Efficient Evaluation of 2-D Collision Probability Derivatives for Uncertain k-scaled Covariances: A PcMax Case Study

Denis Arzelier, Mioara Joldeş, Matthieu Masson

Efficient Evaluation of 2-D Collision Probability Derivatives for Uncertain k-scaled Covariances: A PcMax Case Study

Denis Arzelier*, Mioara Joldes†, and Matthieu Masson‡

Abstract

This paper focuses on efficiently evaluating derivatives of the 2-D collision probability, treated as a function of parameters, which appear as linear forms in the entries of the covariance matrix. This boils down to computing moments of the associated Gaussian measure restricted to a disk. Specifically, we propose an optimization-based solution to computing the maximum collision probability when the covariance data is unreliable, implementing an alternative method to the traditional k-scaled covariance approach. Preliminary results indicate our method's potential for improving the understanding of Pc's validity as a measure of conjunction likelihood.

1 Introduction

The rapidly increasing number of near-earth Resident Space Objects (RSOs), including a large proportion of space debris has been a strong incentive for the development of Space Surveillance and Tracking (SST) applications. Indeed, the last years have witnessed a great deal of effort on satellite conjunction risk analysis and reliable orbit collision risks assessment procedures are particularly vital for maintaining the safety and sustainability of space operations in the future. The data collected for these procedures (namely, relative positions of the two objects involved in a conjunction) are uncertain leading the Owners/Operators (O/O) of space assets and space agencies or organizations, to rely on the Probability of Collision (PoC) as a quantifiable metric of decision. When its value exceeds some tolerance threshold, a maneuver is performed. As a consequence, methods that incorporate a probabilistic representation of orbital uncertainties and provide estimates of the collision likelihood were developed in the literature in the last 20 years[1–11] depending on the assumptions defining the encounter model.

*Senior Researcher, LAAS-CNRS, 7, Av. du Colonel Roche, 31031, Toulouse.
†Senior Researcher, LAAS-CNRS, 7, Av. du Colonel Roche, 31031, Toulouse.
‡PhD Student, LAAS-CNRS, 7, Av. du Colonel Roche, 31031, Toulouse.

Among those, the two-dimensional (2D) probability of collision (Pc) computation is presently a fast, primarily analytical standard tool used by the vast majority of conjunction assessment practitioners to estimate the likelihood of a collision. Under the following three main assumptions: 1-) the collision duration is short enough to allow for a rectilinear relative motion; 2-) their relative position uncertainty follows a 3d multivariate Gaussian distribution; 3-) the velocities are not uncertain implying that the relative position covariance matrix is time-invariant during the encounter, the Pc computation boils down to the integration of a two-dimensional Gaussian Probability Density Function (PDF) on a disk of radius R (the spherical objects' combined hard-body radius), centered at 0. Besides their simplicity of formulation, the numerical methods[1–7] designed for the integration are computationally very fast and practical, explaining the intense enthusiasm of its use by decision-makers in the space industries. Nevertheless, the accuracy of this computation heavily relies on a proper knowledge of the parameters (i.e. the hard-body radius R, the 2 entries of the mean relative position vector and the entries of the 2×2 relative position covariance matrix projected in the encounter plane) entering the final formula of the Pc. For various reasons detailed in the literature[12–15] and mainly related to the actual limitations of the orbit determination process of the covariance matrices, these parameters may not completely characterize the uncertainty and errors involved in the estimation of Pc.This situation is particularly illustrated by the so-called dilution of the probability phenomenon for which detailed explanations may be found in the references[13,15,16] . In short, it may be observed that the collision risk measured by Pc counter-intuitively decreases when the uncertainty level (measured by the ratio of the covariance size over the miss distance, see Figure 2 in[15]) is beyond some given value for which Pc reaches a maximum PcMax, separating the *robust region* in which Pc is increasing from *the dilution region* where Pc is decreasing. This can be interpreted as the maximum value of the aforementioned integral Pc, when some of the parameters involved in its calculation are poorly known or unreliable.

Several works[12–15,17,18] addressed the problem of covariance realism and the associated dilution phenomenon by resorting to variations of the "k-scaling" (see for instance the "`kp-ks` method"[17,19] implemented by Centre National d'Etudes Spatiales) approach for covariance dilation and computation of PcMax. The combined covariance matrix in the conjunction plane (alternatively the primary and secondary covariance matrices or only the secondary covariance[14]) is scaled by a set of factors belonging to some closed interval defined by a minimum and a maximum value (typically 0.25 and 4 for the reference[17]) and the resulting PcMax is computed and compared to a threshold for mitigation decision. The numerical methods which aim at computing the PcMax roughly fall into two categories. In 2005 S. Alfano proposed a closed-form approximation of the PcMax by reducing the overall size of the covariance while maintaining its aspect ratio and the event's nominal miss distance. An implementation is reported for the STK/CAT tool[20] . It was pointed out in[15] that operational experience suggests that the underlying assumptions for this method may not be satisfied in practice.

