Passively-Safe and Robust Multi-Agent Optimal Control with Application to Distributed Space Systems

Tommaso Guffanti * and Simone D’Amico †‡
Stanford University, Stanford, California 94305

This paper presents a novel methodology to enforce motion safety guarantees even in the event of a sudden loss of control capabilities by any agent within a multi-agent system. This passive safety methodology permits the replacement of point-evaluated constraints on the trajectory following a contingency with a single function of the integration constants of the equations of motion at contingency instants. The effects of uncertainties on the trajectory are compensated for through the method of variation of parameters. This permits a reduction in the number of required constraints by one polynomial degree in the number of discrete time samples, and allows computationally efficient enforcement of passive safety within a multi-agent optimal control problem, solvable using direct methods. The main application is distributed space systems employing miniaturized low-size-weight-and-power and commercial-off-the-shelf technology, which reduce mission financial costs at the expense of reliability. Experimental results for the upcoming Virtual Super-resolution Optics with Reconfigurable Swarms (VISORS) mission, as well as complementary formation-flying test cases in eccentric orbits, demonstrate the advantages of the proposed methodology in achieving fault-tolerant safety guarantees, as well as computational efficiency.

I. Introduction

Distributed space systems (DSS) demand high levels of safety, robustness and fault-tolerance. This risk-aversion is mainly due to the potential consequences a collision in space may have in terms of debris generation and investment losses. In recent years, the trends of spacecraft bus miniaturization and use of low-size-weight-and-power (low-SWaP) as well as commercial-off-the-shelf (COTS) technologies have fostered the distribution of payload and tasks among multiple coordinated agents enabling functionalities that are otherwise not achievable by single monolithic systems [1]. On the other hand, these increased the likelihood of contingencies that directly or indirectly cause permanent or temporary inability to control a spacecraft [2]. Fault-tolerant strategies are therefore fundamental to mitigate the consequences and keep the risks associated to flying these systems within tolerable levels [3]. From a collision safety perspective, different strategies are possible, including: reactive [4], proactive [5], and passive [6–9]. In general terms, reactive means that spacecraft have to re-compute trajectories on-line to avoid collision after a contingency. Proactive can be intended as a variant of reactive, in which the availability of a controlled escape trajectory/maneuver has to be guaranteed at any time even in the presence of a degradation, but not complete loss, of control capabilities. Finally, passive means that the controlled trajectories have been pre-designed to guarantee safe separation even in the event of sudden loss of control capabilities, being it temporary or permanent. To fly a real space mission, possibly all of these strategies are needed, and must be implemented within the guidance and control pipeline. Nevertheless, relying exclusively on a reactive or even proactive approach has limitations, such as not being robust to contingencies that do not permit trajectory re-planning and/or control on the short term of one or multiple spacecraft. These types of contingencies (e.g., sudden bus safe modes, sudden temporary or permanent failure of critical sub-systems) is increasingly common on miniaturized buses employing low-SWAP and COTS components [2], and demand for a passively-safe approach to fault-tolerant motion safety. Looking at the literature, examples of reactive and proactive collision avoidance strategies are relatively recent [4,5]. Whereas, the first examples of passively-safe strategies date back to the 60’s with the co-elliptic rendezvous of the Apollo missions [10], and to the 80’s with the colocation of geostationary satellites in shared longitude slots [6]. Subsequently, in the last decades, passive safety (PS) became an enabler of the first binary formation-flying missions in low-Earth orbit such as GRACE, TANDEM-X, PRISMA and BIROS [11]. Where the relative orbits were designed such that safe separation was always guaranteed in the radial/normal orbital directions, irrespective of failures of...
controlling and/or estimating the separation in the along-track direction \[7\]. This work has been then generalized to \(N\)-spacecraft swarms in near-circular orbits \[12\]. The limitations of this literature are multiple. Firstly, except few cases in the close-proximity operations literature \[5, 8\], fault-tolerant motion safety strategies are explored at orbit design level \[7, 9, 12\] and not as a formal constraint within the optimal control problem (OCP). Secondly, PS guarantees are given under assumptions on the orbit type and dynamics, e.g., circularity \[5, 8\] or near-circularity \[7, 12\], and linearized two-body dynamics \[5, 8\], or including just specific perturbations \[12\]. Thirdly, PS is enforced through possibly over-conservative constraints on the relative orbit design, e.g., energy matching conditions, sub-state designs \[7, 12\]. In general, these limitations cause the current state-of-the-art to fall short in addressing the needs required by future DSS missions, both in terms of applicability, breadth and robustness.

This paper explores the mathematical method of variation of parameters (VoP) \[13, 14\] to present a novel methodology for the modeling of dynamics-dependent constraints with application to fault-tolerant control, and to enforce them efficiently within an OCP solvable through direct methods. Specifically, VoP is a general method for the solution of both linear and nonlinear ordinary differential equations (ODE) \[13\], which models the effects of non-integrable dynamics on the integration constants (IC) of an integrable portion of the governing dynamics itself. VoP was firstly developed for the orbital dynamics \[14\] under the form of the Gauss variational equations (GVE). GVE model the effect of non-integrable orbital perturbations and control accelerations on the Keplerian orbital elements (OE), which are IC of the homogeneous part of the orbital ODE. Similarly, the IC of the homogeneous part of the ODE governing the relative orbital dynamics linearized for small spacecraft separations are first-order equivalent to the relative orbital elements (ROE), combination of the OE of two spacecraft or a reference orbit and a spacecraft. This holds in both near-circular \[15, 16\] and eccentric orbits \[17, 18\] by tuning appropriately the ROE definition. Using VoP and isolating the effect of non-integrable dynamics on the IC state has various advantages. Firstly, it fosters the formulation of highly efficient analytical closed-form dynamics models \[15, 22\]. Secondly, it permits direct compensation of the effects of non-integrable dynamics in guidance and control strategies \[12, 23\]. Thirdly, it permits direct specification of the "long-term" behavior of the natural motion of the system, which is directly connected to the value of the IC themselves. This is useful when constraints such as collision avoidance have to be enforced over a specified time horizon even in absence of applied control input. This is evident by looking at the DSS literature, where PS is easily obtained by properly selecting the OE or ROE of two resident space objects, such as for co-elliptic rendezvous \[9\], colocation \[6\], and formation-flying \[7\]. In addition, given the applicability of VoP to a wider range of dynamical systems \[13\], the advantages brought by its use are possibly generalizable.

From an algorithmic perspective, this paper explores the optimization and robotics literature in five topics of motion planning and control. The first topic is open-loop trajectory optimization under state and control constraints. In particular, this paper uses sequential convex programming (SCP), which is a local direct optimization method that seeks convergence to a local stationary point of a non-convex problem by successively convexifying the non-convex cost and constraints and solving a series of convexified problems \[24, 26\]. The second topic is robust motion planning, to achieve probabilistic guarantees of constraints satisfaction in the presence of system uncertainties. Usually the literature seeks either probabilistic guarantees of safety for a stochastic model of uncertainty or considers a bounded model of uncertainty \[27\]. This paper models the uncertainty as linearized Gaussian \[28\], uses the Kalman filter equations to propagate the error covariance matrix through the dynamics linearized around the mean estimate, and computes safety margins by using the relevant diagonal components. Moreover, the effect of applied control on the uncertainty propagation is explicitly taken into account, being relevant to model the actuation uncertainty present, for example, on spacecraft thrusters. The third topic is model predictive control (MPC) \[29\], to retrieve closed-loop performances on-line. In particular, in this paper the open-loop robust motion planner is used within a MPC pipeline to retrieve closed-loop performances on-line in the presence of unmodeled or mismodeled effects. The fourth topic is multi-agent control, to deploy algorithms on multi-vehicle systems. In general, the computational complexity of checking separation between any pair of agents within an SCP setting scales quadratically with their number \[30\], this makes the solutions of the centralized multi-agent collision avoidance problem computationally intense for large number of agents. Possible mitigation strategies are to limit the collision checks to a specified subset of neighboring agents \[30\], or to decouple the problem in a centralized open-loop planning and a decentralized closed-loop tracking that keeps the agents within specified tracking error bounds \[27\]. In this paper the latter approach is followed. Finally, fault-tolerance is a topic largely addressed in automatic control literature, assuming diverse connotations in different application fields \[31\]. Looking at the multi-agent control literature specifically, consensus schemes \[32\] are often used, and fault-tolerance is not included as a formal constraint within the optimization problem. Therefore, this paper addresses the relevant, and not exhaustively explored, problem of efficiently and explicitly enforcing fault-tolerant motion safety guarantees within a multi-agent OCP solvable through direct methods.
This paper builds over the previous work of the authors [33-36] to provide five main contributions to the state-of-the-art. Firstly, in Section II it presents the novel methodology for the modeling of dynamics-dependent constraints with application to fault-tolerant control, in the presence of nonlinear non-integrable dynamics and realistic system uncertainties from sensing, actuation, and unmodeled system dynamics. This methodology permits replacing point-evaluated constraints on the trajectory following a contingency with a single function of the IC at contingency instants, while compensating for the effects of uncertainties through VoP. Secondly, in Section III it demonstrates that the number of constraints required to guarantee multi-agent PS is polynomial (super-linear) in the number of time samples, and that the proposed methodology reduces them by one polynomial degree. Thirdly, in Section IV it shows how this methodology can be used within a sequential optimization framework for the solution of the multi-agent passively-safe OCP. This includes a SCP-based centralized algorithm that enforces guarantees of multi-agent PS, and a decentralized tracking logic that reinforces them within an MPC-like closed-loop pipeline. Fourthly, in Section V it specializes the methodology to the DSS dynamics, and it derives novel closed-form solutions of PS in eccentric orbits, which can be used both for relative orbit design and within a constrained OCP. Finally, in Section VI it applies the algorithmic framework to the Virtual Super-resolution Optics with Reconfigurable Swarms (VISORS) mission [37], a first-of-a-kind nanosatellite segmented telescope due to launch in 2024 for high-resolution imaging of the solar corona, and to complementary formation-flying test cases in eccentric orbits. These test cases demonstrate the advantages of the proposed methodology in achieving fault-tolerant safety guarantees, as well as computational efficiency. Moreover, a comparison with closed-form impulsive control solutions with flight-heritage [38] shows the potential of the proposed algorithmic framework in terms of gained control accuracy and fuel-efficiency, while enforcing the desired safety guarantees in closed-loop.

II. The Integration Constants Approach

A. Mathematical Preliminaries

Consider a system characterized by an s-dimensional operational state $\chi(t) \in \mathbb{R}^s$, where $t \in \mathbb{R}_{\geq 0}$ is the independent variable here identified with time. The ordinary differential equation (ODE) that governs the operational state is assumed nonlinear control-affine as

$$\dot{\chi}(t) = f(\chi(t)) + B(\chi(t))u(t)$$  \hspace{1cm} (1)$$

with $f : \mathbb{R}^s \rightarrow \mathbb{R}^s$ the drift vector term, $u(t) \in \mathbb{R}^r$ the r-dimensional applied control input, and $B : \mathbb{R}^s \rightarrow \mathbb{R}^{s \times r}$ the control input matrix. Both $f$ and $B$ are assumed continuously differentiable. Moreover, assume the drift vector term in Eq. (1) can be partitioned as

$$\dot{\chi}(t) = \iota(\chi(t)) + d(\chi(t)) + B(\chi(t))u(t)$$  \hspace{1cm} (2)$$

where, the $\iota : \mathbb{R}^s \rightarrow \mathbb{R}^s$, is a reduced part of the governing ODE integrable analytically in closed-form, and $d : \mathbb{R}^s \rightarrow \mathbb{R}^s$ is the part of the governing ODE that makes it non-integrable. The reduced part $\dot{\chi}(t) = \iota(\chi(t))$ can be integrated analytically starting at instant $t_i$ as

$$\chi(t) = \Psi(t, c(t_i))$$  \hspace{1cm} (3)$$

where, $c \in \mathbb{R}^s$ is the integration constants (IC) state, and $\Psi : \mathbb{R}^{s+1} \rightarrow \mathbb{R}^s$ is the integrated dynamics map defined in closed-form upon integration, and is here assumed to be continuous, differentiable, invertible (with the inverse defined as $\Psi^{-1}(t, \chi(t)) : \mathbb{R}^{s+1} \rightarrow \mathbb{R}^s$), and possible ambiguities for multivalued functions can be discriminated. The IC state is guaranteed to exist as long as Eq. (2) holds, and its components are an independent linear or nonlinear combination of the operational state components at the integration instant $\chi(t_i)$ (i.e., the initial conditions), and possibly depend on $t_i$ itself. The IC can have a clear physical meaning (e.g., conserved energy or angular momentum), or, more simply, they can be the initial conditions of the system at integration instant. If the integrable dynamics is nonlinear, the map $\Psi$ is generally a nonlinear function of the IC. Therefore, its Jacobian $\mathbf{\Psi}(t, c) = \partial_0 \Psi(t, c)/\partial_0 c : \mathbb{R}^{s+1} \rightarrow \mathbb{R}^{s \times s}$ is a function of the IC, too. On the contrary, if the integrable dynamics is linear or linearized (i.e., $\iota(\chi(t)) \approx A(t)\chi(t)$, with $A \in \mathbb{R}^{s \times s}$), the map $\Psi$ becomes linear with respect to the IC as $\Psi(t, c) = \mathbf{\Psi}(t) c$, where now its Jacobian coincides with the fundamental matrix $\mathbf{\Psi}(t) \in \mathbb{R}^{s \times s}$ solution of the linear ODE and just dependent on time. Note that the non-integrable part may include truncated higher-order terms if a linearization is performed to isolate a linear integrable part.

Variation of parameters (VoP) [13] is a mathematical tool that permits the modelling of the effect of non-integrable dynamics and control on the IC. In particular, Eq. (2) can be rewritten exactly as an ODE of IC variation as

$$\dot{c}(t) = d_c(t, c(t)) + B_c(t, c(t))u(t)$$  \hspace{1cm} (4)$$

where, the drift vector is defined as $d_c(t, c(t)) = \Psi^{-1}(t, c(t)) \dot{d}(c(t)) : \mathbb{R}^{s+1} \rightarrow \mathbb{R}^s$, whereas control input matrix is defined as $B_c(t, c(t)) = \Psi^{-1}(t, c(t)) B(c(t)) : \mathbb{R}^{s+1} \rightarrow \mathbb{R}^{s \times r}$. Eq. 4 shows how both non-integrable dynamics and control have a direct effect on the IC state. On the other hand, they do not alter the structure of the map $\Psi$. In particular, the motion trajectory of $\chi(t)$ governed by Eq. 2 is instantaneously contained and tangent to $\psi(t, c(t))$, such that $\chi(t) = \psi(t, c(t))$, with the evolution in time of $c(t)$ governed by Eq. 4. From a topological perspective, $\psi(t, c(t))$ describes a submanifold of $\mathbb{R}^s$ on which $\chi(t)$ lies instantaneously. This idea is known as the “osculation principle” [14].

As mentioned in the introduction and further detailed in Section V, the assumption of integrability of the dynamics after a contingency, this permits rewriting the sufficient condition in Eq. 6 as

$$M(\chi(t)) \leq 0, \quad \forall t \in [t_i, t_i + T]$$

such that: $\chi(t) = \psi(t, c(t)) + B(\chi(t)) u(t)$

AND

$$M(\chi(t)) \leq 0, \quad \forall t \in [t_i, t_i + T], \quad \forall t_i \in [0, t_f]$$

such that: $\dot{\chi}(t) = B(\chi(t)) u(t)$

In a discrete time setting, if the interval $[0, t_f]$ is discretized in $n$ samples, and the intervals $(t_i, t_i + T)$ in $m$ samples, and if contingencies are accounted for at each of the $n$ samples, the total number of constraints to be enforced is $nm$. In the following, the IC representation and VoP are used to reduce this required number of constraints.

