid Search (^{L,N})	168%	0%	-12%	-21%	-4.7	0%
Grid Search (^{L,O})	138%	-1%	-27%	21%	-4.7	0%
Grid Search (Chan's v)	7%	0%	-11%	-22%	-4.7	0%
IPOPT (Chan's v)	1%	0%	-10%	-23%	-4.7	0%
4) **Minimize** P_c with Delta-V Magnitude and In-track Direction Constraints						
Grid Search (^{L,N})	9%	-2%	-10%	-17%	-4.7	0%
Grid Search (^{L,O})	8%	-10%	-22%	19%	-4.7	0%
Grid Search (Chan's v)	12%	0%	-9%	-19%	-4.7	0%
IPOPT (Chan's v)	4%	0%	-9%	-19%	-4.7	0%
5) **Minimize** P_c with Delta-V Magnitude and Shared-Burden Constraints						
IPOPT (Chan's v)	23%	0%	125%	-8%	-4.5	0%
6) **Minimize Delta-V Magnitude** with $P_c \leq 1 \times 10^{-6}$ Constraint						
Grid Search (^{L,N})	448%	541%	95%	-86%	-6.1	14%
Grid Search (^{L,O})	351%	228%	41%	-49%	-5.2	6%
IPOPT (Chan's v)	0%	730%	103%	-87%	-5.8	0%
Nesterov (^{L,N})	-5%	525%	224%	-99%	-8.0	60%
Nesterov (Chan's v)	4%	525%	208%	-98%	-8.0	57%

^{F} Full dynamics, ^{L} Linearized dynamics

^{N} New t_{ca}, ^{O} Original t_{ca}

Overall, the first three categories show unsurprising results regarding optimality. Miss distance maximization (2) aligns well with fixed posi-/retro-grade in-track maneuvers (1). P_c minimization solutions (3) achieve lower P_c by diverging from (1) and (2), with possible maneuver off the in-track axis, at certain tradespace epochs. When the maneuver is

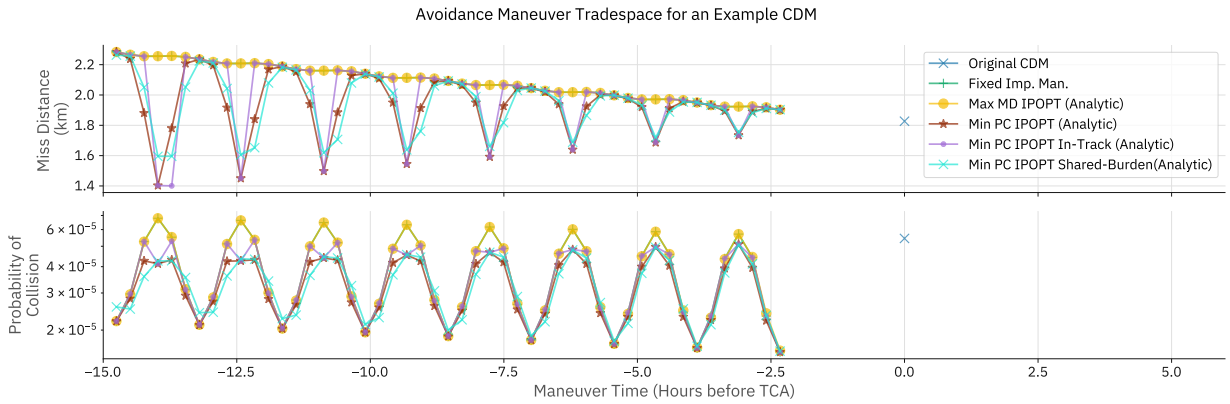


Fig. 4: Sample tradespace of avoidance maneuver plans for one of the tested CDMs. The miss distance (top) P_c (bottom) metrics shown are from high-fidelity propagation and screening of each *maneuver plan*, independent of optimization problem used.

constrained to lie on the in-track axis (4), the P_c minimization does not achieve results as low as (3), but they are still better than (1) and (2), since the opposite in-track direction sometimes yields an increase in miss distance but a decrease in P_c. Similar behavior is observed with the shared-burden minimization of P_c (5), where P_c minimization achieved is better than in (1) but not as well as in (3), due to the constraining geometry of two maneuvering spacecraft.