Another approach (the so-called `kp-ks` method[17,18]), which takes into account the reshaping of the joint covariance via two scaling factors on the primary and secondary covariance matrices, is reportedly implemented by Centre National d'Etudes Spatiales (CNES) in the European Space Surveillance and Tracking (EU SST) Support Framework[19] . The impact of covariance uncertainty on the probability of collision is analyzed by a brute force gridding technique. A similar way of dealing with the dilution effect was implemented also in the Conjunction Assessment and Risk Analysis Software Development Kit (CARA SDK) of NASA[1], by performing a logarithmic grid on the scaling factors values which are then multiplied by the original covariance reported at the Time of Closest Approach (TCA) for both objects (before the projection on the encounter plane). An alternative[14] method, dedicated to conjunction events for which only one covariance matrix is known, is also implemented in CARA's software.

However, according to the present CARA's best practices recommendations[21] , "no single set of best practices has emerged to govern how to address conjunction events in this region.[...] Maximum Pc methods might be deployable with current event densities but could not be sustained with the expected increases that the Space Fence radar will bring;[...] As such, it is CARA's present recommended practice to treat the Pc, regardless of dilution region positioning, as a durable assessment of collision likelihood for the purposes of considering and selecting remediation actions". Our main objective is to provide more sound and well-grounded mathematical tools which allow for a deeper understanding of what should be the best practices in these cases. Keeping these key points in mind, the first contribution of this article is to show that the partial derivatives of Pc, seen as a function of some parameters, which appear as linear forms in the entries of the covariance matrix, can be efficiently computed. These derivatives are expressed in closed-form as a linear combination of some moments of the associated Gaussian measure restricted to the disk of radius R. If their evaluation is needed in practice, one accurate (i.e. not using the basic finite difference methods) evaluation of a partial derivative is roughly equivalent to 4 Pc evaluations, while the second order derivative accounts to 8 Pc evaluations. Furthermore, we show that the methods[6,22] based on the theory of so-called holonomic or D-finite functions can be easily adapted to the evaluation of moments and by consequence to the derivatives evaluation. The second objective is to show how the efficient evaluation of the Pc derivatives of first order (and possibly of second order) can be used to leverage the numerical solving of optimization problems involving the Pc such as the computation of PcMax.

We proceed by briefly recalling the notations and the basic assumptions relevant to the Pc computation and to our analysis.

[1]The implementation is available in the distributed code `https://github.com/nasa/CARA_Analysis_Tools`

2 The 2D collision probability Pc between spherical objects

The short-term encounter model for the computation of the probability of collision Pc between two objects (a primary object p, typically an active spacecraft, and a secondary object s) considered as spheres of combined radius R is now precisely defined and well understood in the literature.[1–7,10,11] The derivations of the various formulations of Pc rely on four main ingredients: 1) a list of simplifying assumptions[5,6] defining the type of encounter dealt with and the possible mathematical derivations, 2) the encounter frame characterizing the encounter plane at TCA (orthogonal to the relative velocity vector assumed to be invariant in direction and module during the encounter), 3) the volume swept out by the combined spherical object (a cylinder with its axis aligned along the relative velocity vector), 4) an approximation based on the extension of the swept volume allowing to compute Pc as the evaluation of the integral of the a two-dimensional normal Probability Density Function (PDF) within a circle, which is the projection of the hard-body on the encounter plane, instead of the integral of a three-dimensional normal PDF evaluated within the initial swept volume. These essential features and notations required for the derivations of the formulation of the probability of collision Pc, relevant to the next section, are now quickly recalled.

The so-called short-term encounter model, applies to cases with high relative velocities assumed to be precisely known and the relative motion is approximated as linear and the position vectors of the primal and secondary follow a constant Gaussian multivariate distribution at TCA (assumed to be 0) defined as:

$$x_p^0 \sim \mathcal{N}_3\left(\mu_p^0, \Sigma_p^0\right), \quad x_s^0 \sim \mathcal{N}_3\left(\mu_s^0, \Sigma_s^0\right). \tag{1}$$

The above assumptions motivate the choice of introducing the so-called encounter frame, which has one axis along the relative velocity and is centered on the mean position of one of the two objects. The plane containing the origin, located at the mean position of the primary object p at TCA, orthogonal to the relative velocity $v_r^0 = v_s^0 - v_p^0$ is called the encounter plane. The basis vector $e_{\tilde{z}}$ is oriented along the relative velocity. The basis vector $e_{\tilde{x}}$ belongs to the encounter plane: it points towards the orthogonal projection of the mean relative position μ_r^0 of the relative vector $r^0 = x_s^0 - x_p^0$ within the encounter plane. Finally, the basis vector $e_{\tilde{y}}$ completes the right-handed system and thus belongs to the encounter plane as well. In summary, one has

$$e_{\tilde{z}} = \frac{v_r^0}{\|v_r^0\|}, \quad e_{\tilde{y}} = \frac{v_r^0 \times \mu_r^0}{\|v_r^0 \times \mu_r^0\|}, \quad e_{\tilde{x}} = e_{\tilde{y}} \times e_{\tilde{z}}. \tag{2}$$

The statistical characteristics (mean position vectors and covariance matrix) of both objects at TCA are then expressed in this basis (encounter frame) via a linear transformation which preserves the normal nature of their distribution. The distribution of the relative vector $\tilde{r}_0$ (at TCA, in the encounter frame) is easily deduced as:

$$\begin{aligned} &\tilde{r}^0 \sim \mathcal{N}_3\left(\tilde{\mu}_r^0, \tilde{\Sigma}_r^0\right), \\ &\tilde{\mu}_r^0 = \tilde{\mu}_s^0 - \tilde{\mu}_p^0, \\ &\tilde{\Sigma}_r^0 = \tilde{\Sigma}_p^0 + \tilde{\Sigma}_s^0. \end{aligned} \tag{3}$$