To do so, assume an instant of worst case occurrence of $M$ exists within $[t_i, t_i + T]$. In such a way, the satisfaction of the constraint at this instant is a sufficient condition for the satisfaction of the constraint at any other instant within the interval. Mathematically, assuming $M$ continuously differentiable,

$$\exists \tau^* \in [t_i, t_i + T]: \{M(\chi(t^*)) \leq 0\} \Rightarrow \{M(\chi(t)) \leq 0, \quad \forall t \in [t_i, t_i + T]\}$$

The instant of worst case occurrence is defined as

$$t^* = \arg \max_{t \in [t_i, t_i + T]} M(\chi(t))$$

Moreover, assume preliminarily that the non-integrable part of the dynamics is set to zero after a contingency together with the control. Under this assumption, the dynamics after a contingency is fully integrable and, therefore it is expressible in closed-form using Eq. 3 $\forall t \in [t_i, t_i + T]$. Under this assumption, Eq. 7 can be rewritten as

$$t^* = \arg \max_{t \in [t_i, t_i + T]} M(\psi(t, c(t_i)))$$

where, by definition, the IC state $c(t_i)$ is a constant linear or nonlinear combination of $\chi(t_i)$ and possibly $t_i$ itself. Note that in Eq. 4 $M$ is a closed-form expression of time, which is the optimization variable, whereas the IC state acts as a parameter vector in the optimization. $M$ is a generic non-convex function of the state, and in general, $\psi$ is a non-convex function of time. Therefore, Eq. 8 is a non-convex parametric optimization problem, in the parameter vector $c(t_i)$. By definition, the solution of a parametric optimization problem is a function of the parameters themselves. Therefore, by definition, $t^*$ is a function of the IC. This function is assumed identifiable as $t^* = t^*(c(t_i)) : \mathbb{R}^s \rightarrow \mathbb{R}^{\geq 0}$. Under the assumption of integrability of the dynamics after a contingency, this permits rewriting the sufficient condition in Eq. 6 as

$$M(\chi(t^*)) = M(\psi(t^*(c(t_i)), c(t_i))) = M^*(c(t_i)) \leq 0$$

B. Efficient Modeling of Dynamics-Dependent Constraints with Application to Fault-Tolerant Control

In this section, the advantage brought by the IC representation regarding the modeling of dynamics-dependent constraints with application to fault-tolerant control is presented in general terms. Then, in Section III, it is applied specifically to the modeling of multi-agent passive safety (PS), and, in Section IV, it is specialized to the distributed space systems (DSS) dynamics. From a fault-tolerant perspective, there is interest in providing guarantees of constraints satisfaction even in the event of a sudden loss of control capability. In particular, consider the state-dependent scalar constraint $M(\chi(t)) \leq 0$, where $M : \mathbb{R}^s \rightarrow \mathbb{R}$ is a generic non-convex scalar function of the state, assumed continuously differentiable. The constraint function is assumed identifiable as $M(\chi(t))$ exists within $\mathbb{R}^s$. In particular, consider the state-dependent scalar constraint $M(\chi(t)) \leq 0$, where $M : \mathbb{R}^s \rightarrow \mathbb{R}$ is a generic non-convex scalar function of the state, assumed continuously differentiable. The constraint function is assumed identifiable as $M(\chi(t))$ exists within $\mathbb{R}^s$. In such a way, the satisfaction of the constraint at this instant is a sufficient condition for the satisfaction of the constraint at any other instant within the interval. Mathematically, assuming $M$ continuously differentiable,

$$\exists \tau^* \in [t_i, t_i + T]: \{M(\chi(t^*)) \leq 0\} \Rightarrow \{M(\chi(t)) \leq 0, \quad \forall t \in [t_i, t_i + T]\}$$

The instant of worst case occurrence is defined as

$$t^* = \arg \max_{t \in [t_i, t_i + T]} M(\chi(t))$$

Moreover, assume preliminarily that the non-integrable part of the dynamics is set to zero after a contingency together with the control. Under this assumption, the dynamics after a contingency is fully integrable and, therefore it is expressible in closed-form using Eq. 3 $\forall t \in [t_i, t_i + T]$. Under this assumption, Eq. 7 can be rewritten as

$$t^* = \arg \max_{t \in [t_i, t_i + T]} M(\psi(t, c(t_i)))$$

where, by definition, the IC state $c(t_i)$ is a constant linear or nonlinear combination of $\chi(t_i)$ and possibly $t_i$ itself. Note that in Eq. 4 $M$ is a closed-form expression of time, which is the optimization variable, whereas the IC state acts as a parameter vector in the optimization. $M$ is a generic non-convex function of the state, and in general, $\psi$ is a non-convex function of time. Therefore, Eq. 8 is a non-convex parametric optimization problem, in the parameter vector $c(t_i)$. By definition, the solution of a parametric optimization problem is a function of the parameters themselves. Therefore, by definition, $t^*$ is a function of the IC. This function is assumed identifiable as $t^* = t^*(c(t_i)) : \mathbb{R}^s \rightarrow \mathbb{R}^{\geq 0}$. Under the assumption of integrability of the dynamics after a contingency, this permits rewriting the sufficient condition in Eq. 6 as

$$M(\chi(t^*)) = M(\psi(t^*(c(t_i)), c(t_i))) = M^*(c(t_i)) \leq 0$$
which makes the constraint a function of the IC state evaluated at the contingency instant independent of the time along the uncontrolled trajectory: \( \mathcal{M}'(c(t_i)) : \mathbb{R}^s \to \mathbb{R} \). By removing the dependency on the time along the uncontrolled trajectories, fault-tolerant constraint satisfaction is achieved for at least \( T \) after a contingency by enforcing as few constraints as the number of possible contingency instants. In particular, enforcing Eq. (9) in place of Eq. (5) implies a reduction of the number of constraints by one polynomial degree in the number of discrete time samples, from \( nm \) to \( n \).

In Eq. (9) the formalization of the function \( \mathcal{M}' \) assumes the capability of identifying the function \( t' \cdot c(t_i) \). In this paper, a general algorithmic procedure is proposed in Section IV, moreover, closed-form solutions applied to DSS PS are presented in Section V. Furthermore, Eq. (9) assumes the integrability of the dynamics after a contingency. Nevertheless, VoP can be used to make it robust to non-integrable dynamics effects. This is done by bounding the integrated effect of non-integrable dynamics on \( \mathcal{M}' \) over \( [t_i, t_i + T] \) as

\[
\left. \frac{\partial \mathcal{M}'(c)}{\partial e} \right| e(t_i) \int_{t_i}^{t_i+T} \mathcal{M}'(\tau, c(\tau)) d\tau \leq \beta(t_i, T) \tag{10}
\]

In Eq. (10) \( \frac{\partial \mathcal{M}'}{\partial e} \in \mathbb{R}^s \) is a Jacobian vector mapping the integrated effect of non-integrable dynamics to a net variation of \( \mathcal{M}' \), whereas, \( \beta \in \mathbb{R}_{>0} \) is a bounding constant. This constant is added to Eq. (9) as

\[
\mathcal{M}'(c(t_i)) + \beta(t_i, T) \leq 0 \tag{11}
\]

to compensate for non-integrable dynamics effects on constraint satisfaction. Eq. (10) represents a first-order sensitivity model, which, depending on the nature of the function \( \mathcal{M}'(c) \), may lose accuracy the further away from the neighborhood of \( c(t_i) \), i.e., the larger the integrated effect of non-integrable dynamics on the IC state gets over \( [t_i, t_i + T] \). If needed, this first-order model can be upgraded to a higher-order one, without impacting the proposed constraint reformulation advantage. As analyzed in Section V, the proposed first-order model is sufficiently accurate for the DSS PS application. In the following, the deterministic analysis developed in this section is generalized to account for bounded effects of uncertainties on the constraint satisfaction.

C. Inclusion of Uncertainties

Consider now a closed-loop control setting where the system has to find a control plan \( \tilde{u}(t) \in \mathbb{R}^r \), for \( t \in [t_0, t_f] \), to move from an initial configuration at \( t_0 \) to a target one at \( t_f \), acquiring at instants \( t_e \in [t_0, t_f] \) an updated estimate of its IC state \( \hat{c}(t_e) \in \mathbb{R}^r \). This estimate is the mean of a Gaussian distribution \( \mathcal{N}(\hat{c}(t_e), \mathcal{C}(t_e)) \), provided by on-board sensing and navigation, where \( \mathcal{C}(t_e) \in \mathbb{R}^{xs} \) is the error covariance matrix. Note that, depending on the estimation filter implementation, the Gaussian distribution may be provided on the operational state \( \mathcal{N}(\hat{x}(t_e), X(t_e)) \), with \( \hat{x}(t_e) \in \mathbb{R}^r \) and \( X(t_e) \in \mathbb{R}^{xs} \), and then mapped to the IC state inverting the integrated dynamics closed-form map as \( \hat{c}(t_e) = \Psi^{-1}(t_e, \hat{x}(t_e)) \) and \( \mathcal{C}(t_e) = \Psi^{-1}(t_e, \hat{c}(t_e))X(t_e)\Psi^{-1}(t_e, \hat{c}(t_e)) \), where \( (\cdot)^T \) means transpose. Moreover, assume the system has available on-board a nonlinear dynamics model, \( d_c : \mathbb{R}^{sx+1} \to \mathbb{R}^s \), approximating the real effect of non-integrable dynamics on the IC. Using this model and VoP, the system propagates on-board the estimated mean IC state as

\[
\hat{c}(t) = \tilde{d}_c(t, \hat{c}(t)) + B_c(t, \hat{c}(t))\tilde{u}(t) \tag{12}
\]

Given this on-board dynamics, assume the system uses a linearized approach of propagation of uncertainties. The evolution in time of the Gaussian approximation of the uncertainty distribution is modeled through a linear ODE governing the state error covariance matrix \( \Psi \)

\[
\dot{\Psi}(t) = \tilde{D}_c(t, \hat{c}(t), \tilde{u}(t))\Psi(t) + C(t)\mathcal{M}'(t, \hat{c}(t), \tilde{u}(t)) + B_c(t, \hat{c}(t))U(t)B_c^T(t, \hat{c}(t)) + Q(t) + E(t) \tag{13}
\]

which holds accurately in the neighborhood of the mean state estimate. In particular, \( \tilde{D}_c = \frac{\partial \tilde{d}_c}{\partial \hat{c}}|_e + \frac{\partial B_c}{\partial \hat{c}}|_e \tilde{u} : \mathbb{R}^{sx+1} \to \mathbb{R}^{xs} \) is a Jacobian matrix evaluated at the mean state estimate. \( Q(t) \in \mathbb{R}^{sx, sx} \) is the power spectral density of a zero-mean white Gaussian process-noise modeling the discrepancy between the real non-integrable dynamics and the available approximation: \( \tilde{d}_c(t, \hat{c}(t)) - d_c(t, c(t)) \). The process noise power spectral density and covariance may be time-dependent, but no time correlation is assumed to exist. \( U(t) \in \mathbb{R}^{rxr} \) is the power spectral density of a zero-mean white Gaussian noise modeling the discrepancy introduced by the actuation system between the planned control input and the real applied control input: \( \hat{u}(t) - u(t) \). Also in this case no time correlation is assumed to exist. On the other hand, the relation between planned control input and associated execution uncertainty is modeled as:

\[
U(t) = \mathcal{W}(\tilde{u}(t)) + \tilde{U}(t) \quad \text{where} \quad \mathcal{W} : \mathbb{R}^r \to \mathbb{R}^{rxr} \quad \text{and} \quad \tilde{U}(t) \in \mathbb{R}^{rxr} \quad \text{is a remaining aleatory component independent} \]
of the planned control. \( E(t) = \delta(t - t_e)E(C(t_e)) \) is an impulsive function modeling the effect of future expected measurements on the state error covariance, where the Dirac-delta equals one at the future predicted measurements instants \( t_e \), and \( E : \mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n_x} \) is a function of the predicted future estimated state covariance \( C(t_e) \). Modeling the effect to future measurements on the covariance is important, since it informs the controller of the expected closed-loop behavior of the covariance even when it has to solve for an open-loop plan. This prevents possible unfeasibility of the open-loop control solution, due to the growth of the uncertainty propagated in open-loop.

1. Robustification of Fault-Tolerant Constraint Formulation

It is of interest to make Eq. [1] applicable in this real closed-loop control setting, robustifying it against the modeled uncertainties evaluated at a specified \( q-\sigma \) confidence level (with \( q \in \mathbb{N}_{>0} \) and \( \sigma \) being a statistical standard deviation). To do so, the integrated effect of non-integrable dynamics and uncertainty on \( M^* \) over \([t_i, t_i + T]\) are bounded using VoP as

\[
\left| \frac{\partial M^*(e)}{\partial e} \right| \int_{t_i}^{t_i+T} \hat{d}_e(\tau, \hat{e}(\tau))d\tau + \left| \frac{\partial M^*(e)}{\partial e} \right| q \sqrt{\text{Diag} \left( \int_{t_i}^{t_i+T} \left( D_e(\tau, \hat{e}(\tau))C(\tau) + C(\tau)\hat{D}_e^T(\tau, \hat{e}(\tau)) + Q(\tau) \right) d\tau + C(t_i) \right)} \leq \beta(t_i, T, q) \tag{14}
\]

where \( \frac{\partial M^*}{\partial e} \in \mathbb{R}^x \) is a Jacobian vector, the \( \text{Diag}(.) \) operator extracts the diagonal vector from a square matrix, the absolute value \( |.| \) is applied element-wise for vector quantities, \( q \in \mathbb{N}_{>0} \) models the \( q-\sigma \) uncertainty level considered, and \( \hat{e}(t_i) \) and \( C(t_i) \) are obtained by propagating Eq. [12][13] up to the contingency instant \( t_i \). In Eq. [14] \( \beta \in \mathbb{R}_{>0} \) is an enlarged bounding constant with respect to the one defined in Eq. [10], which allows the compensation of both non-integrable dynamics and uncertainty effects on the fault-tolerant constraint satisfaction, as

\[
M^*(\hat{e}(t_i)) + \beta(t_i, T, q) \leq 0 \tag{15}
\]

where now the function \( M^* \) is evaluated at the mean IC state estimate. Eq. [14] represents a first-order sensitivity model, which, depending on the nature of the function \( M^*(e) \), may lose accuracy the further away from the neighborhood of \( c(t_i) \), i.e., the larger the integrated effect of non-integrable dynamics and uncertainty on the IC state gets over \([t_i, t_i + T]\). If needed, this first-order model can be upgraded to a higher-order one, without impacting the proposed constraint reformulation advantage. As analyzed in Section [VI], the proposed first-order model is sufficiently accurate for the DSS PS application.