The linearized dynamics provide an acceptable approximation of the full dynamics for this data set, as can be observed in the global grid search for miss distance maximization. This was the only category to include a full dynamics run due to its computational cost. In contrast, original versus new t_{ca} modeling proved to be a less acceptable model approximation. There are no obvious issues with the global grid search cell size chosen and the smoothness of the optimization problems, yielding a good balance between performance and discretization error. Chan's depth of intrusion shows higher performance over numerical P_c. These modeling approximations are further explored below in Sec. 4.1.

The P_c targeting at minimum delta-V magnitude (6) demonstrated large increases in delta-V since the desired target P_c usually required more than the 1 cm/s of previous categories. Global grid search at updated t_{ca} and IPOPT using Chan's depth of intrusion both performed reasonably well in this category, averaging close to the target P_c, though the grid search had a higher convergence failure due to stricter convergence criteria. The Nesterov gradient descent did not perform as well, with higher convergence failure and overshooting of the target P_c, but as already described in Sec. 3.3.3, the algorithm is not yet fully implemented. The gradient approximation in this case is binary, and a poor substitute for the optimization function's gradient.

With regards to performance, avoidance algorithm run times for optimization problems using analytic optimization functions show negligible difference when compared to the trivial fixed maneuver algorithm, i.e., the final validation of the *maneuver plan*, propagation and screening, dominate the computational cost in these cases. One row, for Nesterov (^{L,N}), shows a performance increase because of early convergence failure bypassing *maneuver plan* validation, e.g., from an empty maneuver plan. The global grid search for the shared-burden case, even if only using the cheaper Chan's depth of intrusion, would have greatly suffered from the "Curse of dimensionality" already discussed, since the decision variable array grows from three to six dimensions squaring the solution space.

4.1 Modeling Approximations: Linearized Dynamics, t_{ca} Update, and Chan's Depth of Intrusion

The modeling approximation discrepancies presented in Table 2 are further explored in this section.

The use of linearized dynamics as a cheaper surrogate model for the full spacecraft propagation dynamics was already shown above to greatly decrease the computational cost with small loss in optimality. The global grid search for maximization of miss distance with linearized dynamics was 80x faster than its full dynamics counterpart, which showed a negligible gain in the miss distance maximization of just 5%. Using the same global search data generated for Table 2, the discrepancies between the full dynamics versus linearized cost functions evaluated over the whole solution space, i.e., for every discrete delta-V combination, is shown in Fig. 5. The discrepancy between the delta-V coordinates corresponding to the minimum of the cost functions in this solution space is also plotted in Fig. 5. It shows

the majority of the cost function discrepancy in this space is within 1.5 km, with some higher discrepancies observed and about 10% of the minima shifted from one side of the solution space to its opposite counterpart, from global minima to local minima. The discrepancy should not impact optimization in most cases, where the global minimum is much deeper than other minima. But even for cases where the global minimum is not as distinct from other local minima, the sub-optimal local minima in this case will be close enough to the global minima that it may not matter for the application. Regardless, due to this shift of the minima with this model approximation, a validation of the suggested avoidance algorithm output, *maneuver plan*, is warranted just as described in 3.1.