Let $(\tilde{x}_m, 0, \tilde{z}_m)$ be the coordinates of the mean relative position $\tilde{\mu}_r^0$ in the encounter frame (where the $\tilde{y}$ coordinate is zero by construction). It is worth noticing that, since the relative trajectory is rectilinear, $\tilde{x}_m$ is in fact equal to the mean miss distance. Furthermore, since the relative motion is assumed to be rectilinear along v_r^0 during the conjunction, the calculation of the collision probability reduces to the integration of the aforementioned 3-dimensional multi-normal distribution over a right circular cylinder domain (swept-volume by the hard-body) and then reduced by integration on $\mathbb{R}$ with respect to the v_r^0 direction to the calculation of a 2-D integral in the encounter plane: the domain of integration is a closed disk $\bar{\mathcal{B}}((0,0), R)$ centered at the origin of radius R. The 2-D PDF involved in the probability of collision describes the distribution of the relative position in the encounter plane, as illustrated on Figure 1.

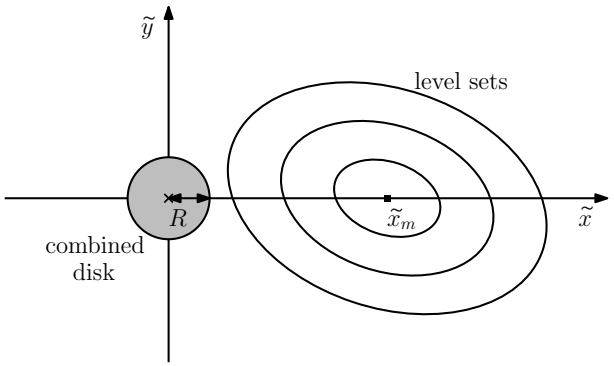

Figure 1: Illustration of 2-D projection of the the hard-body and combined covariances ellipsoids in the encounter plane.

Let $\tilde{\Sigma}_{\tilde{x}\tilde{y}}$ be the covariance matrix of the relative coordinates in the encounter plane (obtained by the first two lines and columns of the matrix $\tilde{\Sigma}_r^0$). The probability of collision can then be written as:

$$Pc = \frac{1}{2\pi\sqrt{|\tilde{\Sigma}_{\tilde{x}\tilde{y}}|}} \iint_{\bar{\mathcal{B}}((0,0),R)} \exp\left(-\frac{1}{2}\left[\tilde{x}-\tilde{x}_m \quad \tilde{y}\right]\tilde{\Sigma}_{\tilde{x}\tilde{y}}^{-1}\left[\tilde{x}-\tilde{x}_m \quad \tilde{y}\right]^T\right) \mathrm{d}\tilde{x}\mathrm{d}\tilde{y}. \tag{4}$$

Equation (4) is one of the classical formulations commonly used in the literature for computing the short-term collision probability and several numerical algorithms have been proposed for its evaluation[2–7] . In Equation (4), the integral Pc clearly depends on the 5 parameters R, $\tilde{x}_m$ and the three independent entries of the covariance matrix $\tilde{\Sigma}_{\tilde{x}\tilde{y}}$. The values affected to these parameters

are the direct outputs of RSO tracking problems for which a proper representation of uncertainty is at stake. For instance, it is well-known that position covariance matrices at TCA of the two RSOs may not accurately represent the actual dispersion of position errors due to imperfect knowledge of the dynamic behaviour of the debris or to oversimplifying assumptions used for the propagation of uncertainty for instance. The next section focuses on characterizing the sensibility of Pc to uncertainty affecting the covariance matrix $\tilde{\Sigma}_{\tilde{x}\tilde{y}}$ in the framework of k-scaled covariances and the computation of PcMax.

3 A Case Study - Covariance Scaling and PcMax computation

In the so-called `kp-ks` approach developed by CNES (a variation around this method is also implemented in CARA SDK), it is assumed that the covariance $\tilde{\Sigma}_p^0, \tilde{\Sigma}_s^0 \in \mathbb{R}^{3\times 3}$ in Equation 3 are not precisely numerically known and, models the distribution of the relative vector $\tilde{r}_0$ (in the encounter frame) by:

$$\begin{aligned} &\tilde{r}^0 \sim \mathcal{N}_3\left(\tilde{\mu}_r^0, \tilde{\Sigma}_r^0\right), \\ &\tilde{\mu}_r^0 = \tilde{\mu}_s^0 - \tilde{\mu}_p^0, \\ &\tilde{\Sigma}_r^0 = k_p\tilde{\Sigma}_p^0 + k_s\tilde{\Sigma}_s^0,\ k_p > 0,\ k_s > 0, \end{aligned} \tag{5}$$

where $(k_p, k_s) \in I$ are user-defined parameters, belonging to a specific range I. Usually in practice one sets $I = [0.25, 4] \times [0.25, 4]$[19] . It follows that the 2-D calculation of the collision probability now depends on these two parameters:

$$Pc(k_p, k_s) = \frac{1}{2\pi\sqrt{|\tilde{\Sigma}_{\tilde{x}\tilde{y}}(k_p, k_s)|}} \int\limits_{\tilde{\mathcal{B}}((0,0),R)} \exp\left(-\frac{1}{2}\xi\tilde{\Sigma}_{\tilde{x}\tilde{y}}(k_p, k_s)^{-1}\xi^T\right) \mathrm{d}\tilde{x}\mathrm{d}\tilde{y}, \tag{6}$$

where $\xi = [\tilde{x} - \tilde{x}_m \quad \tilde{y}]$.