D. Efficient Formulation of the Fault-Tolerant Optimal Control Problem

The optimal control problem (OCP) the system must solve in closed-loop, possibly at every instant \( t_e \in [t_0, t_f] \) an updated state estimate is available, until a specified target set \( C_f \) is reached at \( t_f \), is

\[
\begin{align*}
\text{minimize} & \quad \int_{t_e}^{t_f} J(\dot{\hat{e}}(t), C(t), \hat{u}(t), t)dt \\
\text{subject to} & \quad \dot{\hat{e}}(t) = \hat{d}_e(t, \hat{e}(t)) + B_e(t, \hat{e}(t))\hat{u}(t) \\
& \quad \hat{C}(t) = \hat{D}_e(t, \hat{e}(t), \hat{u}(t))C(t) + C(t)\hat{D}_e^T(t, \hat{e}(t), \hat{u}(t)) + B_e(t, \hat{e}(t))U(t)B_e^T(t, \hat{e}(t), \hat{u}(t)) + Q(t) + E(t) \\
& \quad M^*(\hat{e}(t)) + \beta(t, T, q) \leq 0 \\
& \quad \hat{u}(t) \in U \\
& \quad \hat{e}(t_f) \in C_f \\
& \quad N(\hat{e}(t_e), C(t_e)) \equiv \text{state estimate}
\end{align*}
\]

Where, \( J \in \mathbb{R}_{>0} \) is a generic cost functional modeling for example control input and/or time minimization, \( U \) is the set of admissible control inputs, and the bound \( \beta(t_i, T, q) \) is defined as in Eq. [14]. The control horizon progressively contracts from \([t_0, t_f]\) to zero, while \( t_e \rightarrow t_f \). This represents a generic MPC-like implementation that permits the retrieval of closed-loop properties by recomputing the control plan when an updated state estimate is available. In the
last line of Eq. 16, the state estimate available at instant \( t_e \) is the initial condition of the optimization problem solved at that instant. The boundary condition at \( t_f \) can be enforced either deterministically on the mean state, or probabilistically on the distribution \( \mathcal{N}(\hat{e}(t_f), C(t_f)) \). In the second to last line in Eq. 16, the former approach is followed. The OCP in Eq. 16 enforces fault-tolerant constraint satisfaction for at least \( T \) after a contingency, at \( q-\sigma \) confidence, at the cost of checking as many constraints as the number of possible contingency instants in \([t_e, t_f]\). This implies a reduction of the number of constraints by one polynomial degree in the number of discrete time samples with respect to the formulation in Eq. 5 (i.e., from \( nm \) to \( n \)). Eq. 16 is a non-convex optimization problem that can be solved, for example, using direct methods, as discussed in Section IV.

III. Multi-Agent Passively-Safe Optimal Control

In this section, the developed methodology is applied to the constraint of multi-agent passive safety (PS), which is defined in this paper as follows.

Definition 1 – (Passive safety on \([t_0, t_f]\), for at least \( T \), of at least \( \epsilon \), at \( q-\sigma \) confidence): Along the trajectory towards the target condition, a multi-agent system is passively-safe if all the agents maintain a separation of at least \( \epsilon \), accounting for \( q-\sigma \) uncertainties, even in the event that any agent loses control at any instant \( t \in [t_0, t_f) \) without possibility of re-planning until \( t_f + T \).

Note that, in principle, it is possible to distinguish between different PS classes depending on the type of failure causing the loss of control capabilities. For example, it is possible to distinguish between hard failures, which prevent re-planning capabilities on the short term, or temporary failures, which allow re-planning capabilities on the short term. Furthermore, in the multi-agent setting, it is possible to just account for the loss of control of a specified subset of agents. PS as in Definition 1, is a rather conservative class, in which any subset or combination of agents may lose control at any instant in \([t_0, t_f]\), without being able to recover it and re-plan to avoid collision until \( t_f + T \). The intention is to set the bar for milder PS classes deducible by down-grading the one considered in this paper. Note that Definition 1 directly accounts for the presence of possible "permanently passive" agents, whose trajectory is uncontrolled since \( t_0 \). In a distributed space system, these may be a subset of spacecraft orbiting passively. Moreover, the presented definition relies on the capability to retrieve an estimate of active and permanently passive agents states through sensing, navigation and prediction. A graphical comparison between this PS class and the standard collision avoidance is presented in Figure 1. The controlled trajectories of the active agents are in solid lines, the uncontrolled ones in dashed lines, the shaded blue ellipses represent the uncertainty distributions of the Cartesian positions evaluated at \( q-\sigma \) confidence interval.

![Fig. 1 Comparison between standard collision avoidance and passive safety](image)

In order to describe PS mathematically, consider a multi-agent system in the Cartesian space, composed of \( N \) agents, of which \( N_a \) are active and controlled, whereas \( N_p \) are permanently passive from \( t_0 \). By definition, \( N_a + N_p = N \). The operational state \( \chi_k(t) \) of each \( k^{th} \) agent includes at the top the Cartesian position of the agent's center of mass, and is governed by Eq. \[1\] For the permanently passive agents, \( u(t) = 0 \), \( \forall t \geq t_0 \). Whereas, for active agents, control goes
to zero after a contingency. In particular, define with $\chi_k(t_i)$ the operational state of $k^{th}$ agent on the uncontrolled trajectory generated at instant $t_i$. If $k^{th}$ agent is active, $t_i$ is the contingency instant; if it is permanently passive $t_i = t_0$ by definition. Furthermore, define as $S_{kjl}(t) = ||S(\chi_k(t_i) - \chi_j(t_i))||_2$ the separation reached at time $t$ by $k^{th}$ and $j^{th}$ agents on the uncontrolled trajectories generated at instants $t_i$ and $t_l$ respectively. The matrix $S \in \mathbb{R}^{3x3}$ selects the components of the Cartesian position within the state, and $||.||_2$ is the $L_2$-norm. Reasoning deterministically, PS can be formulated by using Eq. 5 with the $M$ function defined as

$$M(\chi_k(t), \chi_j(t)) = \epsilon - S_{kjl}(t) = \epsilon - ||S(\chi_k(t_i) - \chi_j(t_i))||_2$$

(17)

$\forall t_i \in [t_0, t_f], \forall t_l \in [t_0, t_f], \forall t \in [\max\{t_i, t_l\}, t_f + T], k = 1, \ldots, N - 1, j = k + 1, \ldots, N$. Where, for example, if $j^{th}$ agent is permanently passive then $t_l = t_0$. Using the $M$ formulation defined in Eq. 17 the computational requirements in terms of number of constraints to be enforced are reported in Table 1 on the top, and formal proof is developed in Appendix VIII.A. The interval $[t_0, t_f]$ is discretized in $n$ samples, whereas the interval $[t_f, t_f + T]$ in $m$ samples. A distinction between "continuous" and "impulsive" control types is made. This distinction is motivated by the fact that failure to apply control can occur just when control application is actually planned. In particular, in the continuous type, control is assumed to be planned and possibly lost at each of the $n$ sample instants in $[t_0, t_f]$. In the impulsive type, control/impulse maneuvers are assumed to be planned and possibly failed just in a subset $n_M \leq n$ of the $n$ instants, where for simplicity the ratio $n/n_M \in \mathbb{N}_{>0}$. In both types, safe separation must be enforced at each of the $n$ and $m$ instants. This guarantees that even the uncontrolled arcs in between impulse maneuvers are collision free. The number of constraints required to enforce PS between active agents are reported in Table 1 on the left, whereas between active and permanently passive agents on the right. To summarize, using the $M$ formulation, for both control types, the number of constraints to be enforced is cubic in the number of discrete time samples for PS between active agents, whereas it is quadratic in the number of discrete time samples for PS between active and permanently passive agents. For $n_M \rightarrow n$, the number of constraints required in impulsive control tends to the ones required in continuous control, without affecting the polynomial degree behavior. The number of constraints scales quadratically with the number of agents.

#### Table 1 Passive safety constraint requirements, numerical examples are shown in Figure 2

| Formulation | Control type | Constraints Between Active Agents | Constraints Between Active and Permanently Passive Agents |
|-------------|--------------|----------------------------------|--------------------------------------------------------|
| M           | Continuous   | $\frac{n(n+1)}{2} + (n+1)nM$   | $n(N N_p)$                                             |
|             | Impulsive    | $\frac{n^2}{2} + \frac{n^2}{2} - \frac{N N_p}{2}$ | $n(M n M_p)$                                          |
| $M^*$       | Continuous   | $n^2 - \frac{n^2}{2} + \frac{n^2}{2} - \frac{N N_p}{2}$ | $nN_p N_p$                                            |
|             | Impulsive    | $n^2 - \frac{n^2}{2} + \frac{n^2}{2} - \frac{N N_p}{2}$ | $nM n M_p$                                            |

By introducing the assumption in Eq. 17, this high number of constraints is reduced. In particular, the procedure presented in Section III is applied and, under the assumption of integrability of the dynamics after a contingency, the $M^*$ function is defined as

$$M^*(c_k(t_i), c_j(t_i)) = \epsilon - S'_{kjl}(c_k(t_i), c_j(t_i)) = \epsilon - ||S(\Psi(t'_{kjl})(c_k(t_i), c_j(t_i)), c_k(t_i)) - \Psi(t'_{kjl})(c_k(t_i), c_j(t_i), c_k(t_i))||_2$$

(18)

$\forall t_i \in [t_0, t_f], \forall t_l \in [t_0, t_f], k = 1, \ldots, N - 1, j = k + 1, \ldots, N$. In Eq. 18 $S'_{kjl}$ is the minimum separation (in terms of $L_2$-norm distance) reached by agents $k^{th}$ and $j^{th}$ on the uncontrolled trajectories generated at instants $t_i$ and $t_l$ respectively. Using Eq. 8 the instant $t'_{kjl}$ of minimum separation on the uncontrolled trajectories is computed as

$$t'_{kjl}(c_k(t_i), c_j(t_l)) = \arg \min_{t \in [\max\{t_i, t_l\}, t_f + T]} S_{kjl}(t) = \arg \min_{t \in [\max\{t_i, t_l\}, t_f + T]} ||S(\Psi(t, c_k(t_i)) - \Psi(t, c_j(t_i)))||_2$$

(19)

As derived in Eq. 8 and 9 $t'_{kjl}$ is a function of the integration constants (IC) states at contingency instants. This makes $S_{kjl}$ a function of these IC states, and independent of the time along the uncontrolled trajectories. This implies that safe separation all-along the uncontrolled trajectories is guaranteed by enforcing as many constraints as the number of combinations of contingency instants, i.e. once per each $kjl$-combination. As presented in Table 1 on the bottom and proved in Appendix VIII.A this reduces the number of constraints required to enforce PS by one polynomial degree of the number of discrete time samples. Specifically, from cubic to quadratic for each pair of active agents, and from quadratic to linear for each pair of active-permanently passive agents. Moreover, using the $M^*$ formulation, the number of constraints required to enforce PS becomes independent of the length of the PS horizon $T$ itself. Figure 2 presents
Fig. 2 Number of constraints as a function of the number of discrete time samples

numerical examples of the trends in Table I. To isolate the benefit of the $M^*$ formulation, the number of agents is set to $N_a = 2$ in Figures 2a-2b, and to $N_a = N_p = 1$ in Figures 2c-2d, moreover for simplicity, $m$ is set equal to $n$.

The $M^*$ formulation assumes integrability of the dynamics after a contingency, therefore, as presented in Section II, it requires the compensation of the effects of non-integrable dynamics and uncertainties within a bound defined as in Eq. 14. For multi-agent PS, assume these effects can be superimposed as

$$
\frac{\partial M'(e_k, e_j)}{\partial \tilde{e}_k} \int_{t_i}^{t_i+T} \tilde{d}_c(\tau, \tilde{e}_k(\tau)) d\tau + \frac{\partial M'(e_k, e_j)}{\partial \tilde{e}_j} \int_{t_i}^{t_i+T} \tilde{d}_c(\tau, \tilde{e}_j(\tau)) d\tau
$$

$$
+ \frac{\partial M'(e_k, e_j)}{\partial e_k} \int_{t_i}^{t_i+T} \text{diag} \left( \int_{t_i}^{t_i+T} \left( \tilde{D}_c(\tau, \tilde{e}_k(\tau)) \tilde{C}_k(\tau) + C_k(\tau) \tilde{D}_c^T(\tau, \tilde{e}_k(\tau)) + \tilde{Q}_k(\tau) \right) d\tau + C_k(t_i) \right) d\tau
$$

$$
+ \frac{\partial M'(e_k, e_j)}{\partial e_j} \int_{t_i}^{t_i+T} \text{diag} \left( \int_{t_i}^{t_i+T} \left( \tilde{D}_c(\tau, \tilde{e}_j(\tau)) \tilde{C}_j(\tau) + C_j(\tau) \tilde{D}_c^T(\tau, \tilde{e}_j(\tau)) + \tilde{Q}_j(\tau) \right) d\tau + C_j(t_i) \right) d\tau
$$

where $\beta_{k,j,l,t} \in \mathbb{R}_{>0}$. Note that Eq. 20 assumes the capability of retrieving an estimate of both active and permanently passive agents’ states through navigation and prediction. Using Eq. 20 and Eq. 18 evaluated at the mean IC states, Eq. 15 is specialized for multi-agent PS as

$$
S^*_k, j(t_i, \tilde{e}_k(t_i)) \geq \epsilon + \beta_{k,j,l,t} \geq (t_i, t_l, T, q)
$$

A graphical representation is provided in Figure 3. Assuming integrability of the dynamics after a contingency, $S^*_k,j,l$ is the minimum separation between agents $k^{th}$ and $j^{th}$ who experienced control losses at instants $t_i$ and $t_l$ respectively. The bound $\beta_{k,j,l,t}$ accounts for non-integrable dynamics and uncertainties effects on this minimum separation.
The OCP in Eq. 22 enforces PS, as in Definition 1, over an optimization problem has to be time discretized. To not replicate analyses already available in literature [24–26], the guarantees can be achieved. To apply SCP, non-convex constraints and cost function have to be linearized with respect to reformulation can be used within a sequential optimization routine, whereby the desired fault-tolerant motion safety envisioned reformulation of the PS constraint, the minimum separation number of constraints presented in Table 1 on the bottom. From a practical implementation perspective, to achieve the subject to $\mathbf{u}_k(t) = \mathbf{d}_e(t, \hat{\mathbf{c}}(t)) + \mathbf{B}_c(t, \hat{\mathbf{c}}(t))\hat{\mathbf{u}}_k(t)$ $\forall t \in [t_e, t_f], \; k = 1, \ldots N_a$ $\hat{\mathbf{c}}_k(t) = \hat{\mathbf{d}}_e(t, \hat{\mathbf{c}}_k(t))$ $\forall t \in [t_e, t_f], \; k = 1, \ldots N_p$ $\mathbf{C}_k(t) = \mathbf{D}_e(t, \hat{\mathbf{c}}_k(t))\mathbf{C}_k(t) + \mathbf{C}_k(t)\mathbf{D}_e^T(t, \hat{\mathbf{c}}_k(t), \hat{\mathbf{u}}_k(t)) + \mathbf{B}_c(t, \hat{\mathbf{c}}_k(t))\mathbf{U}_k(t)\mathbf{B}_c^T(t, \hat{\mathbf{c}}_k(t)) + \mathbf{Q}_k(t) + \mathbf{E}_k(t)$ $\forall t \in [t_e, t_f], \; k = 1, \ldots N_p$ $S_{ijk}^k(\hat{\mathbf{c}}_k(t_i), \hat{\mathbf{c}}_j(t_i)) \geq \varepsilon + \beta_{ijk}(t_i, t_i, T, q)$ $\forall t \in [t_i, t_f], \; k = 1, \ldots N_a$ $\hat{\mathbf{u}}_k(t) \in \mathcal{U}_k$ $\hat{\mathbf{c}}_k(t_f) \in \mathcal{C}_{i_f}$ $N(\hat{\mathbf{c}}_k(t_e), \mathbf{C}_k(t_e)) \equiv k^{th}$ agent’s state estimate

The OCP in Eq. 22 enforces PS, as in Definition 1, over $[t_e, t_f], \; \forall t_e \in [t_0, t_f]$, at the cost of checking the reduced number of constraints presented in Table 1 on the bottom. From a practical implementation perspective, to achieve the envisioned reformulation of the PS constraint, the minimum separation $S_{ijk}^k$ has to be computed as a function of the IC states at contingency instants. This requires the preliminary solution of the optimization problem in Eq. 19. In the following section, an algorithmic solution framework is presented. Finally, note that, if PS is enforced explicitly within the optimization problem, the trend of the number of required constraints is directly connected to the similar trend of the computational effort required to solve the optimization problem itself. Therefore the proposed efficient formulation is expected to provide benefits in terms of solution run-time. This is verified and analyzed in Section VI applied to the solution of the passively-safe reconfiguration of VISORS [37].