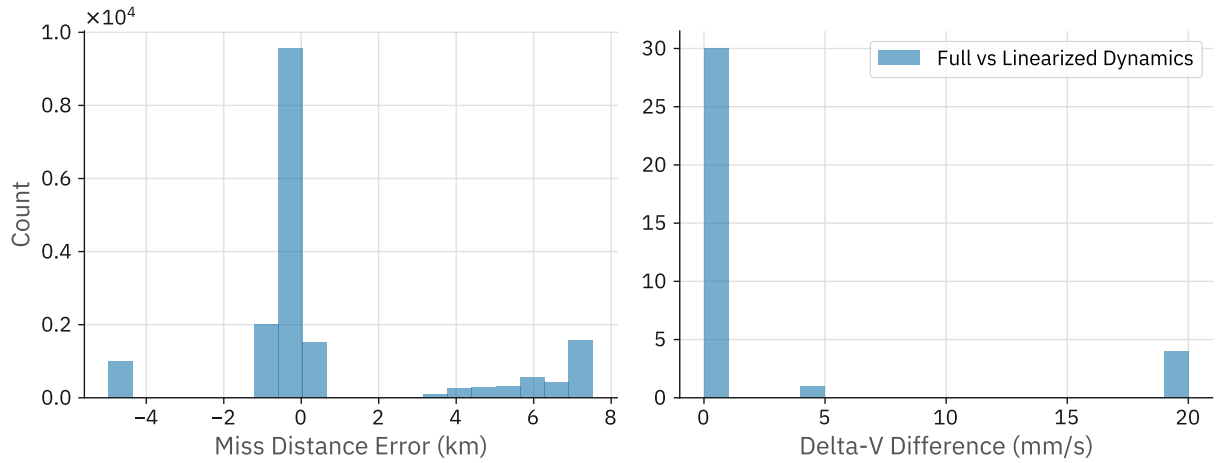


Fig. 5: Errors associated with the full dynamics vs. linearized cost function evaluations across all global grid search tradespaces, for CDM miss distance maximization. The left plot shows the histogram of these errors, and the right plot shows the delta-V difference between the global minimum of the two cost function spaces.

The issues observed for using original versus updated t_{ca} modeling are similar to those identified in the previous paragraph but exacerbated, as seen in a similar analysis shown in Fig. 6. The performance gain is much smaller, while minima shift is much less robust. More often, global minima for the original t_{ca} cost function actually correspond to local minima of the updated t_{ca} cost function, e.g., as illustrated in Fig. 7. The P_c minimization was specifically chosen for this analysis to highlight this issue with robustness. The Foster 2D PC implementation in NASA CARA codebase [29] assumes the input states are at the closest approach, and the relative position is on the B-plane. Thus, when the 2D PC is evaluated at the original t_{ca}, the computation incorrectly assumes a larger miss distance, incorporating the relative position displacement along η-direction, which explains the robustness issue.

Also in Fig. 6, the discrepancy between the numeric 2D P_c at the updated t_{ca} is compared against the analytic Chan's depth of intrusion. A sample of the cost functions for both are shown in Figs. 8 and 9, respectively. Chan's depth of intrusion metric is not as sensitive to t_{ca} changes because the relative position vector is mapped onto the B-plane, making the metric insensitive to the position shift along η-direction. The difference between Foster 2D PC at the original t_{ca} and Chan's PC highlights the significance of B-plane mapping to compute a miss vector. Since they share extrema locations, the depth of intrusion metric comes out as a suitable replacement for numeric 2D P_c as a cheaper objective function.

Finally, note that these approximations will limit the accuracy of constraint targeting. This holds especially true for the use of Chan's P_c depth of intrusion to target a 2D P_c.

5. CONCLUSION

The future of the space industry hinges on the establishment and wide adoption of STM and SEM policies and infrastructure. The main advancements required for STM were discussed in this work, in particular practical and effective maneuver suggestion generation, and automated data sharing and decision coordination. The current and projected market value of the STM industry was also discussed. To address these key areas of conjunction risk assessment, col-