Assuming a uniform distribution of values for these two parameters in the interval I, a grid of Pc values is then built and a PcMax may be identified which is compared to the Pc threshold defining the activation of a mitigation action from the active asset. Different variations revolving around this very same idea, may be found in the CARA SDK and in the references[13,14] . Formally, the problem of computing PcMax in this context resorts to solving the following optimization problem:

$$\max_{(k_p,k_s)\in I} \quad P_c(k_p, k_s). \tag{7}$$

In this work, we propose a new tractable mathematical reformulation, which allows for an efficient numerical solving of Problem (7) by standard numerical methods. The key observation is that the partial successive derivatives of the function $P_c(k_p, k_s)$ with respect to these two parameters can be efficiently computed. In the next subsection we show how the gradient of the function

$P_c(k_p, k_s)$ can be efficiently numerically evaluated. Firstly, we provide a closed-form formula which depends on few moments (4) of the Gaussian measure restricted to the disk corresponding to Equation (4). Secondly, we briefly discuss how the theoretical ground laid in our previous work[6, 22] allows for an efficient evaluation of the required moments.

3.1 Explicit Gradient Formula for Pc

Firstly, let us fix some notations. The covariance matrix is a real symmetric positive definite 2-by-2 matrix, whose entries are denoted by

$$\tilde{\Sigma}_{\tilde{x}\tilde{y}}(k_p, k_s) = \begin{bmatrix} k_p s_{p1} + k_s s_{s1} & k_p s_{p2} + k_s s_{s2} \\ k_p s_{p2} + k_s s_{s2} & k_p s_{p3} + k_s s_{s3} \end{bmatrix}, \tag{8}$$

where s_{pi}, s_{si}, $i = 1, \ldots, 3$, are given corresponding entries in the upper 2-by-2 part of the transformed primary and secondary covariance matrices appearing in Equation (5). Denoting the determinant of $\tilde{\Sigma}_{\tilde{x}\tilde{y}}(k_p, k_s)$ by

$$D(k_p, k_s) := -(k_p s_{p2} + k_s s_{s2})^2 + (k_p s_{p1} + k_s s_{s1})(k_p s_{p3} + k_s s_{s3}), \tag{9}$$

and

$$S_p := \begin{bmatrix} s_{p3} & -s_{p2} \\ -s_{p2} & s_{p1} \end{bmatrix}, \qquad S_s := \begin{bmatrix} s_{s3} & -s_{s2} \\ -s_{s2} & s_{s1} \end{bmatrix}, \tag{10}$$

the inverse $Q(k_p, k_s)$ of $\tilde{\Sigma}_{\tilde{x}\tilde{y}}(k_p, k_s)$ becomes by straightforward calculation

$$Q(k_p, k_s) := \tilde{\Sigma}_{\tilde{x}\tilde{y}}(k_p, k_s)^{-1} = D(k_p, k_s)^{-1} \left(k_p S_p + k_s S_s\right). \tag{11}$$

Then, for the quadratic form appearing in the exponential of Equation (4), denoted for simplicity by $f : I \to \mathbb{R}$, $I \subset \mathbb{R}^2$, $f(k_p, k_s) := -(\xi Q(k_p, k_s)\xi^T)/2$, the gradient can be computed in explicit form as follows (the equations omit the explicit dependence on the parameters for simplicity and the partial derivative is denoted by ∂):

$$\nabla f := D^{-1}/2 \begin{bmatrix} (\partial_{k_p} D)\xi Q\xi^T - \xi S_p \xi^T \\ (\partial_{k_s} D)\xi Q\xi^T - \xi S_s \xi^T \end{bmatrix} = -D^{-1} \begin{bmatrix} (\partial_{k_p} D) f + \xi S_p \xi^T/2 \\ (\partial_{k_s} D) f + \xi S_s \xi^T/2 \end{bmatrix}, \tag{12}$$

and

$$\nabla^2 f \;=\; 2D^{-2} f \nabla D (\nabla D)^T - D^{-1} f \nabla^2 D + \frac{D^{-2}}{2} \left[\begin{bmatrix} \xi S_p \xi^T \\ \xi S_s \xi^T \end{bmatrix} (\nabla D)^T + \nabla D \begin{bmatrix} \xi S_p \xi^T \\ \xi S_s \xi^T \end{bmatrix}^T \right]. \tag{13}$$

Similarly, for the quotient multiplying the integral in Equation (6), denoted by $g(k_p, k_s) := D(k_p, k_s)^{-1/2}$, one has:

$$\nabla g := -\frac{D^{-3/2}}{2} \begin{bmatrix} (\partial_{k_p} D) \\ (\partial_{k_s} D) \end{bmatrix} = -\frac{D^{-3/2}}{2} \nabla D, \tag{14}$$

and

$$\nabla^2 g \quad = \quad \frac{D^{-3/2}}{2}\left(\tfrac{3}{2}D^{-1}\nabla D(\nabla D)^T - \nabla^2 D\right). \tag{15}$$

Using the above equations, one can compute the gradient of $Pc(k_p, k_s)$ in Equation (6) with Leibniz rule (since we are allowed to differentiate under the integral sign):

$$\nabla Pc := \frac{Pc}{g}\nabla g + \frac{g}{2\pi}\int\limits_{\bar{\mathcal{B}}((0,0),R)} \exp(f)\nabla f \mathrm{d}\tilde{x}\mathrm{d}\tilde{y}, \tag{16}$$

where the convention that the integral is taken component-wise is made.