IV. Algorithmic Solution Framework

A. Solution of the Centralized Problem

Eq. 22 is a non-convex centralized optimization problem. To solve it in a centralized fashion, various optimization algorithms can be used. Here a direct approach based on sequential convex programming (SCP) is followed [24–26]. Specifically, this Section does not aim to derive a novel SCP algorithm, but to show how the proposed constraint reformulation can be used within a sequential optimization routine, whereby the desired fault-tolerant motion safety guarantees can be achieved. To apply SCP, non-convex constraints and cost function have to be linearized with respect to the previous solution in the series, defined as "SCP reference", and noted with $(\cdot)$: $\{\hat{\mathbf{c}}_k(t), \hat{\mathbf{u}}_k(t)\}$. Moreover, the optimization problem has to be time discretized. To not replicate analyses already available in literature [24–26], the
focus here is on the newly formulated passive safety (PS) constraint. The turning point from an algorithmic perspective lies in the availability or not of an analytical expression of $S_{k,jil}^*$ as a function of the integration constants (IC). This is presented in Algorithm 1 and analyzed in the following.

#### Algorithm 1: Enforces passive safety on $[t_0, t_f]$ for at least $T$, of at least $\epsilon$, at $\sigma$-confidence

**Input:** Initial state estimate: $\mathcal{N}(\tilde{e}_k(t_0), C_k(t_0))$, $k = 1, \ldots N$. Initial guess on: $\{\tilde{e}_k(t_i), \tilde{u}_k(t_i)\}, t_i \in [t_0, t_f]$.

**Output:** $\tilde{u}_k(t_i), t_i \in [t_0, t_f], k = 1, \ldots N_a$.

**Data:** $t_0, t_f, T, \epsilon, q, C_{k,f}, MAX\_ITER$, SCP parameters.

**begin**

for $t_e = t_0$ to $t_f$ do

Get $\mathcal{N}(\tilde{e}_k(t_e), C_k(t_e))$, $k = 1, \ldots N$.

$\{\tilde{e}_k(t_i), \tilde{u}_k(t_i)\} \leftarrow \{\tilde{e}_k(t_i), \tilde{u}_k(t_i)\}, t_i \in [t_e, t_f], k = 1, \ldots N_a$

if Closed-form expression of $S_{k,jil}^*$ is available then

Standard SCP, noting that $\tilde{\beta}_{k,jil}$ in Eq. 23 can be precomputed at the reference.

else

Precompute $\{\tilde{f}_{k,jil}^*, \tilde{\beta}_{k,jil}\}$ at the reference, $\forall k,j,l$.

$PS \leftarrow 0$

while SCP not converged or $PS = 0$ do

$PS \leftarrow 0$, $ITER \leftarrow 0$

while $PS = 0$ and $ITER < MAX\_ITER$ do

Solve Eq. 23, enforcing Eq. 24.

Get $\{\tilde{e}_k(t_i), \tilde{u}_k(t_i)\}, t_i \in [t_0, t_f], k = 1, \ldots N_a$.

Recompute $\{\tilde{f}_{k,jil}^*, \tilde{\beta}_{k,jil}\}$ at the new solution, $\forall k,j,l$.

if $S_{k,jil}(\tilde{f}_{k,jil}^*, \tilde{e}_k(t_i), \tilde{e}_j(t_i)) \geq \epsilon + \tilde{\beta}_{k,jil}, \forall k,j,l$ then

$PS \leftarrow 1$

$\{\tilde{f}_{k,jil}^*, \tilde{\beta}_{k,jil}\} \leftarrow \{\tilde{f}_{k,jil}^*, \tilde{\beta}_{k,jil}\}, \forall k,j,l$.

$ITER \leftarrow ITER + 1$

if $PS = 0$ then

Select among the $MAX\_ITER$ solutions computed the one that got closer to satisfy

$S_{k,jil}(\tilde{f}_{k,jil}^*, \tilde{e}_k(t_i), \tilde{e}_j(t_i)) \geq \epsilon + \tilde{\beta}_{k,jil}, \forall k,j,l$.

$\{\tilde{e}_k(t_i), \tilde{u}_k(t_i)\} \leftarrow \{\tilde{e}_k(t_i), \tilde{u}_k(t_i)\}, t_i \in [t_e, t_f], k = 1, \ldots N_a$.

Recompute $\{\tilde{f}_{k,jil}^*, \tilde{\beta}_{k,jil}\}$ at the new reference, $\forall k,j,l$.

end

Get $\tilde{u}_k(t_e), k = 1, \ldots N_a$.

**return** $\tilde{u}_k(t_i), t_i \in [t_0, t_f], k = 1, \ldots N_a$.

If an analytical expression of $S_{k,jil}^*$ is available and is, in general, a non-convex function of the IC state, it can be sequentially linearized around the SCP reference as

$$
\frac{\partial S_{k,jil}^*(\tilde{e}_k, \tilde{\varepsilon})}{\partial \tilde{e}_k} \epsilon_k(t_i, \tilde{\varepsilon}_j(t_i)) + \frac{\partial S_{k,jil}^*(\tilde{e}_k, \tilde{\varepsilon})}{\partial \varepsilon_j} \varepsilon_j(t_i) + S_{k,jil}^*(\tilde{e}_k(t_i), \tilde{\varepsilon}_j(t_i)) \geq \epsilon + \tilde{\beta}_{k,jil} \quad (23)
$$

where $\tilde{\beta}_{k,jil}$ (defined in Eq. 20) is, by choice, directly evaluated at the reference. Since, from a PS enforcement perspective, the IC state covariances are used just to quantify this bound, this permits: 1) pre-propagating them around the reference mean IC states and control inputs, 2) quantifying $\tilde{\beta}_{k,jil}$ evaluating the terms on the left-hand side of Eq. 20 at the reference, 3) enforcing Eq. 23 within the optimization problem, without having to enforce explicitly the covariance dynamics constraint. This is an approximate but simpler solution than linearizing the terms on the left-hand side of Eq. 20. The accuracy of this approximation can be regulated through additional trust region constraints [25, 26]. Note that, by definition, the integrals in Eq. 20 cannot be solvable explicitly and in closed-form. Therefore, they are pre-computed at the SCP reference using, for example, numerical integration, or even semi-analytical techniques if models of the non-integrable portion of the dynamics are available, as for the distributed space system application [19, 23]. The resulting bound $\beta$ is fed as a parameter in the optimization problem solved at each SCP iteration. Moreover,
note that the IC states of permanently passive agents at \( t_0 \) can be treated as input parameters in the optimization, and not as optimization variables. This simplifies the left-hand side of Eq. \( 23 \) for active-permanently passive agents pairs. In particular, the linearization with respect to the permanently passive agents’ IC states is not performed.

If an analytical expression for \( S_{kjil} \) is not available, due to the fact that Eq. \( 19 \) is too complicated to be solved in closed-form, then an algorithmic development is needed. This is presented in Algorithm \( 1 \) (lines 9-24) and in the following. In particular, at each iteration of the SCP loop, \( \hat{t}_{k,jil} \) is precomputed at the reference through a simple search algorithm on the discretized interval \( \{ t_0, t_f \} \). Using Eq. \( 19 \), this search algorithm selects the discrete instant/index that minimizes \( S_{kjil}(t_i, \hat{e}_k(t_i), \hat{e}_j(t_i)) \), available as a closed-form expression of the reference mean IC states under assumption of integrability of the dynamics after a contingency. Using the precomputed \( \hat{t}_{k,jil} \), the PS constraint is enforced as

\[
\frac{\partial S_{kjil}(\hat{t}_{k,jil}, \hat{e}_k(t_i), \hat{e}_j(t_i))}{\partial \hat{e}_k(t_i)} (\hat{e}_k(t_i) - \hat{e}_k(t_0)) + \frac{\partial S_{kjil}(\hat{t}_{k,jil}, \hat{e}_k(t_i), \hat{e}_j(t_i))}{\partial \hat{e}_j(t_i)} (\hat{e}_j(t_i) - \hat{e}_j(t_0)) \geq \epsilon + \beta_{k,jil} \tag{24}
\]

With arguments similar to the one used for the standard collision avoidance in \( 30 \), it is possible to prove that Eq. \( 24 \) represents a sufficient condition for the satisfaction of \( S_{kjil}(\hat{t}_{k,jil}, \hat{e}_k(t_i), \hat{e}_j(t_i)) \geq \epsilon + \beta_{k,jil} \). The formal proof is presented by the authors in \( 34,36 \). Nevertheless, Eq. \( 24 \) does not constitutes a sufficient condition of having the actual minimum separation at \( \hat{t}_{k,jil} \) being greater than \( \epsilon + \beta_{k,jil} \). This is represented graphically in Figure 4a. At the SCP reference, minimum separation on the uncontrolled trajectories is reached at instant \( \hat{t}_{k,jil} \). At the current solution, at instant \( t_{k,jil}' \) the separation reached at instant \( \hat{t}_{k,jil} \) by the current solution may be larger than the minimum one since \( t_{k,jil}' \neq \hat{t}_{k,jil} \). Therefore, \( \hat{t}_{k,jil} \) has to be sequentially updated within the SCP pipeline to guarantee that, at convergence,

\[ t_{k,jil} \]

\[ \text{Fig. 4} \quad \text{Enforcement of passive safety (PS) within the sequential convex programming (SCP) pipeline} \]

the actual worst case minimum separation is constrained. In particular, 1) the optimization problem is solved by enforcing Eq. \( 24 \) which uses \( \hat{t}_{k,jil} \) and \( \beta_{k,jil} \) evaluated at the reference, then 2) \( t_{k,jil}' \) and \( \beta_{k,jil} \) are computed at the new solution, then 3) the actual minimum separation \( S_{kjil}(t_{k,jil}', \hat{e}_k(t_i), \hat{e}_j(t_i)) \) is checked a-posteriori to be \( \geq \epsilon + \beta_{k,jil} \). If this condition is true, the algorithm proceeds to the next SCP-loop iteration. If it is not true, the optimization problem is solved again keeping the same linearization reference but using the newly computed \( t_{k,jil}' \) and \( \beta_{k,jil} \). This inner-loop breaks if PS is satisfied a-posteriori, or if a maximum number of iterations \( (\text{MAX}_\text{ITER} \in \mathbb{N}_{>0}) \) is reached. If the maximum number of iterations is reached, the solution in the inner-loop closest to satisfying PS is taken as reference and the algorithm proceeds to the subsequent SCP iteration. Introducing a maximum number of iterations in the inner-loop prevents from getting stuck into intermediate solutions, in which the PS constraint does not get satisfied in few iterations. Figure 4b presents a sample trend of PS enforcement moving through the inner-loops of Algorithm \( 1 \) (lines 12-18, light-blue line, with \( \text{MAX}_\text{ITER} = 3 \), and \( \epsilon = 5 \mathrm{m} \)) and the SCP outer-loop (blue line). The algorithm is considered converged to a feasible solution, only if the final solution out of the SCP loop is passively-safe. Otherwise, the problem is deemed unfeasible.

The feasibility of finding a passively-safe solution trajectory depends on various factors, including: the boundary conditions, the level of uncertainty present in the system (which influences the size of the PS bounds), and optimization parameters such as the initial warm-starting trajectory, and the defined time discretization. Certain optimization
parameters can be properly selected and tuned to increase the chances of finding a feasible solution. Other factors such as the boundary conditions or the level of uncertainty are defined given the system and control scenario. Therefore, given a certain system, control scenario and optimization set-up, the possibility of not finding a feasible passively-safe solution is present. In risk sensitive applications, if a feasible solution is repeatedly not found, it is necessary to have a logic that falls back on an alternative action. For distributed space systems, this alternative action can be, for example, an escape maneuver performed by those agents that have not completely lost control.

Both if $S_{kji}^*$ is available in closed-form and not, Eq. [23] and [24] enforce the reduced number of constraints presented in Table 1 on the bottom. This possibly cuts the computational expense, and increases the chances of converging in useful time to a feasible solution using off-the-shelf solvers [39, 40]. As final mention, as in any SCP algorithm, to implement Algorithm 1 an initial "warm-start" guess on the profile: $\{\tilde{c}_k(t_i), \tilde{u}_k(t_i)\}$, $\forall k$, is required as input. This initial guess does not have to be feasible, i.e., it does not have to satisfy the PS constraint. An option is to initialize the algorithm with a profile that simply solves for the two-point-boundary-value-problem from initial to final conditions, or, better, that satisfies the standard collision avoidance constraint but is not passively-safe. Obviously, the computational effort is reduced if a sub-optimal passively-safe solution is already available, as analyzed in Section VI.