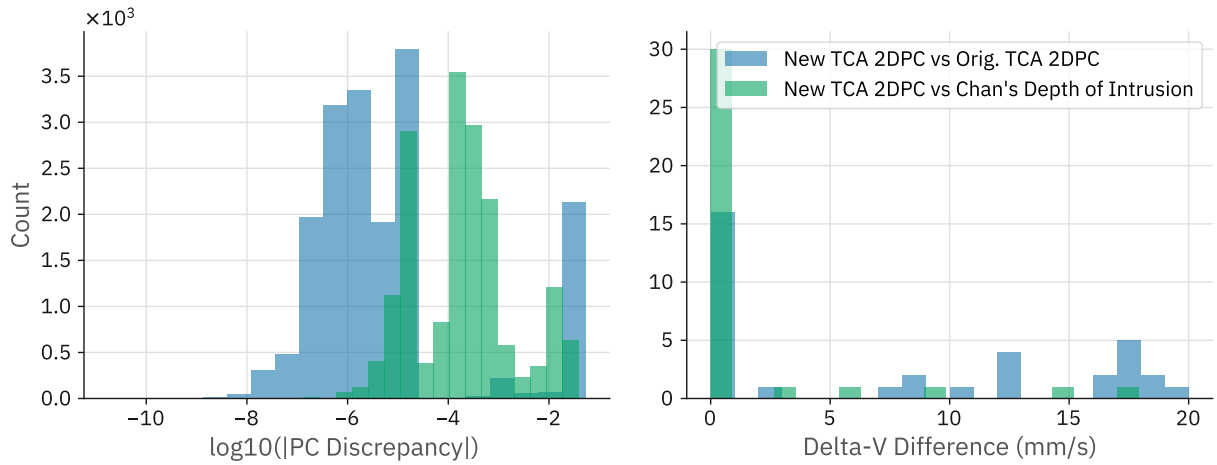
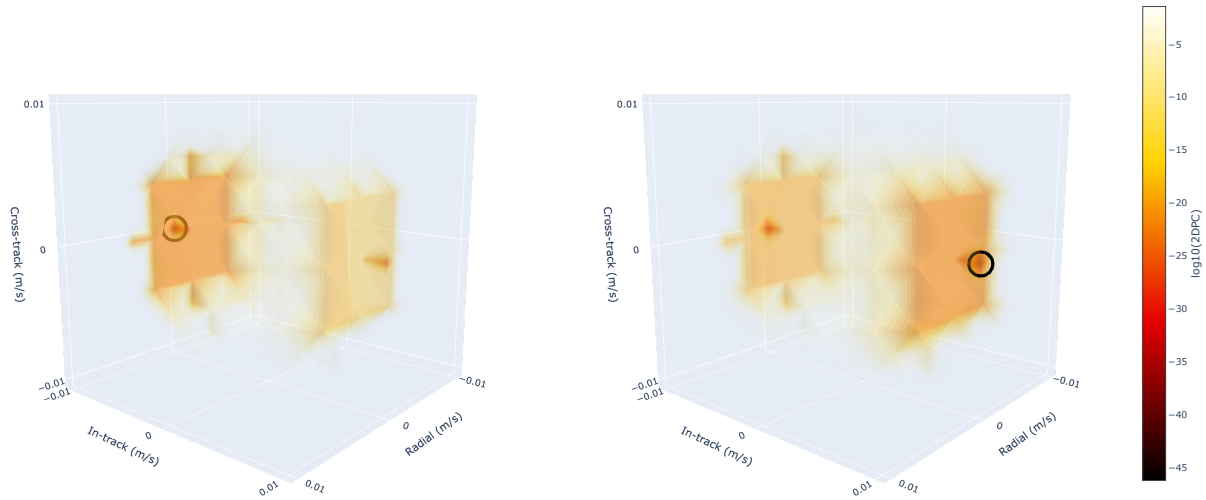


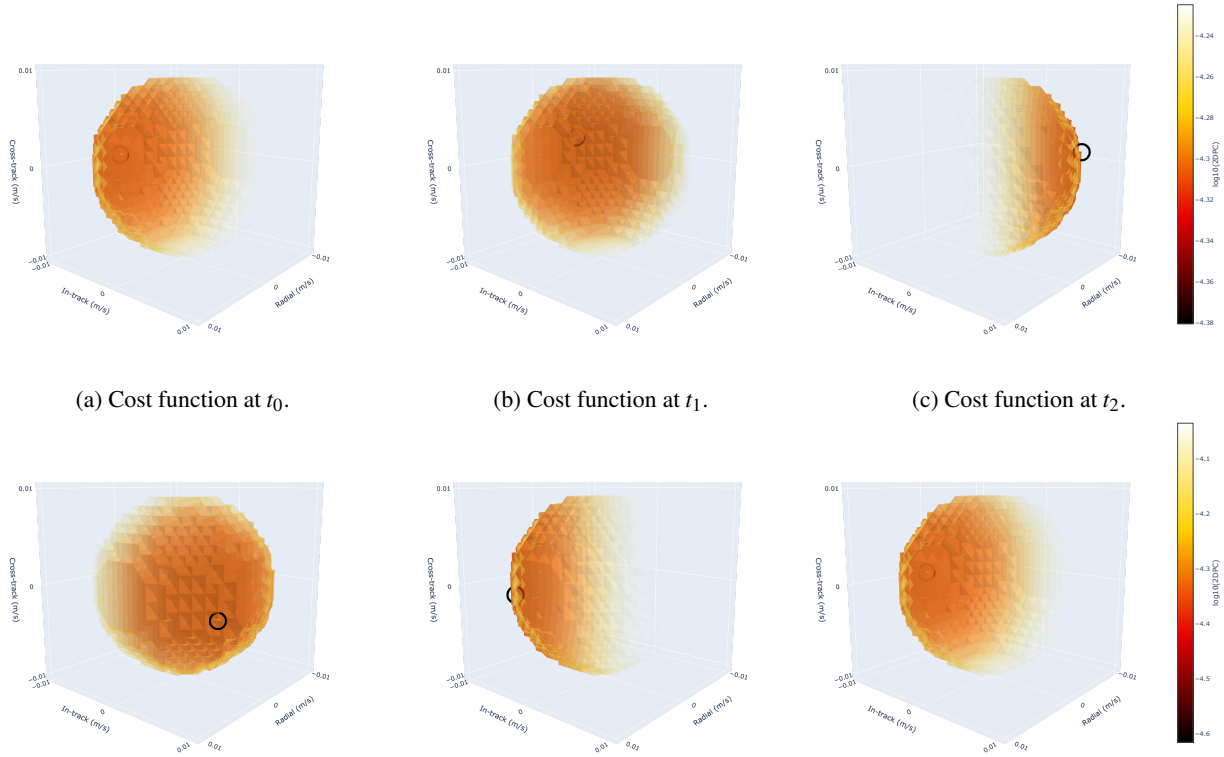
Fig. 6: Errors associated with the (1) original vs. updated t_{ca} (blue) and (2) Chan's depth of intrusion (green) cost function evaluations across all global grid search tradespaces, for CDM P_c minimization. The left plot shows the histogram of these errors, where Chan's depth of intrusion is transformed into Chan's P_c for proper comparison. The right plot shows the delta-V difference between the global minimum of the two cost function spaces.