The key step is how to efficiently evaluate the integral appearing in Equation (16). With this in mind, let us analyze only the first integral, involved in the derivative with respect to k_p, since the second one is similar. One observes in Equation (12) that

$$\partial_{k_p} f = D^{-1}/2\left((\partial_{k_p} D)\xi Q\xi^T - \xi S_p \xi^T\right),$$

is a bivariate polynomial of total degree 2 with respect to the variables $(\tilde{x}, \tilde{y})$, which can be written symbolically using the notations in Equation (4) in the form:

$$\partial_{k_p} f := c_{p20}(\tilde{x} - \tilde{x}_m)^2 + c_{p02}\tilde{y}^2 + c_{p11}(\tilde{x} - \tilde{x}_m)\tilde{y}, \tag{17}$$

where the coefficients c_{pij} (and similarly c_{sij} corresponding to $\partial_{k_s} f$) are identified as rational fractions depending on the parameters k_p and k_s.

It follows that the evaluation of ∇Pc (at given values of k_p and k_s) mainly depends on the evaluation of the moments of order at most 2 of the bivariate Gaussian measure (restricted to the disk of radius R) appearing in Equation (4). Formally, the required moments m_{ij} are defined by:

$$m_{ij} := \frac{g}{2\pi}\int\limits_{\bar{\mathcal{B}}((0,0),R)} (\tilde{x} - \tilde{x}_m)^i \tilde{y}^j \exp\left(-\frac{1}{2}\begin{bmatrix}\tilde{x} - \tilde{x}_m & \tilde{y}\end{bmatrix} Q \begin{bmatrix}\tilde{x} - \tilde{x}_m & \tilde{y}\end{bmatrix}^T\right) \mathrm{d}\tilde{x}\mathrm{d}\tilde{y}. \tag{18}$$

Finally, Equation (16), becomes

$$\nabla Pc := \frac{m_{00}}{g}\nabla g + \begin{bmatrix} c_{p20} & c_{p02} & c_{p11} \\ c_{s20} & c_{s02} & c_{s11}\end{bmatrix}\begin{bmatrix} m_{20} & m_{02} & m_{11}\end{bmatrix}^T. \tag{19}$$

Following the same lines, an explicit expression for the Hessian matrix of Pc is

deduced:

$$\begin{aligned}\nabla^2 Pc &:= \frac{Pc\nabla^2 g}{g} + \frac{1}{2\pi}\int_{\bar{\mathcal{B}}((0,0),R)} \exp(f)\nabla f(\nabla g)^T \mathrm{d}\tilde{x}\mathrm{d}\tilde{y} + \cdots \\ &\quad \frac{\nabla g}{2\pi}\int_{\bar{\mathcal{B}}((0,0),R)} (\nabla f)^T \exp(f)\mathrm{d}\tilde{x}\mathrm{d}\tilde{y} + \cdots \\ &\quad \frac{g}{2\pi}\int_{\bar{\mathcal{B}}((0,0),R)} \exp(f)\left(\nabla^2 f + \nabla f(\nabla f)^T\right)\mathrm{d}\tilde{x}\mathrm{d}\tilde{y} \\ &= \frac{m_{00}\nabla^2 g}{g} + Cm\frac{(\nabla g)^T}{g} + \frac{(\nabla g)}{g}m^T C + \cdots \\ &\quad \begin{bmatrix} \frac{\partial C}{\partial k_p}m & \frac{\partial C}{\partial k_s}m \end{bmatrix} + C\begin{bmatrix} m_{40} & m_{22} & m_{31} \\ m_{22} & m_{04} & m_{13} \\ m_{31} & m_{13} & m_{22} \end{bmatrix} C^T,\end{aligned} \tag{20}$$

where $C = \begin{bmatrix} c_{p20} & c_{p02} & c_{p11} \\ c_{s20} & c_{s02} & c_{s11} \end{bmatrix}$, $m = [m_{20} \quad m_{02} \quad m_{11}]^T$, and $\frac{\partial C}{\partial *}$ is the matrix which entries are the partial derivatives of the entries of C with respect to k_p or k_s. Provided that the required moments can be efficiently evaluated, numerical values of the gradient are available, which allow for the application of standard numerical optimization methods to compute a good approximation of PcMax.