B. Decentralization

In a multi-agent setting, using Algorithm 1 at each $t_e \in [t_0, t_f]$ requires: 1) the communication of the state estimates of each agent to the one assigned with the role of computing the centralized plan, and 2) the communication back to each agent of the newly computed plan. From an operational stand-point, it may be beneficial to do the computation of the centralized plan at the initial instant $t_0$ and then at subsequent instants $t_e$ only when necessary. This subdents the need of complementing the centralized problem with an additional decentralized optimization problem, defined conventionally as the “tracking problem”. This enables each agent to independently compute a “tracking control plan” that tracks a previously computed centralized plan, any time an updated state estimate is available. The tracking problem has to compensate for the presence of initial tracking errors, as well as for the acting dynamics, while reinforcing guarantees of constraint satisfaction. In [36], the authors present a possible formulation of the tracking problem, as well as the logic regulating the interplay between centralized and decentralized problems solution. Here, for conciseness, the key aspects are summarized. Introduce the superscript $(.)^G$ to indicate variables associated to the centralized "guidance" and the superscript $(.)^T$ to indicate variables associated to the decentralized "tracking". Algorithm 1 is used at $t_0$ to produce the centralized plan $\{\tilde{u}_k^G(t), \tilde{c}_k^G(t), C_k^G(t)\}, \forall t \in [t_0, t_f]$, for each active agent. If feasible, the centralized plan is certified as passively-safe. At each instant $t_e$, each agent acquires a state estimate $N(\tilde{e}_k(t_e), C_k(t_e))$. Given this estimate and the available centralized plan, a tracking error is possibly present and defined as $\tilde{e}_k^T(t_e) = \tilde{e}_k(t_e) - \tilde{c}_k^G(t_e)$. This tracking error propagates through the dynamics of the system. In addition, the newly estimated covariance $C_k(t_e)$ is not necessarily equal to the one predicted during the solution of the centralized problem, and used to evaluate the bounds in Eq. 23 and 24. To make the guidance plan robust to these tracking errors, the idea is to enlarge the bound used at guidance level by an additional margin. In particular, for each involved agent, the following term can be added to the left-hand side of Eq. [20] $\int (\partial M^T / \partial \tilde{e}_k) \tilde{c}_k(t_e) \cdot v_{ki}^T$. Here, $v_{ki}^T \in \mathbb{R}^s_{\geq 0}$ is a constant total worst-case tracking error which is then reinforced at tracking level on the mean tracking state as

$$\left|\hat{e}_k^T(t_i)\right| + \beta_{ki}^T \leq v_{ki}^T$$

(25)

$\forall t_i \in [t_0, t_f], for k = 1, \ldots, N_a$. In Eq. 25 $\beta_{ki}^T \in \mathbb{R}^s_{\geq 0}$ is quantified such that

$$\beta_{ki}^T \geq \left|\int_{t_i}^{t_f+t} \tilde{d} c_e(\tau, \tilde{e}_k(\tau))d\tau - \int_{t_i}^{t_f+t} \tilde{d} c_e^G(\tau)d\tau\right|$$

$$+ q\sqrt{\text{Diag} \left(\int_{t_i}^{t_f+t} \left(\tilde{D} c_e(\tau, \tilde{e}_k(\tau))C_k(\tau) + C_k(\tau)\tilde{D} c_e^G(\tau, \tilde{e}_k(\tau)) + Q_k(\tau)\right)d\tau + C_k(t_i)\right)}$$

$$- q\sqrt{\text{Diag} \left(\int_{t_i}^{t_f+t} \left(\tilde{D} c_e(\tau, \tilde{e}_k(\tau))C_k^G(\tau) + C_k^G(\tau)\tilde{D} c_e(\tau, \tilde{e}_k(\tau)) + Q_k^G(\tau)\right)d\tau + C_k^G(t_i)\right)}$$

(26)

to capture the net effect of tracking on the non-integrable and uncertainty effects quantified at guidance level. In particular, in Eq. 26 the non-superscripted state quantities refer to the ones computed at tracking level, which depend on the tracking error itself.
Finally, a high-level logic regulating the interplay between guidance and tracking is implemented. Specifically, at every instant \( t_e \), an updated state estimate is available, the system has two options: a) tracking the centralized plan or b) recomputing the centralized plan. The high-level logic has to decide autonomously on-board between a) or b). The triggers in this decision are mainly two: 1) the feasibility of the tracking, and 2) the prediction that the currently planned control solution leads to constraints satisfaction, given the latest state estimate and the on-board knowledge of the system dynamics. Therefore, at each instant \( t_e \), each agent acquires an updated state estimate and tracks. If for some agents, the tracking is not feasible the agent(s) share the information and the “planner agent” recomputes the centralized plan using the latest state estimates. If the tracking is feasible for everyone, the agent(s) share their updated state estimates and the corrected control plans to the planner agent, which performs a sanity-check. In particular, it propagates in open-loop both mean and covariance trajectories of each agent starting at the newly acquired state estimates using the corrected control plans that has been communicated. This open-loop propagation permits checking if the expected worst case minimum separations reached on the uncontrolled trajectories are actually greater than \( \epsilon \). If any of them is not greater than \( \epsilon \), PS is not guaranteed by the current control plan given the newly acquired state estimate. Therefore, the centralized plan must be recomputed. Otherwise, if all of them are greater than \( \epsilon \), the agents receive the go ahead to actuate the corrected control plan. This sanity-check performed by the planner agent introduces delay and computational overhead in the control pipeline, requiring the communication back and forth of data between agents as well as the open-loop propagation needed for checking the minimum separations. Depending on the risk sensitivity of the control scenario it is possible to skip this sanity-check or perform it just at key instants of the multi-agent reconfiguration (e.g., after the actuation of a largely uncertain maneuver). If the sanity-check is not performed, the only decision trigger on recomputing the centralized plan or not remains the possible unfeasibility of the tracking.

V. Application to Distributed Space Systems

The orbital motion of a \( N \)-spacecraft distributed space system can be described either by \( N \) orbits in inertial space around the main attractor or by one reference orbit, either virtual or occupied by a chief spacecraft, and by \( N \) (or \( N - 1 \)) relative trajectories with respect to this reference. These two ways of modeling the spacecraft motion influence the structure of the governing ODE, and therefore give rise to two different ways of applying variation of parameters and the methodology proposed in Sections [11][14]. As described in the introduction, one approach uses the orbital elements (OE) as integration constants (IC) state, while the other approach uses the relative orbital elements (ROE). From an operational stand-point, for spacecraft in close-proximity, modeling directly the relative motion is advantageous both from a dynamics modeling standpoint [19][22] and navigation accuracy standpoint [41]. In particular, spacecraft absolute motion navigation solutions are usually at least one order of magnitude less accurate than the relative motion ones. Navigation accuracy plays a fundamental role in the developed framework, since it determines the level of uncertainty present, and in face of which passive safety (PS) guarantees have to be provided (see Eq. [20]). This is analyzed in the test cases in Section [71].

Following the relative motion approach, the reference orbit is uniquely defined by a set of OE: \( \textbf{e} \in \mathbb{R}^6 \). Use the quasi-nonsingular OE definition [14]: \( \textbf{e} = \{a, \nu, \alpha, e_x, e_y, i, \Omega \} \), with \( a \) the semi-major axis, \( \nu \) the true anomaly, \( \alpha \) the argument of latitude, \( i \) the inclination, \( \Omega \) the right ascension of the ascending node. The \( N \) (or \( N - 1 \)) relative trajectories are defined by the relative Cartesian state expressed in the radial/tangential/normal (RTN) [14] frame centered on the reference, as \( \chi_k = \{\delta r_k, \delta \dot{r}_k\} \in \mathbb{R}^6 \), for \( k = 1, \ldots, N \), where \( \delta r_k \in \mathbb{R}^3 \) and \( \delta \dot{r}_k \in \mathbb{R}^3 \) are correspondingly the relative position and velocity of the \( k \)th spacecraft. The dynamics of \( \chi_k \) is governed by a nonlinear ODE [42]

\[
\dot{\chi}_k(t) = z(\chi_k(t)) + p(\chi_k(t)) + Bu_k(t)
\]

(27)

where, \( z \in \mathbb{R}^6 \) models the relative effect of two-body spherical gravity as well as Euler, centrifugal, and Coriolis accelerations stemming from the non-inertial (and potentially perturbed) nature of the rotating RTN frame. \( p \in \mathbb{R}^6 \) models the differential effect of orbital perturbations, and \( u_k \in \mathbb{R}^3 \) is the \( k \)th spacecraft control acceleration expressed in RTN frame, with \( B = [0_{3x3}; 1_{3x3}] \in \mathbb{R}^{6x3} \) the control input matrix. Although the relative dynamics in Eq. [27] depends on the reference OE, this dependence is left implicit in the notation for compactness. In literature, analytical solutions for the homogeneous nonlinear part of Eq. [27] are not available. Nevertheless, linearizing the differential spherical gravity terms in \( z \) for small spacecraft separations, Eq. [27] is rewritten exactly as

\[
\dot{\chi}_k(t) = A(t)\chi_k(t) + d(\chi_k(t)) + Bu_k(t)
\]

(28)

where \( A \in \mathbb{R}^{6x6} \) is the linear(ized) integrable part, and \( d \in \mathbb{R}^6 \) is the nonlinear non-integrable part, which contains both
\[ \frac{1}{\rho} - \frac{3}{2} \tilde{\rho} \sin(\nu) \omega t \quad 0 \quad -\cos(\lambda_{\nu}) \quad -\sin(\lambda_{\nu}) \quad 0 \quad 0 \\
-\frac{\rho}{\eta} \sin(\omega t) \quad 0 \quad 0 \quad 0 \quad 0 \quad \frac{1}{\eta} \sin(\lambda_{\nu}) \quad -\frac{1}{\eta} \cos(\lambda_{\nu}) \\
-\frac{1}{\eta} \sin(\nu) \omega t \quad 0 \quad \sin(\lambda_{\nu}) \nu \quad -\cos(\lambda_{\nu}) \nu \quad 0 \quad 0 \\
-\frac{1}{\rho} \nu t - \frac{1}{\beta} \mu \nu t \quad -\frac{1}{\rho} \nu \quad \rho \nu \sin(\lambda_{\nu}) \nu \quad \rho \nu \cos(\lambda_{\nu}) \nu \quad 0 \quad 0 \\
0 \quad 0 \quad 0 \quad 0 \quad 0 \quad \frac{\rho}{\eta} \cos(\lambda_{\nu}) \nu \quad \frac{\rho}{\eta} \sin(\lambda_{\nu}) \nu \\
\]
eccentric orbits, instead, at each $v(t) = \bar{v}$, the real relative motion trajectory is tangent to the ellipses in Fig. 5 at the point $\lambda = \omega + \bar{v}$, with $\bar{\rho} = 1 + e \cos(\bar{v})$. This implies that, a lowerbound of the real separation between spacecraft on the uncontrolled trajectories can be computed in either RN or RT planes by evaluating the minimum separations of the two ellipses from the origin. As presented in the following, these minimum separations assume extreme values at the perigee and apogee of the eccentric orbit. Therefore, a lowerbound over the full orbit is obtained evaluating at the contingency instants. Taking the partials with respect to $\psi_{k,j,l}$, Eq. 21 to achieve PS. Since $c_{1,k} \sim 0$, the uncontrolled relative motion is bounded and periodic, therefore PS is achieved infinitely ($T=\infty$). Note that, if assuming $c_{1,k} \sim 0$ is too limiting for a specific application, it can be avoided by following the numerical path to $t'_{k,j,l}$ computation presented in Algorithm 1 (lines 9-24).

1. Passive Safety in RN-plane

The minimum separation of the RN ellipse from the origin is computed as the minimum eigenvalue $S_{k,j,l}^{RN} (\bar{\rho})$ of a singular value decomposition (Fig. 5, right) as

$$S_{k,j,l}^{RN} (\bar{\rho}) = \sqrt{\frac{\left| c_{34,k,j,l} \cdot \frac{\epsilon_{56,k,j,l}}{\bar{\rho}} \right|}{\left( c_{34,k,j,l} \right)^2 + \left( \frac{\epsilon_{56,k,j,l}}{\bar{\rho}} \right)^2 + c_{34,k,j,l} + \frac{\epsilon_{56,k,j,l}}{\bar{\rho}} \cdot \left| c_{34,k,j,l} - \frac{\epsilon_{56,k,j,l}}{\bar{\rho}} \right|}}^{1/2}$$ (31)

Consistently with Eq. 9, $S_{k,j,l}^{RN} (\bar{\rho})$ is, for a fixed value of $\bar{\rho}$, independent of time and a function of the IC states evaluated at the contingency instants. Taking the partials with respect to $\bar{v}$, its smallest and largest values are achieved correspondingly at the perigee and apogee of the eccentric orbit. Therefore, a lowerbound over the full orbit is obtained substituting $\bar{\rho}$ with $1 + e$. Using Eq. 31 the sufficient condition of PS in Eq. 21 can be rewritten as

$$\left| \dot{c}_{34,k,j,l} \cdot \dot{c}_{56,k,j,l} \right| = \left| \cos(\theta_{34,k,j,l} - \theta_{56,k,j,l}) \right| \geq \cos(\xi_{k,j,l}(\bar{\rho}))$$ (32)

with

$$\cos(\xi_{k,j,l}(\bar{\rho})) = \frac{\epsilon_{k,j,l} \bar{\rho}}{a c_{34,k,j,l} c_{56,k,j,l}} \sqrt{\left( c_{34,k,j,l} \right)^2 + \left( \frac{\epsilon_{56,k,j,l}}{\bar{\rho}} \right)^2 - \frac{\left( \epsilon_{k,j,l} \right)^2}{a^2}}$$ (33)

$\forall t \in [t_0, t_f], \forall t_j \in [t_0, t_f]$, for $k = 1, \ldots N - 1$, $j = k + 1, \ldots N$. Here, $c_{ab,k,j,l} = c_{ab,k,j,l} \dot{c}_{ab,k,j,l}$. Eq. 32, 33 have an interesting geometrical interpretation in IC space, as presented in Fig. 6a. In particular, to maintain PS in RN-plane the phase angle between the vectors $c_{34,k,j,l}$ and $c_{56,k,j,l}$, i.e. $\theta_{34,k,j,l} - \theta_{56,k,j,l}$, must be either contained in the interval $[-\xi_{k,j,l}(\bar{\rho}), \xi_{k,j,l}(\bar{\rho})]$ (parallel configuration) or in the interval $[\pi - \xi_{k,j,l}(\bar{\rho}), \pi + \xi_{k,j,l}(\bar{\rho})]$ (anti-parallel configuration), where the most and least tight conditions are achieved correspondingly at the perigee and apogee of the eccentric orbit. Eq. 32, 33 represent the $c_{34}/c_{56}$-separation concept, which in near-circular orbits, using ROE, becomes the relative eccentricity/inclination-separation concept [7], used for the relative orbit design in the GRACE, TANDEMEM-X, PRISMA and BIROS missions flown in low-Earth orbit [11].
2. Passive Safety in RT-plane

The minimum separation of the RT ellipse from the origin is computed by solving analytically Eq. 19 formulated as a function of \( \lambda \), and by using Eq. 30 and fixing \( \phi \). The resulting minimum separation is

\[
S_{k,j}^{RT}(\bar{\rho}) = \begin{cases} \frac{1}{N} c_{2,k,j} \bar{\rho} & \text{not-encircling} \\
\sqrt{a^2 c_{4,k,j}^2 - \frac{1}{\bar{\rho}} a^2 c_{2,k,j}^2} & \text{encircling} \end{cases}
\]

where "not-encircling" refers to Fig. 6b, top left, whereas, "encircling" refers to Fig. 6b, center and bottom left. Consistently with Eq. 29, \( S_{k,j}^{RT}(\bar{\rho}) \) is, for a fixed value of \( \bar{\rho} \), independent of time and a function of the IC states evaluated at the contingency instants. Taking the partials with respect to \( \bar{\rho} \), its smallest and largest values are achieved correspondingly at the perigee and apogee of the eccentric orbit for the not-encircling case, and at the apogee and perigee for the encircling case. Therefore, a lowerbound over the full orbit can be obtained by substituting \( \bar{\rho} \) with \( 1 + e \) for the not-encircling case, whereas with \( 1 - e \) for the encircling case. Using Eq. 34 the sufficient condition of PS in Eq. 21 can be rewritten as

\[
|c_{2,k,j}| \begin{cases} \geq (1 + \bar{\rho}) c_{34,k,j} + \frac{e_{j,k}}{a} & \text{not-encircling} \\
\sqrt{(1 + 2\bar{\rho}) c_{34,k,j}^2 - \left(\frac{e_{j,k}}{a}\right)^2} & \text{encircling} \end{cases}
\]

\[
\forall t_i \in [t_0, t_f], \forall t_j \in [t_0, t_f], \text{ for } k = 1, \ldots, N - 1, j = k + 1, \ldots, N. \text{ In IC space, Eq. 35 is represented as in Fig. 6b, right, where the most tight and least tight conditions are achieved correspondingly at the perigee and apogee for the not-encircling case, and at the apogee and perigee for the encircling case. In near-circular orbits, using ROE, Eq. 35 reduces to the solutions in [12].