(a) Cost function for numeric P_c at updated t_{ca}.

(b) Cost function for numeric P_c at original t_{ca}.

Fig. 7: Cost function for numeric P_c at updated and original t_{ca} on the solution space domain (with delta-V magnitude constraint), for a specific CDM and algorithm epoch corresponding to the same position in the orbit as the upcoming conjunction. A black circle indicates the global minimum in each case.



(a) Cost function at t_0.

(b) Cost function at t_1.

(c) Cost function at t_2.

(d) Cost function at t_3.

(e) Cost function at t_4.

(f) Cost function at t_5.

Fig. 8: Cost function for numeric P_c at updated t_{ca} on the solution space domain (with delta-V magnitude constraint), for a specific CDM and algorithm epoch. The algorithm epochs correspond to five uniformly-spaced times, t_0 through t_5, between two antipodal times around 12 hours prior to t_{ca}. A black circle indicates the global minimum in each case.

lision avoidance planning, and STC, Kayhan Space continues to improve its software-as-a-service (SaaS) Pathfinder platform. In particular, the avoidance maneuver suggestion engine driving Kayhan's coordinated avoidance maneuver framework was described and characterized in this paper. The engine's architecture, which was designed to solve the need for performant, comprehensive, and easy expandability, was presented at a high level. The supported optimization problems and avoidance algorithms were detailed. The engine's performance for certain collision mitigation strategies, including (1) maximization of miss distance, (2) minimization of P_c, and (3) minimization of delta-V while targeting maximum P_c, were characterized, along with several constraints, such as maneuver direction and shared-burden between spacecraft. Successful collision mitigation was consistently observed in each category of mitigation strategy for some common avoidance algorithms and optimization problem setup. The performance of certain modelling simplifications were also characterized including the use of linearized dynamics and Chan's depth of intrusion, and their suitable domain clarified. Future work planned includes characterizing Kayhan's end-end avoidance maneuver framework, and several additions to its avoidance maneuver engine. These additions may include support for multi-objective problems, new avoidance algorithms, decision variables, and optimization functions.

6. ACKNOWLEDGEMENTS

We would like to express our gratitude to the open-source software community for their invaluable contributions in technological and scientific fields and leveraged in this research.

7. REFERENCES

[1] ESA Space Debris Office. ESA's Annual Space Environment Report. Technical report, June 2023. URL https://www.sdo.esoc.esa.int/environment_report/Space_Environment_Report_latest.pdf.