Remark 1 *Closed-form expressions for the Hessian matrix could be used in second order optimization methods (quasi-Newton methods for instance).*

3.2 Moments Evaluation

Let us now focus on the efficient numerical evaluation of moments of the form (18). The theoretical framework based on Laplace Transform and the class of so-called holonomic functions (solutions of linear differential equations with polynomial coefficients) developed in details in[6, 22] can also be used in this similar case. For completeness, we briefly recall the technique and use it for the moment m_{11}:

$$m_{11} := \frac{D^{-1/2}}{2\pi}\int\limits_{\bar{\mathcal{B}}((0,0),R)} (\tilde{x}-\tilde{x}_m)\tilde{y}\exp\left(-\frac{1}{2}\left[\tilde{x}-\tilde{x}_m \quad \tilde{y}\right] Q \left[\tilde{x}-\tilde{x}_m \quad \tilde{y}\right]^T\right)\mathrm{d}\tilde{x}\mathrm{d}\tilde{y}. \tag{21}$$

- Step 0: Perform a rotation to the principal axis of the covariance matrix in the encounter plane. The new coordinates, denoted $(\hat{x}, \hat{y})$, are respectively along the major and the minor axis of the ellipse defined by the quadratic form in the exponential function of Equation (21). Furthermore, regard m_{11} as a function of R, all the other parameters being fixed. Denote by $\hat{\sigma}_x$ and $\hat{\sigma}_y$ the standard deviations associated to the new coordinates. The integral to evaluate is equivalent to the following one:

$$h(R) := \frac{1}{2\pi\hat{\sigma}_x\hat{\sigma}_y} \int\limits_{\bar{\mathcal{B}}((0,0),R)} (\hat{x}-\hat{x}_m)(\hat{y}-\hat{y}_m) \exp\left(-\frac{1}{2}\left(\frac{(\hat{x}-\hat{x}_m)^2}{\hat{\sigma}_x^2} + \frac{(\hat{y}-\hat{y}_m)^2}{\hat{\sigma}_y^2}\right)\right) \mathrm{d}\hat{x}\mathrm{d}\hat{y}. \tag{22}$$

- Step 1: Laplace Transform. A closed-form expression of the Laplace transform $\mathcal{L}_h$ of $h(R) := m_{11}(R)$ may be easily obtained as:

$$\begin{aligned} \mathcal{L}_h(z) &= \int_0^{+\infty} \exp(-z\zeta) h(\zeta) \mathrm{d}\zeta \\ &= \frac{\exp\left(-z\left(\frac{\hat{x}_m^2}{2z\hat{\sigma}_x^2+1} + \frac{\hat{y}_m^2}{2z\hat{\sigma}_y^2+1}\right)\right)}{z\sqrt{2z\hat{\sigma}_x^2+1}\sqrt{2z\hat{\sigma}_y^2+1}} \cdot \frac{4\hat{\sigma}_x^2\hat{\sigma}_y^2\hat{x}_m\hat{y}_m z}{(2z\hat{\sigma}_x^2+1)(2z\hat{\sigma}_y^2+1)}, \text{ for all } \mathrm{Re}(z) \geq 0. \end{aligned} \tag{23}$$

  The reader will identify the first term of the above product with the Laplace transform of the Pc. This gives the intuition that the Laplace transform for the moments correspond to the Laplace transform of the Pc multiplied by a rational fraction.

- Step 2: From the closed-form of the function h, one can infer that h and therefore m_{11} are so-called holonomic functions with respect to all the parameters involved (that is, solutions of sufficiently many linear homogeneous differential equations with polynomial coefficients).

- Step 3. At this point, two main ideas for obtaining a numerical approximation may be considered. They were analyzed in detail in[6, 22] for the computation of the 2D Pc. Either, the aforementioned Laplace transform is expanded in a Laurent series on which a term-by-term application of the inverse Laplace transform provides a convergent power series expansion. Alternatively, the inverse Laplace transform formula leads to a complex-variable integral, which is evaluated by the so-called saddle-point method.

It is important to remark that the moments m_{ij} are holonomic functions (with respect to all the parameters involved) and can be evaluated with the same computational complexity as the original integral for the 2D Pc with the methods proposed in[6, 22] .

3.3 Optimization Methods for Problem 7

Given that gradient information is available via efficient numerical evaluations of moments, Problem 7 can be solved with classical gradient descent related methods. In fact, for the case of holonomic functions, this approach was already applied in some different statistics problems and it is known under the name of holonomic gradient method (HGM). For instance, it makes use of holonomic systems of differential equations for evaluating normalization constants of non normalized probability distributions[23] or it gives a method to find maximal likelihood estimates in certain cases[24] . It uses however rather heavy computer

algebra machinery so, for the purpose of this work, we focused on a less sharp technique, by doing pen-and-paper differentiation, since our problem depends only on two parameters. The preliminary results presented in the next section solve Problem 7 with the `fmincon` routine of Matlab©, using an interior point method to show its feasibility. Nevertheless, the final objective is to go beyond these preliminary results and to provide an in-house method based on the HGM approach.

4 Numerical Tests

Table 1 compares the results obtained with the optimization procedure `fmincon` (with the default options: interior point algorithm, with optimality tolerance of $1e-6$ and step tolerance $1e-10$) of Matlab©2016a with the available CARA SDK code, on 5 test cases extracted from the CARA Analysis Tools package[1] (one of the 6 cases tested in their code is not publicly available). Firstly, we consider that only the primary covariance is scaled and report results on rows 4-7 in the table, while only the secondary covariance is scaled for rows 8 to 11. Finally, we suppose that both the primary and secondary covariance matrices are scaled.