Fig. 6 Graphical representation of the sufficient conditions of passive safety (PS) in eccentric orbits

B. Eccentric Swarms Designs

The newly derived analytical PS conditions in RN and RT planes permit the generalization of the spacecraft swarm designs developed in [12] from near-circular orbits to eccentric orbits. The "high-density" swarm design is defined such that all spacecraft are distributed at equal distance in the \( c_{34} \)-plane and RT PS is achieved by enforcing sufficient separation in along-track direction through Eq. 35. To do so, design the swarm such that: \( c_{1,k} = 0 \), and \( c_{34,k,j} = ||c_{34,k} - c_{34,j}|| \geq c_{34,min} \geq \epsilon / a \). Moreover, all spacecraft are distributed at equal distance (\( c_{34,min} \)) in the \( c_{34} \)-plane as

\[
c_{34,k} = \begin{cases} c_{34,min} \left( W_k \cos(\theta_{34}^h) + X_k \cos(\theta_{34}^h + \frac{\pi}{2}) \right) & \text{if } c_{34,k,j} \leq c_{34,min} \\
& \text{otherwise} \end{cases}
\]
where, the angular phase $\theta^{thd}_k$ is selected to arrange the spacecraft in the plane, and $W_k$ and $X_k$ are not-simultaneously zero integers, that cannot simultaneously satisfy $W_k = W_j$ and $X_k = X_j$, for any $k \neq j$. Given this design, by noticing that $|c_{2,k,j}| \leq |c_{2,k}| + |c_{2,j}|$, RT PS is enforced for the entire swarm by using Eq. (32-33) evaluated at the apogee (the most tight condition for the encircling case) as

$$|c_{2,k}| \leq \sqrt{\frac{2-\pi e}{\pi a} \left(c_{34,\text{min}} - \frac{\pi}{a}\right)} \text{ if } \frac{c_{34,\text{min}}}{a} \leq c_{34,\text{min}} < \frac{2-\pi e}{\pi a} \quad (37)$$

The RN separated swarm is designed instead such that all the spacecraft share same phase angles of the $c_{34}/c_{56}$ vectors, while having these vector separated according to Eq. (32-33). To do so, design the swarm such that: $c_{1,k} = 0$, $c_{34,k,j} = ||c_{34,k} - c_{34,j}||_2 \geq c_{34,\text{min}} \geq e/a$, and $c_{56,k,j} = ||c_{56,k} - c_{56,j}||_2 \geq c_{56,\text{min}} \geq (e/a)(1 + e)$. Moreover, all the spacecraft must share same phase angles of the $c_{34}/c_{56}$ vectors, $\theta_3^{\text{rep}}$ and $\theta_5^{\text{rep}}$, as: $\hat{c}_{34,k} \equiv \left\{ \cos(\theta_3^{\text{rep}}), \sin(\theta_3^{\text{rep}}) \right\}$, and $\hat{c}_{56,k} \equiv \left\{ \cos(\theta_5^{\text{rep}}), \sin(\theta_5^{\text{rep}}) \right\}$. Given this design, RN PS is enforced for the entire swarm by using Eq. (32-33) evaluated at the perigee (most tight condition) as

$$|\cos(\theta_3^{\text{rep}}) - \theta_5^{\text{rep}}| \geq \frac{e(1+e)}{ac_{34,\text{min}}c_{56,\text{min}}} \sqrt{(c_{34,\text{min}})^2 + \left(c_{56,\text{min}}^2 + 1 + e \right)^2 - \frac{e^2}{a^2}} \quad (38)$$

Note that for eccentricity going to zero, these two swarms designs reduce exactly to the ones in [12], with the selected IC state reducing to the quasi-nonsingular ROE through the map in Eq. (38) in Appendix VIII.B. These two passively-safe spacecraft swarm designs are showcased in the next section, together with a reconfiguration between them enforcing PS in 3-D. This capability is not published in literature and is enabled by Algorithm 1.

VI. Numerical Results

A. The Virtual Super-resolution Optics with Reconfigurable Swarms (VISORS)

Due to launch in 2024, VISORS is a distributed telescope composed by two 6U CubeSats that will operate in a near-circular sun-synchronous low-Earth orbit to image the coronal area of the sun with unprecedented resolution [37]. One spacecraft (OSC) contains the optical payload, the other (DSC) contains the detector. During nominal operations, DSC acts as passive chief, whose center of mass sets the radial/tangential/normal (RTN) frame’s origin, whereas OSC actively controls the relative motion. The test cases considered model the challenging “transfer phase”, where a passively-safe control plan that moves the formation from the stand-by mode (initial condition) to the science mode (target condition) has to be computed. Both initial and target modes are designed in closed-form to achieve radial/normal (RN) passive safety (PS) through relative eccentricity/inclination-separation [12] (Eq. (32-33)). Nevertheless, PS has not only to be guaranteed at initial and terminal states, but at every instant of the transfer too. Therefore, Algorithm 1 has to be used to generalize over pure closed-form analysis and compute the transfer. All the parameters of the reconfiguration scenario are reported in Table 2. Two transfers are considered, 2 and 5 orbits long, in which the $L_2$-norm of the control input is minimized, and PS is guaranteed in RN plane for at least two orbits ($T$), of at least $3$ or $5m$ ($e$), at $3$-or confidence ($\sigma$). The uncertainty stems from differential GPS navigation [41], cold-gas impulsive actuation [43], and process-noise modeling the discrepancy between the full-force ground-truth [37, 41] and the on-board dynamics model [20, 22] used for optimization. In particular, looking at Table 2 on the right, the relation between maneuver magnitude and associated actuation execution uncertainty is proportional, which motivates the uncertainty model formalized in Section 11. Moreover, looking at Table 3 on the right, the process noise has to account for the unmodeled and possibly stochastic effects of differential drag present in low-Earth orbit. Finally, note that the slowly time-varying nature of the orbital elements state used in the numerical integration of the Gauss variational equations (GVE) allows for the large integration time steps considered in Table 3, while retaining high propagation accuracy [44].

I. Two-orbit Transfer

Algorithm 1 is used to solve for the two-orbit transfer. The solution of the convex two-point-boundary-value-problem (TPBVP) [40] is used as initial warm-start, MAX_JTER is set to 3, and trust-regions varying from 100m to 5m along the transfer are implemented. The underlying optimization problem is a second order cone program (SOCP) [39]. The optimizer looks for candidate impulsive maneuvers every 30deg along the orbit, so $n_M = 25$, whereas the discretization
Table 2  VISORS mission’s reconfiguration scenario

| Reference orbit | Relative orbit modes | Transfer & safety | GPS Navigation | Impulsive Actuation |
|-----------------|----------------------|-------------------|---------------|-------------------|
| Type            | Chief                | Stand-by (t<sub>0</sub>) | Science (t<sub>f</sub>) | PS type | RN |
| ω (deg)         | 98                   | ac<sub>y</sub> (m) | -100          | 10 or 34.76 | T (orbits) | 2 |
| e (-)           | 0.001                | ac<sub>x</sub> (m) | 0            | 25 or -34.64 | e (m) | 3 or 5 |
| i (deg)         | 98                   | ac<sub>y</sub> (m) | 0            | 200 or 0.296 | q (°) | 3 |
| Ω (deg)         | 252 or 330           | ac<sub>x</sub> (m) | 0            | 15 or -19.72 | J<sup>2</sup> (m/s) | Σ<sub>/</sub> | 25 or 75 |
| ω<sub>0</sub> (deg) | -90 or 1             | ac<sub>x</sub> (m) | 200          | 0 or 1.69 | n<sub>e</sub> (-) | 720 or 1800 |
| Period (h)      | 1.58                 |                   |               |                   | n (-) | 720 or 720 |
| Epoch           | 1-1-2024             |                   |               |                   |     |     |
| GPS time        | 00° 00’ 00”          |                   |               |                   |     |     |

Table 3  Spacecraft features and dynamics models

| Spacecraft features | Dynamics models |
|---------------------|----------------|
| Mass (kg) 12        | Ground-truth | On-board prediction | On-board optimization |
| Cross-section area (m<sup>2</sup>) | nom: 0.285 | Geopotential | GVE numerical integration | GVE numerical integration | Analytical |
| avg: 0.206          | type       | GGM01S (60 x 60) | GGM01S (20 x 20) | J2 secular & long-period |
| C<sub>d</sub> (-)   | 2.3        | Atmospheric drag | NRLMSISE00 | NRLMSISE00 | Secular & long-period model-free |
| C<sub>r</sub> (-)   | 1.88       | Solar radiation pressure | Cannonball model, no eclipses | None | Secular & long-period proportional to differential ballistic coefficients |
| Solar radiation     | Integrator | Third-body gravity | Analytical lunisolar ephemerides | None | Luni-solar secular & long-period |
| pressure            | Step size  |                   |                   | euler |       |
|                   | fixed: 30s |                   |                   |     |       |
|                   | fixed: 60s |                   |                   |     |       |
|                   | fixed: see Table 4 |                   |                   |     |       |
| Process-noise on IC (m/orbit) | None | None | None | 3 |

at which PS must be enforced is at 1deg, therefore n = m = 720. PS is enforced after each maneuver application to guarantee safety both on the trajectory in between maneuvers and, in case of subsequent maneuver failure, afterwards up to t<sub>f</sub> + T. Given the selected orbit discretization, the proposed efficient M<sup>+</sup> formulation permits the reduction of the number of constraints required to enforce PS from > 25 x 720 = 18000 to 25. By explicitly enforcing these constraint within the SOCP, this implies in average a computational saving of a factor > 100 (from 33s to 0.27s) on the solution of an instance of the optimization problem in Eq. 22 using a SOCP solver [40]. The computational overhead introduced by the precomputation of t<sub>k,j,l</sub>* is minimal, since it is performed jointly with the precomputations required to evaluate B<sub>k,j,l</sub> as well as other matrices needed in the SOCP, for a total execution time of 1s. Note that, if all the thousands of constraints are enforced explicitly and t<sub>k,j,l</sub>* is not computed, precomputations are still needed at each sequential convex programming (SCP) iteration, but may be lighter depending on the implementation. On the other hand, the computational gain obtained on the solution of the SOCP by enforcing just 25 constraints is much greater.

Fig. 7  Monte Carlo analysis on the computational effort of solving a second order cone program instance

‡Computation times reported are from a Windows system equipped with a 4GHz processor and 64GB RAM.
Figure 7 enlarges this analysis showing the results of a Monte Carlo on the computational effort by varying the orbit discretization between maneuvers ($n_M$) and at which safe separation must be enforced ($n = m$). Looking at Figures 7a and 7b, the expected correlation between time/orbit discretization, number of constraints, and computation time is clear. In particular, the computational effort trends using the $M$ and $M^*$ formulations follow closely the ones of the number of enforced constraints presented in Figure 2. Looking at Figure 7c, the resulting computational saving factor depends on the time/orbit discretization, and it is up to three orders of magnitude. This validates the benefit of the proposed $M^*$ formulation of the PS constraint.

In Figure 8, the resulting optimal transfer trajectory of the OSC around the DSC (at the origin) is presented both in 3-D RTN space and projected in 2-D RN plane. The corresponding separation profiles are shown in Figures 9a and 9b. Looking at Figure 8, the controlled motion (blue bold line) is always instantaneously tangent to a safe uncontrolled trajectory (dashed lines), which is followed if the OSC loses control at that instant. Spacecraft separation along the uncontrolled trajectories is guaranteed to be $\geq 3\text{m}$ for at least 2 orbits, accounting for 3-$\sigma$ uncertainty (blue shaded ellipsoids). This is achieved in closed-loop, using the model predictive control-like implementation described in Eq. 16 and in Algorithm 1. In particular, the black dot-dashed "closed-loop plan" line is the trajectory planned on-board. The blue "controlled true" line is obtained by executing the control plan through actuation and propagating the resulting motion through the ground-truth dynamics. The difference between these two lines lies in the difference between the on-board dynamics model used for optimization and the ground-truth, as well as in the errors introduced by navigation and actuation. Similarly, the blue dashed "uncontrolled" trajectories are obtained integrating the ground-truth dynamics after each possible contingency. Whereas, the black dashed ones are predicted on-board and are superimposed to the 3-$\sigma$ ellipsoids modeling the uncertainty distributions used to quantify the bound $\beta_{kji}$. The true uncontrolled trajectories are contained within the 3-$\sigma$ ellipsoids implying that the bounds rightly capture and compensate for the difference between on-board belief and reality. This validates the use of the models presented in Eq. 10, 14, and 20. Note how the uncontrolled trajectories are characterized by very large uncertainty that has to be kept out of the origin in RN plane to have PS guarantees. This causes the algorithm to allocate a large bound ($\beta_{kji} \cdot \varepsilon$) on the minimum separation to be kept on the generate uncontrolled trajectory. The varying size of these quantified bound in relation to the size of the applied maneuvers can be appreciated by comparing the purple ($\varepsilon + \beta_{kji}$) dashed lines in Figures 9a and 9c with the control profile. Looking at Figure 9a, the black line is $S_{\text{R} \text{N}}^{kji}$, which assumes integrable dynamics after a contingency, the blue line accounts instead for ground-truth dynamics. $S_{\text{R} \text{N}}^{kji}$ is enforced being $\geq \varepsilon + \beta_{kji}$, while the bound $\beta_{kji}$ rightly capture the effects of non-integrable dynamics and uncertainty. This provides guarantee of PS (blue line $> \varepsilon$) and validates the bound model presented in Eq. 20. The total control cost of the transfer is $0.27\text{m/s}$. The cost of the TPBVP is $0.253\text{m/s}$. A difference of 10%, considering the included constraints and uncertainties, represents a very fuel efficient transfer, well within the
0.3 m/s allocated in the mission delta-V budget [37].

Finally, to compare with the closed-form analysis presented in Section V.A, Figure 9 shows the evolution over the transfer of $ac_1$, the magnitude of the vectors $ae_{34} / ae_{56}$, and their angular separation. The magnitude of $c_{34}$ and $c_{56}$ sizes the ellipses drawn by uncontrolled trajectories in RN plane (see Figure 5, right), their angular separation relates to PS through Eq. 32, whereas, $c_1$ represents an offset in radial direction of these ellipses (see Eq. 29). The trajectory computed using direct optimization achieves PS in RN plane either by having these ellipses centered around the origin with size $\geq \epsilon + \beta_k/H_l$, or, by decentering them in radial direction through a sufficiently large $|c_1|$ if their size is
< \epsilon + \beta_{kijl} \) (see Figure 8b). This analysis is validated in Figure 9. At the second maneuver, the reduction of the \( c_{34} \) and \( c_{56} \) norms requires \( c_1 \approx -80 \text{m} \) to achieve PS in RN plane. Comparing Figures 9e and 9f, when the norm of \( c_{34} \) or \( c_{56} \) becomes < \( \epsilon + \beta_{kijl} \), \( \delta_{kijl} \) shrinks so much that safe separation is not possible by centering the RN ellipse around the origin. Therefore, PS is achieved by properly decentering the ellipse in radial direction. The notable aspect is that this clear geometrical insight and deep connection with closed-form analysis is obtained using Algorithm 1 and the underlying convex optimization. This opens the way to much greater generalization and applicability than the specialized closed-form solutions.

2. Five-orbit Transfer

The five-orbit transfer reflects more closely what it is envisioned for the real mission. Differently from the two-orbit one, where the TPBVP solution was used as initial warm-start, now an initial open-loop design is made to achieve RN PS throughout the transfer using closed-form analysis under simplifying assumptions on the dynamics and neglecting system uncertainties \[36, 37\]. Table 4 presents the statistics of a Monte Carlo of 25 transfers realized by implementing the control profiles within the ground-truth dynamics accounting for actuation errors and in the presence of the various uncertainty sources introduced in Table 2. Looking at the top, when implemented in the ground-truth, the open-loop control design does not reach the desired terminal state (errors in along-track > 100m). Therefore, to retrieve closed-loop properties, tracking is needed. This is done in two different ways. The first, Table 4 on the bottom, uses state-of-the-art closed-form impulsive control solutions \[38\]. The second, Table 4 on the center, feeds the open-loop plan as initial warm-start to Algorithm 1 which refines it, and then tracks it using the logic presented in Section IV.B. These two approaches are compared in the following. The transfer trajectory obtained using Algorithm 1+ tracker is shown in Figure 10. The controlled motion (blue bold line) is always instantaneously tangent to a safe uncontrolled trajectory (dashed lines), which is followed if the spacecraft loses control at that instant. Spacecraft separation along the uncontrolled trajectories is guaranteed to be \( \geq 5 \text{m} \) for at least 2 orbits, accounting for 3-\( \sigma \) uncertainty (blue shaded ellipsoids). The level of uncertainty in the five-orbit transfer is lower than in the two-orbit one due to the smaller applied maneuvers, and consequently smaller actuation uncertainty.