[2] Salvatore Alfano, Daniel L. Oltrogge, and Lorenzo Arona. Operators' Requirements for SSA Services. *The Journal of the Astronautical Sciences*, 69(5):1441–1476, October 2022. ISSN 2195-0571. doi: 10.1007/s40295-022-00346-8. URL https://link.springer.com/10.1007/s40295-022-00346-8.

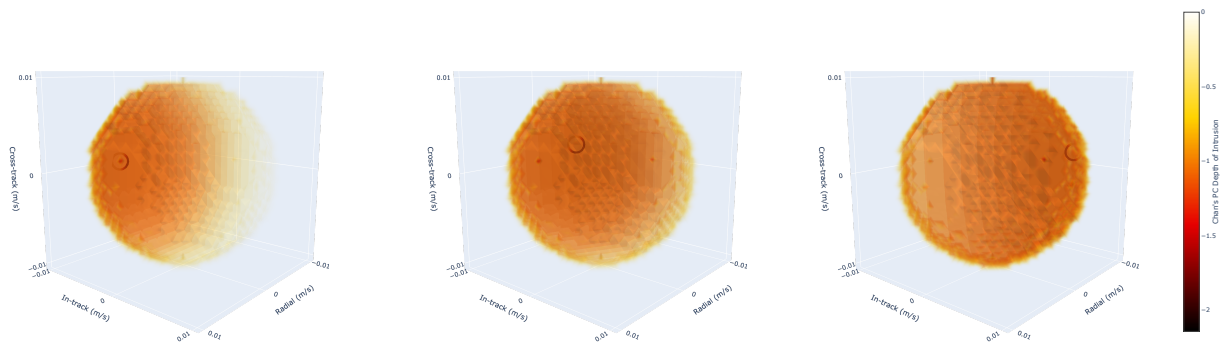
[3] Daniel L. Oltrogge and Salvatore Alfano. The technical challenges of better space situational awareness and space traffic management. *Journal of Space Safety Engineering*, 6(2):72–79, jun 2019. doi: 10.1016/j.jsse.2019.05.004.

[4] M. J. Holzinger and M. K. Jah. Challenges and potential in space domain awareness. *Journal of Guidance, Control, and Dynamics*, 41(1):15–18, jan 2018. doi: 10.2514/1.g003483.

[5] Michael Dominguez, Martin Faga, Jane Fountain, and Patrick Kennedy. Assessment of the Feasibility, Expected Effectiveness, and Funding Implications of a Transfer of Space Traffic Management Functions. 2020.

[6] World Economic Forum. Space Industry Debris Mitigation Recommendations. Technical report, June 2023.

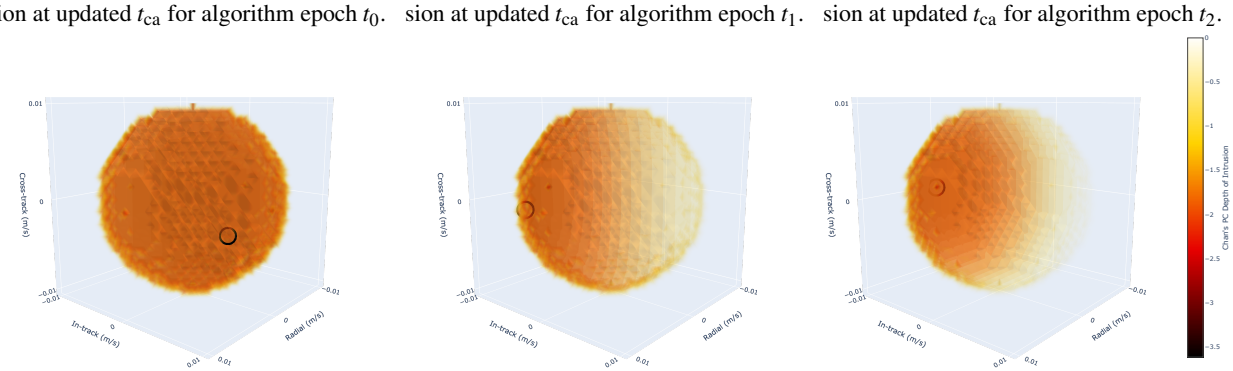
[7] Space Safety Coalition. Best Practices for the Sustainability of Space Operations. Technical report, May 2023.



(a) Cost function for Chan's depth of intrusion at updated t_{ca} for algorithm epoch t_0.