In our code, the range for the parameters is fixed to $(k_p, k_s) \in [0.0625, 4] \times [0.0625, 4]$, while the results extracted from running the CARA DilutionMaxPc.m code, are initialized with $(k_p, k_s) \in [0.25, 4] \times [0.25, 4]$ and a tolerance of $1e-6$. Note that in the CARA SDK code, the initial parameter range is not fixed as a constraint, but may be expanded iteratively. We chose to implement the computation of PcMax as a constrained optimization problem, with a larger initial feasible box. Furthermore, the current CARA SDK test suite uses the same coefficient scaling factor for both the primary and secondary to check for dilution.

PcMax, the obtained optimal solutions, as well as number of iterations and functions evaluations is reported for comparison for each optimization problem. Similarly to the CARA code, a case is considered diluted if the covariance scaling is less than 1 and the ratio between the PcMax/Pc is greater than 10^{-3}, which implies that all the tested cases are representative as diluted ones. One can observe that solving the formulated constrained optimization method requires roughly 8 or 9 times less function evaluations and roughly the same number of iterations in order to reach PcMax that are similar to CARA's (with one or 2 significant digits) in all cases except for the *degenerate* case when the optimal solution converges to 0. For this situation, one is likely to obtain a specific routine which identifies and excludes it from the general screening by using the obtained information about the derivatives at zero.

[1] `https://github.com/nasa/CARA_Analysis_Tools`, commit no b9a92b6, updated July 2024.

	Test Cases					
	Case 1		Case 2		Case 3	
	`fmincon`	CARA	`fmincon`	CARA	`fmincon`	CARA
Pc	0.4202		$3.0218e-3$		$2.6515e-3$	
PcMax(k_p)	0.4555	0.4578	$3.0253e-3$	$3.0254e-3$	$2.7357e-3$	$2.7416e-3$
k_p^*	0.0625	$8e-6$	0.0879	$6.75e-4$	0.0627	$8e-6$
Iter	11	10	12	10	15	10
FunEval	27	152	28	152	46	152
PcMax(k_s)	0.7462	0.7905	0.0169	0.6830	0.0162	0.0867
k_s^*	0.0625	$4.16e-7$	0.0625	$1.33e-11$	0.0625	$8.63e-4$
Iter	12	12	14	19	14	8
FunEval	28	184	30	296	30	148
PcMax(k_p, k_s)	0.8957	1	0.01707	0.999	0.01708	0.0909
(k_p^*, k_s^*)	(0.0625, 0.0625)	(0.0048, 0.0048)	(0.06597, 0.0625)	$(1.26, 1.26)e-7$	(0.0629, 0.0625)	$(8.68, 8.68)e-4$
Iter	18	7	23	14	20	9
FunEval	58	132	72	243	68	148

	Case 5		Case 6	
	`fmincon`	CARA	`fmincon`	CARA
Pc	0.0017		0.11325	
PcMax(k_p)	0.0019	0.0019	0.18248	0.1911
k_p^*	0.06254	$1.82e-6$	0.0625	6.4521e-04
Iter	12	11	11	12
FunEval	29	168	31	184
PcMax(k_s)	0.0108	0.02	0.1564	0.1606
k_s^*	0.0625	$2.16e-8$	0.0625	0.0014
Iter	15	14	11	11
FunEval	35	216	29	168
PcMax(k_p, k_s)	0.0213	0.999	0.4789	0.9662
(k_p^*, k_s^*)	(0.0625, 0.0625)	$(4.14, 4.14)e-6$	(0.0625, 0.0625)	(0.0025, 0.0025)
Iter	24	12	19	7
FunEval	79	213	62	131

Table 1: Results for the test cases available in CARA SDK analysis tool. Test number k corresponds to CDMs available in the `SampleCDMs/OmitronTestCase_Test0k*.cdm`.

5 Conclusion

The focus of this paper was on how to efficiently evaluate derivatives of the so-called 2D collision probability Pc seen as a function of parameters which appear linearly in the covariance matrix. We showed that this boils down to the evaluation of some moments of the associated Gaussian measure restricted to the disk. This property allows for the efficient numerical evaluation of the gradient of Pc (possibly higher derivatives like the Hessian matrix), which in turn opens the way for solving associated optimization problems via classical numerical techniques. In particular, we focused on the computation of the so-called maximum collision probability PcMax, when the covariance information for the secondary and/or primary objects at TCA is not reliable. We implemented an optimization-based alternative method for the classical `kp-ks` method, which is usually implemented by a brute-force griding approach in operational software of NASA or CNES. Our preliminary results show the potential of our prototype method and pave the way for a better understanding of the Pc's validity as a measure of conjunction likelihood.

References

[1] J. Foster and H. Estes, "A parametric analysis of orbital debris collision probability and maneuver rate for space vehicles," Tech. Rep. NASA/JSC-25898, NASA Johnson Space Center, August 1992.

[2] M. Akella and K. Alfriend, "Probability of collision between space objects," *Journal of Guidance, Control and Dynamics*, Vol. 23, No. 5, 2000, pp. 769–772. `https://doi.org/10.2514/2.4611`.