The closed-form tracker \[38\] computes a correction to the open-loop control profile every 1.5 orbits to null the expected terminal state error. The correction is a maneuver sequence composed of three along-track and two cross-track maneuvers. Looking at Table 4 on the bottom, good performances in terms of average terminal state error and PS (\( \min(S^{RN}) \)) are retrieved, at a very small average computational cost (< 0.005s, for computing the correction over the subsequent 1.5 orbits). The benefits of this approach are the simplicity, explainability and computational efficiency. The main limit is that formal guarantees of PS are not provided. In particular, the closed-form tracker achieves PS by tracking closely the open-loop trajectory, designed to satisfy RN PS. The closed-loop trajectory is verified passively-safe a posteriori, but no explicit constrains are enforced during the closed-loop plan computation. In order to employ this algorithmic solution on-board it must be complemented with an additional logic that checks after each maneuver application the PS margin and, if needed, computes an escape maneuver if the safety limit is violated. Moreover, since the closed-loop tracker looks for correction maneuvers on-top of the open-loop plan, it achieves a total control cost which is \( \geq \) than the open-loop one.

As presented throughout this paper, Algorithm 1 instead looks for hard guarantees of PS. Using the open-loop design as a warm-start, Algorithm 1 looks for candidate maneuvers every 30deg plus in the locations of the open-loop design ones, for a total of \( n_M = 75 \) maneuver candidates over \( [t_0, t_f] \). As done for the two-orbit transfer, PS is enforced after

| Terminal state errors | \( c_{34} \) (m) | \( c_{56} \) (m) | \( c_{52} \) (m) | \( c_{53} \) (m) | \( c_{54} \) (m) | \( J \) (m/s) | \( \min(S^{RN}) \) (m) | time (s) | # calls | PS Guarantees |
|-----------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------|-----------------|-----------|---------|---------------|
| Open-loop design \[36, 37\] | mean 4.40 164.89 1.63 3.86 1.33 5.44 | \( 0.07 \) 1.84 0.13 0.21 0.16 0.16 | \( 0.06 \) 1.88 0.17 0.12 0.20 0.11 | \( 0.04 \) 0.37 | \( 0.01 \) 0.91 |
| Algorithm 1 Tracking (Sec. IV.B \[38\]) | mean 4.40 164.89 1.63 3.86 1.33 5.44 | \( 0.07 \) 1.84 0.13 0.21 0.16 0.16 | \( 0.06 \) 1.88 0.17 0.12 0.20 0.11 | \( 0.04 \) 0.37 | \( 0.01 \) 0.91 |
| Closed-form tracking \[35\] | mean 4.40 164.89 1.63 3.86 1.33 5.44 | \( 0.07 \) 1.84 0.13 0.21 0.16 0.16 | \( 0.06 \) 1.88 0.17 0.12 0.20 0.11 | \( 0.04 \) 0.37 | \( 0.01 \) 0.91 |
each maneuver application to guarantee safety both on the trajectory in between maneuvers and, in case of subsequent maneuver failure, afterwards up to \( t_f + T \). The time discretization at which PS must be enforced is at 1 deg, so 1800 samples on \([t_0, t_f]\) and 720 samples on \((t_f, t_f + T]\). The proposed efficient \( M^* \) formulation permits the reduction of the number of constraints required to enforce PS from \( > 75 \times 720 = 54000 \) to 75. The benefit from a computational standpoint is analogous to the one presented for the two-orbit transfer. Looking at Table 4 in the middle, Algorithm 1 achieves an average terminal state error \(~80\%\) smaller, and a total delta-V \(~20\%\) smaller, than the closed-form tracker. This is mainly due to the refinement of the initial warm-start over a larger optimization search space of maneuver candidates. In general, the benefits introduced by Algorithm 1 are the hard safety guarantees, high control accuracy performances, and flexibility of optimizing over a larger design space leading to a more fuel-efficient solution. The main limits are the lower explainability with respect to the closed-form solutions and the increased average computational cost. Looking at the run times presented in Table 4 using Algorithm 1 takes in average 3.5 s, whereas the tracker (Section IV.B 36) takes in average 1.5 s. Note that the run time depends on various SCP parameters and tolerances which can be tuned to trade with other performance parameters, such as the fuel optimality reached at convergence. The ratio of Algorithm 1 and tracking calls over the transfer is dictated by the high-level logic presented at the end of Section IV.B. Specifically, over the 75 decision points, Algorithm 1 is used once at the beginning and then another four times all of which are due to tracking unfeasibility. This shows that the main trigger in recomputing the "guidance" plan is the incapability to remain within the specified tracking bounds, and not having the predicted worst-case minimum RN separation lower than \( \epsilon \), which would entail a much greater risk from a safety perspective. Given the computational constraints for on-board implementation on the CubeSats microprocessors and looking at the average computational cost of using Algorithm 1 with respect to the tracking, it would be ideal to use Algorithm 1 just once at the beginning of the transfer and then track. In this sense, parameters such as the specified size of the tracking bound \((v^T_{ri})\) can be properly tuned to prevent unnecessary re-plannings and reduce the overall computational expense.

### B. Spacecraft Swarm in Eccentric Orbits

In this section, Algorithm 1 is assessed on the centralized open-loop trajectory design of a spacecraft swarm reconfiguration test case in highly eccentric orbit (HEO) around Earth. Three spacecraft have to reconfigure with respect to a virtual reference. The initial and target conditions are correspondingly a "high-density" (Eq. 37) and an RN separated (Eq. 38) swarm designs as defined in Section V.B. These conditions are depicted graphically in Figure 11 in integration constants (IC) space, and in Figures 12a and 12c in Cartesian RTN space, and further characterized in Table 5. Figures 12b and 12d show the spacecraft separations over the initial and target relative orbits. It is clear how the initial swarm configuration is safely separated in radial/tangential (RT) plane but not in radial/normal (RN), and the target one, vice versa, is safely separated in RN plane but not in RT.

A reconfiguration between the two designs while enforcing PS in 3-D is showcased, demonstrating a capability not published in literature. The reconfiguration horizon is fixed to two reference orbit periods, and PS has to be guaranteed in
3-D for at least one orbit ($T$), of at least 100m ($\epsilon$). This open-loop trajectory design is treated as deterministic, therefore no uncertainty sources are included. The spacecraft relative motion dynamics is modeled as Keplerian perturbed by Earth oblateness ($J_2$) and solar radiation pressure (SRP) [22]. The scenario is considered deterministic, therefore the only contribution to be included in the quantification of the bound $\beta_{k,ji}$ in Eq. [20] is the integrated effect of $J_2$+SRP over the PS horizon. Moreover, while in the presence of uncertainties, the quantification of this bound must be corrected in closed-loop within Algorithm 1. When just deterministic effects are included, the worst case expected effect of non-integrable dynamics over the PS horizon can be quantified a priori. This is done for this test case, and a constant PS bound ($\beta = 50$m) is considered. The spacecraft are modeled as continuously controlled by low-thrust actuation. To mimic the realizability of the control profile by the considered actuation a cap $\zeta$ on the control input $L_2$-norm is imposed. In particular, given a constant thrust and the spacecraft mass, using second Newton’s law, $\zeta_i = (\text{Thrust}/\text{Mass})\Delta t_i$, where $\Delta t_i$ is the maximum interval over which a computed delta-V must be realized by the low-thrust. This implies a convex
The control cost of the solution enforcing PS is
may exist a specific combination of

1. Results

For this test case, a comparison between PS and standard collision avoidance (CA) is presented. In particular, when enforcing CA, the horizon is \( T = 0 \) by definition. All the subroutines in Algorithm 1 related to the computation of \( t_{k_{j i l}} \) disappear, and the constraint is simply linearized in the SCP as done in literature [25, 26, 30]. To warm-start the CA SCP solver the TPBVP solution \[40\] is used. When enforcing PS instead, a safety horizon \( T \) (orbits) is used.

| Control cost | # Constraints | Running time | \( \epsilon \) | \( \min_{k_{j i l}} S_{k_{j i l}}^{2} \) |
|--------------|---------------|--------------|-----------|-------------------|
| PS 0.275 | \( n \sqrt{N_{i}N\alpha_{1}} \) | CA 34.5 | 100 | 188.2 | 109.2 |
| CA 0.270 | \( n \sqrt{N_{i}N\alpha_{1}} \) | TPBVP 2.8 | 100 | 248.3 | 57.6 (< \( \epsilon \)) |

The control cost of the solution enforcing PS is 2% larger than the one enforcing CA. This is a fuel efficient solution considering the additional number of constraints included. Looking at the run times in Table 6 on the center, enforcing constraint on the control input. The \( L_{2} \)-norm of the control input is minimized. The orbit is discretized at 30 deg, entailing a time-variant time step (\( \Delta t \)) given the spacecraft varying velocity along the eccentric orbits.

### Table 5 Highly eccentric orbit swarm test case

### Table 6 Comparison of passive safety (PS) and standard collision avoidance (CA)
PS is more computational expensive than enforcing CA. This larger computational expense is related to two factors. Firstly, the number of constraints to be enforced in each underlying SOCP, which is larger for PS \((n^2 N_a(N_a - 1)/2)\) than for CA \((n N_a(N_a - 1)/2)\). Secondly, the computations in Algorithm 1 (lines 9-24) required for enforcing PS given an initial warm-start that is not necessarily feasible (e.g., that satisfies CA but not PS). The enforcement trend is the one presented in Figure 13. In particular, the further the initial warm-start is from being feasible, the more inner-loop iterations are needed, the more solutions of SOCP instances 40 including \(n^2 N_a(N_a - 1)/2\) constraints are required, the higher overall computational cost is. Consequently, a bad initial warm-start has a substantial impact. Looking at Table 6 this is further proved by feeding-back as a warm-start a previously found passively-safe solution. Now, Algorithm 1 is \(~5\) times faster. This test case underlines the importance of properly warm-starting SCP-based algorithms to reduce their computational expense, in particular for on-board execution.
VII. Conclusion

This paper has addressed the problem of enforcing motion safety guarantees even in the event of sudden loss of control capabilities by any agent within a multi-agent system. The paper shows that, to enforce these so-called passive safety guarantees, point-evaluated constraints on the trajectory following a contingency can be replaced with a single function of the integration constants of the equations of motion at contingency instants. Furthermore, effects of uncertainties on the trajectory can be compensated for through the method of variation of parameters. This permits reducing the number of required constraints by one polynomial degree in the number of discrete time samples, and allows computationally efficient enforcement within a multi-agent optimal control problem, solvable using direct methods. Using this novel methodology, experimental results show that a control profile enforcing passive safety in closed-loop in a multi-agent setting can be computed efficiently through an algorithm based on sequential convex programming and model predictive control. This allows the computation of passively-safe spacecraft reconfigurations in orbital scenarios outside of the assumption limits of the current literature, in both near-circular and eccentric orbits, in the presence of realistic system uncertainties stemming from sensing, actuation, and unmodeled system dynamics. In addition, the application of the proposed approach to the spacecraft relative motion dynamics permits the derivation of novel solutions for multi-spacecraft passive safety in eccentric orbits, with application to relative motion design and guidance. The methodology presented in this paper opens the way to efficient, safe and fault-tolerant real-time motion planning and control on-board, which is the core requirement for autonomous functions in future distributed space systems’ missions.

VIII. Appendix

A. Passive Safety Constraint Requirements

In the following, formal proof of the number of constraints required to enforce PS reported in Table 1 is developed. The proof is developed for the "impulsive" control type, it is left to the reader verifying that for $n_M \rightarrow n$, the number of constraints required in impulsive control tends to the ones required in continuous control. The number of active agents’ combinations is $N_a(N_a - 1)/2$, where $N_a$ is the number of active agents. $N_p$ is the number of permanently passive agents.

Proposition. Assume the interval $[t_0, t_f]$ is discretized in $n$ samples, and the interval $(t_f, t_f + T]$ in $m$ samples. Assume impulsive control application is planned and possibly failed just in a subset $n_M \leq n$ of the $n$ instants, such as $n/n_M \in \mathbb{N}_{>0}$. In order to guarantee PS (as in Definition 1 in Section III) at each of the $n$ and $m$ instants in $[t_0, t_f + T]$, using the $M$ formulation, the total number of collision avoidance constraints the multi-agent system has to enforce is: $[((n/n_M)(\sum_{i=1}^{n_M} i^2) + (n_M + 1)^2m)N_a(N_a - 1)/2 = ((n/n_M)(n_M(n_M + 1)(2n_M + 1)/6) + (n_M + 1)^2m)N_a(N_a - 1)/2$, with order of magnitude: $O((n_M + 1)^2mN_a^2)$. Whereas, between active and permanently passive agents is: $[((n/n_M)(\sum_{i=1}^{n_M} i) + (n_M + 1)m)N_aN_p = (n/n_M)(n_M(n_M + 1)/2) + (n_M + 1)m)N_aN_p$, with order of magnitude: $O((n + m)n_MN_aN_p^2)$. Using the $M^+$ formulation, the total number of IC-dependent functions to be constrained between active agents is: $n_M^2N_a(N_a - 1)/2$. Whereas, between active and permanently passive agents it is: $n_MN_aN_p$.