(b) Cost function for Chan's depth of intrusion at updated t_{ca} for algorithm epoch t_1.

(c) Cost function for Chan's depth of intrusion at updated t_{ca} for algorithm epoch t_2.



(d) Cost function for Chan's depth of intrusion at updated t_{ca} for algorithm epoch t_3.

(e) Cost function for Chan's depth of intrusion at updated t_{ca} for algorithm epoch t_4.

(f) Cost function for Chan's depth of intrusion at updated t_{ca} for algorithm epoch t_5.

Fig. 9: Cost function for Chan's depth of intrusion on the solution space domain (with delta-V magnitude constraint), for a specific CDM and algorithm epoch. The algorithm epochs correspond to five uniformly-spaced times, t_0 through t_5, between two antipodal times around 12 hours prior to t_{ca}. A black circle indicates the global minimum in each case.

[8] Space Data Association. Space Data Association. URL http://www.space-data.org/. http://www.space-data.org/.

[9] Samuel Hilton, Federico Cairola, Alessandro Gardi, Roberto Sabatini, Nichakorn Pongsakornsathien, and Neta Ezer. Uncertainty quantification for space situational awareness and traffic management. *Sensors*, 19(20):4361, oct 2019. doi: 10.3390/s19204361.

[10] Giulio Campiti, Giuseppe Brunetti, Vitali Braun, Eugenio Di Sciascio, and Caterina Ciminelli. Orbital kinematics of conjuncting objects in low-earth orbit and opportunities for autonomous observations. *Acta Astronautica*, 208: 355–366, jul 2023. doi: 10.1016/j.actaastro.2023.04.032.

[11] Dario Spiller, Edoardo Magionami, Vincenzo Schiattarella, Fabio Curti, Claudia Facchinetti, Luigi Ansalone, and Alberto Tuozzi. On-orbit recognition of resident space objects by using star trackers. *Acta Astronautica*, 177:478–496, dec 2020. doi: 10.1016/j.actaastro.2020.08.009.

[12] Dan L Oltrogge, P Wauthier, David A Vallado, Salvatore Alfano, and TS Kelso. Results of comprehensive stcm data fusion experiment. In *8th European Conference on Space Debris*, page 190, 2021.

[13] M. D. Hejduk and D. E. Snow. The effect of neutral density estimation errors on satellite conjunction serious event rates. *Space Weather*, 16(7):849–869, jul 2018. doi: 10.1029/2017sw001720.

[14] David J. Gondelach, Richard Linares, and Peng Mun Siew. Atmospheric density uncertainty quantification for satellite conjunction assessment. *Journal of Guidance, Control, and Dynamics*, 45(9):1760–1768, sep 2022. doi: 10.2514/1.g006481.

[15] European Space Agency. ESA & UNOOSA Space Debris Infographics and Podcast. Accessed 31 Aug 2023, https://www.esa.int/Space_Safety/Space_Debris/ESA_UNOOSA_space_debris_infographics_and_podcast.

[16] Yamin Huang, Linying Chen, Pengfei Chen, Rudy R. Negenborn, and P.H.A.J.M. van Gelder. Ship collision avoidance methods: State-of-the-art. *Safety Science*, 121:451–473, January 2020. ISSN 09257535. doi: 10.1016/j.ssci.2019.09.018. URL https://linkinghub.elsevier.com/retrieve/pii/S0925753519306356.

[17] Benjamin Bastida Virgili, Tim Flohrer, Holger Krag, Klaus Merz, and Stijn Lemmens. Cream-esa's proposal for collision risk estimation and automated mitigation. In *Proceedings of the 1st International Orbital Debris Conference. Sugar Land, USA*, volume 6031, 2019.

[18] Sreeja Nag, David Murakami, Miles Lifson, and Parimal Kopardekar. System Autonomy for Space Traffic Management. In *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*, pages 1–10, London, September 2018. IEEE. ISBN 978-1-5386-4112-5. doi: 10.1109/DASC.2018.8569343. URL https://ieeexplore.ieee.org/document/8569343/.

[19] Sreeja Nag, David D. Murakami, Nimesh A. Marker, Miles T. Lifson, and Parimal H. Kopardekar. Prototyping operational autonomy for Space Traffic Management. *Acta Astronautica*, 180:489–506, March 2021. ISSN 00945765. doi: 10.1016/j.actaastro.2020.11.056. URL https://linkinghub.elsevier.com/retrieve/pii/S0094576520307323.