[3] R. Patera, "General Method for Calculating Satellite Collision Probability," *Journal of Guidance, Control, and Dynamics*, Vol. 24, July 2001, pp. 716–722. `http://dx.doi.org/10.2514/2.4771`.

[4] S. Alfano, "A Numerical Implementation of Spherical Objet Collision Probability," *Journal of Astronautical Sciences*, Vol. 53, January-March 2005. `https://doi.org/10.1007/BF03546397`.

[5] F. Chan, *Spacecraft Collision Probability*. Reston, Virginia, USA: American Institute of Aeronautics and Astronautics, 2008. `http://dx.doi.org/10.2514/4.989186`.

[6] R. Serra, D. Arzelier, M. Joldes, J. Lasserre, A. Rondepierre, and B. Salvy, "Fast and accurate computation of orbital collision probability for short-term encounters," *Journal of Guidance Control and Dynamics*, Vol. 39, No. 9, 2016, pp. 1009–1021. `https://doi.org/10.2514/1.G001353`.

[7] R. García-Pelayo and J. Hernando-Ayuso, "Series for Collision Probability in Short-Encounter Model," *Journal of Guidance Control and Dynam-*

ics, Vol. 39, No. 8, 2016, pp. 1908–1916. https://doi.org/10.2514/1.G001754.

[8] R. Patera, "Satellite collision probability for nonlinear relative motion," *Journal of Guidance Control and Dynamics*, Vol. 26, No. 5, 2003, pp. 728–733. https://doi.org/10.2514/2.5127.

[9] S. Alfano, "Satellite conjunction Monte Carlo analysis," *Advances in the Astronautical Sciences*, Vol. 134, jan 2009, pp. 2007–2024. https://doi.org/10.1007/BF03546397.

[10] V. Coppola, "Including Velocity Uncertainty in the Probability of Collision between Space Objects," *Advances in the Astronautical Sciences*, Vol. 143, 2012, pp. 2159–2178.

[11] G. Krier, "Satellite Collision Probability for Long-term Encounters and Arbitrary Primary Satellite Shape," Paper presented at the 7th European Conference on Space Debris, April 2017.

[12] K. Alfriend, M. Akella, J. Frisbee, J. Foster, D.-J. Lee, and M. Wilkins, "Probability of collision error analysis," *Space Debris*, Vol. 1, No. 1, 1999, pp. 21–35.

[13] S. Alfano, "Relating position uncertainty to maximum conjunction probability," *The Journal of the Astronautical Sciences*, Vol. 53, 2005, pp. 193–205.

[14] J. Frisbee, "An upper bound on high speed satellite collision probability when only one object has position uncertainty information," *Proceedings of the AAS/AIAA Astrodynamics Specialists Conference*, No. AAS 15-717, Vail, CO, USA, July 2015.

[15] M. D. Hejduk, D. Snow, and L. Newman, "Satellite conjunction assessment risk analysis for "dilution region" events: issues and operational approaches," *Space Traffic Management Conference*, 2019.

[16] M. S. Balch, R. Martin, and S. Ferson, "Satellite conjunction analysis and the false confidence theorem," *Proceedings of the Royal Society A*, Vol. 475, No. 2227, 2019, p. 20180565.

[17] F. Laporte, "JAC software, solving conjunction assessment issues," *AMOS - Advanced Maui Optical and Space Surveillance Technologies Conference*, Maui, Hawai'i, USA, 2014, p. E4.

[18] S. Laurens, P. Seimandi, J. Couetdic, and J. Dolado, "Covariance Matrix Uncertainty Analysis and Correction," *7th European Conference on Space Debris*, Darmstadt, Germany, April 2017, pp. 46–49.

[19] I. Urdampilleta, E. Delande, V. Morand, J. Gelhaus, E. Vellutini, V. Poenaru, J. Freitas, T. Zubowicz, and D. Garcia-Yarnoz, "System Approach to Analyse the Performance of the current and future EU Space Surveillance and Tracking system at Service Provision level," *AMOS2022 - Advanced Maui Optical and Space Surveillance Technologies Conference*, Maui, Hawai'i, USA, 2022.

[20] S. Alfano and D. Oltrogge, "Probability of Collision: Valuation, variability, visualization, and validity," *Acta Astronautica*, Vol. 148, 2018, pp. 301–316, https://doi.org/10.1016/j.actaastro.2018.04.023.

[21] F. Krage, "Nasa spacecraft conjunction assessment and collision avoidance best practices handbook," tech. rep., NASA, 2023, NASA/SP-20230002470 Rev 1.

[22] M. Masson, D. Arzelier, F. Bréhard, M. Joldes, and B. Salvy, "Fast and Reliable Computation of Instantaneous Orbital Collision Probability," *Journal of Guidance, Control, and Dynamics*, 2024, pp. 1–14.

[23] H. Nakayama, K. Nishiyama, M. Noro, K. Ohara, T. Sei, N. Takayama, and A. Takemura, "Holonomic gradient descent and its application to the Fisher–Bingham integral," *Advances in Applied Mathematics*, Vol. 47, No. 3, 2011, pp. 639–658, https://doi.org/10.1016/j.aam.2011.03.001.

[24] J. Hayakawa and A. Takemura, "Estimation of exponential-polynomial distribution by holonomic gradient descent," *Communications in Statistics-Theory and Methods*, Vol. 45, No. 23, 2016, pp. 6860–6882.