Proof. Looking at Figure 2b, in between each of the instants $t_i \in [t_0, t_f]$, $i = 1, \ldots n_M$, of planned impulsive control application, the number of separations’ combinations to be constrained is $(n/n_M)^2N_a(N_a - 1)/2$ between active agents, and $(n/n_M)N_aN_p$ between active and permanently passive agents. The factor $(n/n_M)$ accounts for the need of guaranteeing safe separation in between the impulses, too. Instead on $(t_f, t_f + T]$, at each of the $m$ instants, the number of separation combinations to be constrained is $(n_M + 1)^2N_a(N_a - 1)/2$ between active agents, and $(n_M + 1)N_aN_p$ between active and permanently passive ones. The +1 in the $n_M + 1$ factor accounts for the case in which control is not lost on $[t_0, t_f]$, but the reached target configuration has to be guaranteed safe with respect to the uncontrolled trajectory of another agent. This implies that the total number of separation combinations to be constrained over $[t_0, t_f + T]$ to have guarantee of PS between active agents is: $[(n/n_M)(\sum_{i=1}^{n_M} i^2) + (n_M + 1)^2m]N_a(N_a - 1)/2$, with order of magnitude: $O((n_M + 1)^2mN_a^2)$. Whereas, between active active and permanently passive agents is: $[(n/n_M)(\sum_{i=1}^{n_M} i) + (n_M + 1)m]N_aN_p$, with order of magnitude: $O((n + m)n_MN_aN_p^2)$. The total number of IC-dependent functions to be constrained equates the total number of IC states’ combinations involved in Eq. (13) which equates the number of contingency instants’ combinations. Therefore, to enforce PS between active agents, the total number of IC states’ combinations to be constrained is: $n_M^2N_a(N_a - 1)/2$. Whereas, between active active and permanently passive agents is: $n_MN_aN_p$. This concludes the proof. □
B. Relative Motion Dynamics

The map between the integration constants (IC) used in this paper and the IC of the Yamanaka-Ankersen (YA) solution of relative motion in eccentric orbits is

\[
M_{YA}(a) = \begin{bmatrix}
0 & 0 & 0 & -2 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 \\
0 & -\sin(\omega) & \cos(\omega) & 0 & 0 & 0 \\
0 & \cos(\omega) & \sin(\omega) & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -\cos(\omega) & -\sin(\omega) \\
0 & 0 & 0 & 0 & -\sin(\omega) & \cos(\omega)
\end{bmatrix}
\] (39)

Moreover, the IC used in this paper are linked to the quasi-nonsingular ROE through the first order map

\[
c_k \approx M_{CW}(a) \begin{bmatrix}
\eta^2 & 0 & 0 & 0 & 0 \\
0 & \frac{1}{\eta} & 0 & \frac{1}{\eta} & \sin(\omega) \\
0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1
\end{bmatrix} \begin{bmatrix}
\frac{(a_k - a)}{a} \\
\omega_k + M_k(t) - \omega - M(t) + (\Omega_k - \Omega) \cos(i) \\
e_k \cos(\omega_k) - e \cos(\omega) \\
e_k \sin(\omega_k) - e \sin(\omega) \\
\Omega_k - \Omega \sin(i)
\end{bmatrix}
\] (40)

where

\[
M_{CW}(a) = \begin{bmatrix}
\eta^2 & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{1}{\eta} & 0 & \frac{1}{\eta} & \sin(\omega) & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & \eta^2
\end{bmatrix}
\] (41)

and, in the ROE, \(M(t)\) is the mean anomaly at instant \(t\). The center matrix in Eq. 40 back-propagates the ROE to initial integration instant \((t = 0)\) through the integrable dynamics. For \(e \to 0\), \(M_{CW}\) becomes identity (the singularities cancel out), and the IC used in this paper reduce to the quasi-nonsingular ROE and to the Clohessy-Wiltshire (CW)’s IC [16].

Acknowledgments

This work was supported by the William R. and Sara Hart Kimball Stanford Graduate Fellowship, and by the VISORS’s mission NSF Award #1936663. The authors are thankful for their support.

References

[1] D’Amico, S., Pavone, M., Saraf, S., Alhussien, A., Al-Saud, T., Buchman, S., Byer, R., and Farhat, C., “Miniaturized Autonomous Distributed Space System for Future Science and Exploration,” 8th International Workshop on Satellite Constellations and Formation Flying, Delft, The Netherlands, 2015, pp. 1–20.

[2] Jacklin, S. A., “Small-satellite mission failure rates,” Tech. Rep. TM-2018-220034, NASA, 2019. URL https://ntrs.nasa.gov/citations/20190002789

[3] Dezfuli, H., Benjamin, A., Everett, C., Maggio, G., Stamatelatos, M., Youngblood, R., Guarro, S., Rutledge, P., Sherrard, J., Smith, C., and Williams, R., “NASA Risk Management Handbook. Version 1.0,” 2011. URL https://ntrs.nasa.gov/citations/20120000033

[4] Scharf, D. P., Açıkmeşe, B., Ploen, S. R., and Hadaegh, F. Y., “Three-dimensional re-active collision avoidance with multiple colliding spacecraft for deep-space and earth-orbiting formations,” Fourth International Conference on Spacecraft Formation Flying Missions and Technologies, 2011, pp. 1–4.

[5] Starek, J. A., Schmerling, E., Maher, G. D., Barbee, B. W., and Pavone, M., “Fast, Safe, Propellant-Efficient Spacecraft Motion Planning Under Clohessy-Wiltshire-Hill Dynamics,” Journal of Guidance, Control, and Dynamics, Vol. 40, No. 2, 2017, pp. 418–438. https://doi.org/10.2514/1.G001913

28
[6] Hubert, S., and Swale, J., “Stationkeeping of a constellation of geostationary communications satellites,” *Astrodynamics Conference*, 1984, p. 2042. [https://doi.org/10.2514/6.1984-2042](https://doi.org/10.2514/6.1984-2042)

[7] D’Amico, S., and Montenbruck, O., “Proximity Operations of Formation-Flying Spacecraft Using an Eccentricity/Inclination Vector Separation,” *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 3, 2006, pp. 554–563. [https://doi.org/10.2514/1.15114](https://doi.org/10.2514/1.15114)

[8] Breger, L., and How, J. P., “Safe Trajectories for Autonomous Rendezvous of Spacecraft,” *Journal of Guidance, Control, and Dynamics*, Vol. 31, No. 5, 2008, pp. 1478–1489. [https://doi.org/10.2514/1.29590](https://doi.org/10.2514/1.29590)

[9] Barbee, B. W., Carpenter, J. R., Heatwole, S., Markley, F. L., Moreau, M., Naasz, B. J., and Van Eepeol, J., “A Guidance and Navigation Strategy for Rendezvous and Proximity Operations with a Noncooperative Spacecraft in Geosynchronous Orbit,” *Journal of the Aerospace Sciences*, Vol. 58, No. 3, 2011, pp. 389–408. [https://doi.org/10.1007/BF03321176](https://doi.org/10.1007/BF03321176)

[10] Woffinden, D. C., and Geller, D. K., “Navigating the Road to Autonomous Orbital Rendezvous,” *Journal of Spacecraft and Rockets*, Vol. 44, No. 4, 2007, pp. 898–909. [https://doi.org/10.2514/1.30734](https://doi.org/10.2514/1.30734)

[11] D’Errico, M., *Distributed space missions for earth system monitoring*, Vol. 31, Springer Science & Business Media, 2012. URL [https://link.springer.com/book/10.1007/978-1-4614-4541-8#toc](https://link.springer.com/book/10.1007/978-1-4614-4541-8#toc) Chaps. 13, 19, 21.

[12] Koenig, A. W., and D’Amico, S., “Robust and Safe N-Spacecraft Swarming in Perturbed Near-Circular Orbits,” *Journal of Guidance, Control, and Dynamics*, Vol. 41, No. 8, 2018, pp. 1643–1662. [https://doi.org/10.2514/1.G003249](https://doi.org/10.2514/1.G003249)

[13] Lakshmikantham, V., *Method of Variation of Parameters for Dynamic Systems*, Routledge, 1998. [https://doi.org/10.1201/9780203747452](https://doi.org/10.1201/9780203747452) Chap. 1.

[14] Vallado, D. A., *Fundamentals of Astrodynamics and Applications*, Space Technology Library, 2013. URL [https://link.springer.com/book/97807892369035](https://link.springer.com/book/97807892369035) Chaps. 2, 9.

[15] Clohessy, W. H., and Wiltshire, R. S., “Terminal Guidance System for Satellite Rendezvous,” *Journal of the Aerospace Sciences*, Vol. 27, No. 9, 1960, pp. 653–658. [https://doi.org/10.2514/8.8704](https://doi.org/10.2514/8.8704)

[16] D’Amico, S., “Relative Orbital Elements as Integration Constants of Hills Equations,” *TN05-08*, German Space Operations Centre (GSOC), Munich, Germany, 2005.

[17] Yamanaka, K., and Ankersen, F., “New State Transition Matrix for Relative Motion on an Arbitrary Elliptical Orbit,” *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 1, 2002, pp. 60–66. [https://doi.org/10.2514/2.4875](https://doi.org/10.2514/2.4875)

[18] Sullivan, J., and D’Amico, S., “Nonlinear Kalman Filtering for Improved Angles-Only Navigation Using Relative Orbital Elements,” *Journal of Guidance, Control, and Dynamics*, 2017, pp. 1–18. [https://doi.org/10.2514/1.G002719](https://doi.org/10.2514/1.G002719)

[19] Gim, D.-W., and Alfriend, K. T., “State Transition Matrix of Relative Motion for the Perturbed Noncircular Reference Orbit,” *Journal of Guidance, Control, and Dynamics*, Vol. 26, No. 6, 2003, pp. 956–971. [https://doi.org/10.2514/2.6924](https://doi.org/10.2514/2.6924)

[20] Koenig, A. W., Guffanti, T., and D’Amico, S., “New State Transition Matrices for Spacecraft Relative Motion in Perturbed Orbits,” *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 7, 2017, pp. 1749–1768. [https://doi.org/10.2514/1.G002409](https://doi.org/10.2514/1.G002409)

[21] Guffanti, T., D’Amico, S., and Lavagna, M., “Long-Term Analytical Propagation of Satellite Relative Motion in Perturbed Orbits,” 27th AAS/AIAA Space Flight Mechanics Meeting, San Antonio, TX, 2017, pp. 1–31.

[22] Guffanti, T., and D’Amico, S., “Linear Models for Spacecraft Relative Motion Perturbed by Solar Radiation Pressure,” *Journal of Guidance, Control, and Dynamics*, Vol. 42, No. 9, 2019, pp. 1962–1981. [https://doi.org/10.2514/1.G002822](https://doi.org/10.2514/1.G002822)

[23] Chernick, M., “Optimal Impulsive Control of Spacecraft Relative Motion,” Ph.D. thesis, Stanford University, 2021. URL [https://purl.stanford.edu/vw016ts7713](https://purl.stanford.edu/vw016ts7713).

[24] Byrd, R. H., Gilbert, J. C., and Nocedal, J., “A Trust Region Method Based on Interior Point Techniques for Nonlinear Programming,” *Mathematical Programming*, Vol. 89, No. 1, 2000, pp. 149–185. [https://doi.org/10.1007/PL00011391](https://doi.org/10.1007/PL00011391)

[25] Liu, X., and Lu, P., “Solving Nonconvex Optimal Control Problems by Convex Optimization,” *Journal of Guidance, Control, and Dynamics*, Vol. 37, No. 3, 2014, pp. 750–765. [https://doi.org/10.2514/1.62110](https://doi.org/10.2514/1.62110)

[26] Malyuta, D., Reynolds, T. P., Szmuk, M., Lew, T., Bonalli, R., Pavone, M., and Açıkmeşe, B., “Convex Optimization for Trajectory Generation: A Tutorial on Generating Dynamically Feasible Trajectories Reliably and Efficiently,” *IEEE Control Systems Magazine*, Vol. 42, No. 5, 2022, pp. 40–113. [https://doi.org/10.1109/MCS.2022.3187542](https://doi.org/10.1109/MCS.2022.3187542)
[27] Singh, S., “Robust Control, Planning, and Inference for Safe Robot Autonomy,” Ph.D. thesis, Stanford University, 2019. URL https://purl.stanford.edu/pr731qc2534

[28] Carpenter, J. R., and D’Souza, C. N., “Navigation filter best practices,” Tech. Rep. TP-2018-219822, NASA, 2018. URL https://ntrs.nasa.gov/citations/20180003657

[29] Borrelli, F., Bemporad, A., and Morari, M., Predictive control for linear and hybrid systems, Cambridge University Press, 2017. https://doi.org/10.1017/9781139061759 Chap. 12.

[30] Morgan, D., Chung, S.-J., and Hadaegh, F. Y., “Model Predictive Control of Swarms of Spacecraft Using Sequential Convex Programming,” Journal of Guidance, Control, and Dynamics, Vol. 37, No. 6, 2014, pp. 1725–1740. URL https://doi.org/10.2514/1.G000218

[31] Zhang, Y., and Jiang, J., “Bibliographical review on reconfigurable fault-tolerant control systems,” Annual Reviews in Control, Vol. 32, No. 2, 2008, pp. 229–252. https://doi.org/10.1016/j.arcontrol.2008.03.008

[32] Semsar-Kazerooni, E., and Khorasani, K., “Team Consensus for a Network of Unmanned Vehicles in Presence of Actuator Faults,” IEEE Transactions on Control Systems Technology, Vol. 18, No. 5, 2010, pp. 1155–1161; https://doi.org/10.1109/TCST.2009.2032921

[33] Guffanti, T., and D’Amico, S., “Integration Constants as State Variables for Optimal Path Planning,” 2018 European Control Conference (ECC), Limassol, Cyprus, 2018, pp. 1–6. https://doi.org/10.23919/ECC.2018.8550448

[34] Guffanti, T., and D’Amico, S., “Multi-Agent Passive Safe Optimal Control using Integration Constants as State Variables,” AIAA Scitech 2021 Forum, 2021, pp. 1–28. https://doi.org/10.2514/6.2021-1101

[35] Guffanti, T., and D’Amico, S., “Robust Passively Safe Spacecraft Swarming via Closed-form and Optimization-based Control Approaches,” 2022 American Control Conference (ACC), Atlanta, GA, USA, 2022, pp. 416–423. https://doi.org/10.23919/ACC53348.2022.9867216

[36] Guffanti, T., “Optimal Passively-Safe Control of Multi-Agent Motion with Application to Distributed Space Systems,” Ph.D. thesis, Stanford University, 2022. URL https://purl.stanford.edu/gh147jp5825

[37] Koenig, A. W., D’Amico, S., and Lightsey, E. G., “Formation flying orbit and control concept for the VISORS mission,” AIAA Scitech 2021 Forum, 2021, pp. 1–22. https://doi.org/10.2514/6.2021-0423

[38] Gaias, G., and D’Amico, S., “Impulsive Maneuvers for Formation Reconfiguration Using Relative Orbital Elements,” Journal of Guidance, Control, and Dynamics, Vol. 38, No. 6, 2015, pp. 1036–1049. https://doi.org/10.2514/1.G000189

[39] Boyd, S., and Vandenberghe, L., Convex Optimization, Cambridge University Press, 2004. URL https://web.stanford.edu/~boyd/cvxbook/ Chaps. 4, 11.

[40] Domahidi, A., Chu, E., and Boyd, S., “ECOS: An SOCP solver for embedded systems,” European Control Conference (ECC), 2013, pp. 3071–3076. https://doi.org/10.23919/ECC.2013.6669541

[41] Giralo, V., “Precision Navigation of Miniaturized Distributed Space Systems using GNSS,” Ph.D. thesis, Stanford University, 2021. URL https://purl.stanford.edu/dp334dn5706

[42] Casotto, S., “The Equations of Relative Motion in the Orbital Reference Frame,” Celestial Mechanics and Dynamical Astronomy, Vol. 124, No. 3, 2016, pp. 215–234, https://doi.org/10.1007/s10569-015-9660-1

[43] Lightsey, E. G., Stevenson, T., and Sorgenfrei, M., “Development and Testing of a 3-D-Printed Cold Gas Thruster for an Interplanetary CubeSat,” Proceedings of the IEEE, Vol. 106, No. 3, 2018, p. 379–390. https://doi.org/10.1109/JPROC.2018.2799898

[44] Sullivan, J., “Nonlinear Angles-Only Orbit Estimation for Autonomous Distributed Space Systems,” Ph.D. thesis, Stanford University, 2020. URL https://purl.stanford.edu/yyx801sv8482

[45] Tapley, B. D., Bettadpur, S., Watkins, M., and Reigber, C., “The Gravity Recovery and Climate Experiment: Mission Overview and Early Results,” Geophysical Research Letters, Vol. 31, No. 9, 2004, pp. 1–4. https://doi.org/10.1029/2004GL019779

[46] Picone, J. M., Hedin, A. E., Drob, D. P., and Aikin, A. C., “NRLMSISE-00 empirical model of the atmosphere: Statistical comparisons and scientific issues,” Journal of Geophysical Research: Space Physics, Vol. 107, No. A12, 2002, pp. SIA–15. https://doi.org/10.1029/2002JA009430