[20] T. Maclay and D. McKnight. Space environment management: Framing the objective and setting priorities for controlling orbital debris risk. *Journal of Space Safety Engineering*, 8(1):93–97, March 2021. ISSN 24688967. doi: 10.1016/j.jsse.2020.11.002. URL https://linkinghub.elsevier.com/retrieve/pii/S2468896720301415.

[21] Sita Sonty, Cameron Scott, Troy Thomas, and Sarah Zhou. Herding rockets: Improved Space Traffic Management will accelerate industry growth. *Space News*, May 2022. URL https://spacenews.com/op-ed-herding-rockets-improved-space-traffic-management-will-accelerate-industry-growth/. Op-Ed.

[22] Juan Luis Gonzalo, Camilla Colombo, and Pierluigi Di Lizia. Analytical Framework for Space Debris Collision Avoidance Maneuver Design. *Journal of Guidance, Control, and Dynamics*, 44(3):469–487, 2021. ISSN 0731-5090. doi: 10.2514/1.G005398. URL https://doi.org/10.2514/1.G005398. Publisher: American Institute of Aeronautics and Astronautics _eprint: https://doi.org/10.2514/1.G005398.

[23] Ken Chan. Comparison of Methods for Spacecraft Maneuever Computations. In *2019 AAS/AIAA Astrodynamics Specialist Conference*, Portland ME, August 2019.

[24] Claudio Bombardelli and Javier Hernando-Ayuso. Optimal Impulsive Collision Avoidance in Low Earth Orbit. *Journal of Guidance, Control, and Dynamics*, 38(2):217–225, February 2015. ISSN 0731-5090, 1533-3884. doi: 10.2514/1.G000742. URL https://arc.aiaa.org/doi/10.2514/1.G000742. Number: 2.

[25] F. Kenneth Chan. *Spacecraft collision probability*. Aerospace Press ; American Institute of Aeronautics and Astronautics, El Segundo, Calif. : Reston, Va, 2008. ISBN 978-1-884989-18-6.

[26] The Consultative Committee for Space Data Systems. *Conjunction Data Message Recommended Standard CCSDS 508.0-B-1 Blue Book*. Washington, DC, USA, 2013. URL .https://public.ccsds.org/Pubs/508x0b1e2s.pdf.

[27] J.L. Foster and H.S. Estes. *A Parametric Analysis of Orbital Debris Collision Probability and Maneuver Rate for Space Vehicles*. NASA, National Aeronautics and Space Administration, Lyndon B. Johnson Space Center, 1992. URL https://books.google.com/books?id=SJ0hygEACAAJ.

[28] Doyle T. Hall. Expected Collision Rates for Tracked Satellites. *Journal of Spacecraft and Rockets*, 58(3):715–728, May 2021. ISSN 0022-4650, 1533-6794. doi: 10.2514/1.A34919. URL https://arc.aiaa.org/doi/10.2514/1.A34919.

[29] Nasa cara analysis tools. https://github.com/nasa/CARA_Analysis_Tools. Accessed: 2023-08-29.

[30] Yurii Nesterov. A method for unconstrained convex minimization problem with the rate of convergence o (1/k2). In *Dokl. Akad. Nauk. SSSR*, volume 269, page 543, 1983.

[31] Andreas Wächter and Lorenz T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25–57, apr 2005. doi: 10.1007/s10107-004-0559-y.

[32] Francesco Biscani and Dario Izzo. A parallel global multiobjective framework for optimization: pagmo. *Journal of Open Source Software*, 5(53):2338, 2020. doi: 10.21105/joss.02338. URL https://doi.org/10.21105/joss.02338.

[33] Christian Blum. Ant colony optimization: Introduction and recent trends. *Physics of Life Reviews*, 2(4):353–373, dec 2005. doi: 10.1016/j.plrev.2005.10.001.

[34] Martin Schlüter, Jose A. Egea, and Julio R. Banga. Extended ant colony optimization for non-convex mixed integer nonlinear programming. *Computers & Operations Research*, 36(7):2217–2229, jul 2009. doi: 10.1016/j.cor.2008.08.015.