Co-Optimization of Navigation System Requirements and Trajectory Design Using a Sweeping Gradient Method and Linear Covariance Analysis

Benjamin W. L. Margolis
NASA Ames Research Center, Moffett Field, CA, 94035

David Woffinden
NASA Johnson Space Center, Houston TX, 77058

We describe the application of a sweeping gradient method for ordinary differential equations with events (SGM) and linear covariance analysis (LinCov) to the co-optimization of navigation system requirement generation and robust trajectory design. SGM is a method for computing the gradient of trajectory analyses defined by performance indices over initial value problems with events with respect to static parameters. LinCov is an analytic technique for predicting stochastic behavior of dynamical systems. By combining SGM with LinCov, it is possible to use efficient, off-the-shelf, gradient-based optimizers to solve a combined robust optimal trajectory and navigation system design problem. In this paper, we formulate the required models to apply the combined SGM and LinCov techniques to a Near-Rectilinear Halo Orbit rendezvous approach scenario and show results for several intermediate problems.

I. Introduction

It is not uncommon that the trajectory design and the GN&C system requirement generation, particularly for the navigation system, are considered independent tasks addressed by different and unrelated teams. A trajectory profile is designed ignoring the GN&C system that will actually execute it, while the GN&C system requirements are derived to achieve the desired profile. Over the design cycle, this process eventually proves to be iterative in nature where modifications to the trajectory are considered to accommodate constraints imposed by the actual GN&C system developed. Unfortunately, these changes typically occur later in the design cycle where the cost of modifying the trajectory has significant impacts in cost, schedule, and resources compared to those adjustments being proposed earlier in the process. A process that could optimally derive both the trajectory and the GN&C system during the preliminary design phase could potentially provide significant development cost savings while increasing design robustness and enhancing both crew safety and mission success.

This paper proposes a process to achieve this ideal by combining a sweeping gradient method for ordinary differential equations with events (SGM) and linear covariance analysis (LinCov) for the co-optimization of navigation system requirement generation and robust trajectory design. SGM is a method for computing the gradient of trajectory analyses defined by performance indices over initial value problems with events with respect to static parameters[1]. LinCov is an analytic technique for predicting stochastic behavior of dynamical systems[2, 3]. By combining SGM and LinCov, it is possible use efficient, off-the-shelf, gradient-based optimizers to solve a combined robust optimal trajectory and navigation system design problem. A rendezvous scenario in a Near-Rectilinear Halo Orbit (NRHO) is utilized to demonstrate how SGM and LinCov can be combined to optimize both the trajectory design and the corresponding navigation requirements simultaneously.

Previous research efforts have developed automated techniques to derive navigation requirements[4] and later extended those concepts to optimally derive multiple GN&C requirements with multiple constraints[5], but in both cases, the process assumed a single trajectory profile that could not be altered. Conversely, non-traditional robust trajectory optimization techniques have been utilized to design trajectory profiles that account for navigation errors, disturbance accelerations, maneuver execution errors, and orbit insertion perturbations; but the uncertainty parameters, such as the navigation errors, were fixed[6–15]. This paper introduces new techniques that optimize both the trajectory design and the GN&C requirements at the same time. Section II gives an overview of the performance metrics and co-optimization techniques. Section III provides detail to the NRHO rendezvous and docking concept of operations. Section IV generates results by applying SGM and LinCov to co-optimize both the rendezvous trajectory profile and

the proposed navigation requirements. Lastly, Section V summarizes the research discoveries and impacts of these developments on the trajectory design and GN&C requirement process.

II. Co-Optimization Techniques Using SGM and LinCov

A. Performance Metrics

To develop the techniques of robust trajectory optimization using a sweeping gradient method and linear covariance analysis, there are several performance metrics that must be defined which include the true trajectory dispersions $\delta\mathbf{x}$, the navigation dispersions $\delta\hat{\mathbf{x}}$, the true navigation error $\delta\mathbf{e}$, and the onboard navigation error $\delta\hat{\mathbf{e}}$ as depicted in Figure 1.

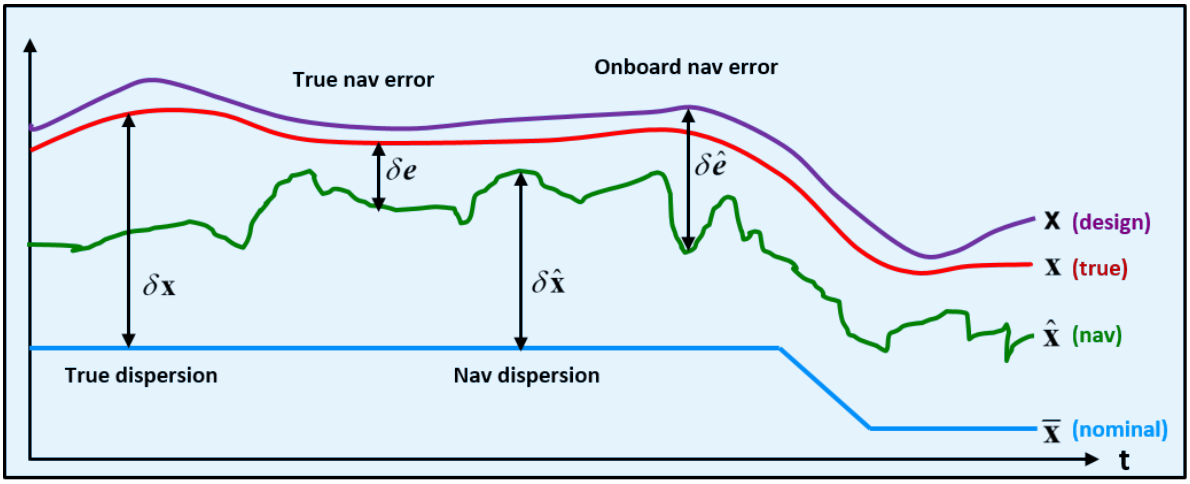


Fig. 1 GN&C Performance Metric Variables

The true dispersions $\delta\mathbf{x}$ are defined as the difference between the true state $\mathbf{x}$ and the nominal state $\bar{\mathbf{x}}$. The true state $\mathbf{x}$ is an n-dimensional vector that represents the *real world* environment or the *realization* of the actual state.

$$\delta\mathbf{x} \triangleq \mathbf{x} - \bar{\mathbf{x}} \qquad \mathbf{D} = E\left[\delta\mathbf{x}\delta\mathbf{x}^{\mathrm{T}}\right] \tag{1}$$

The nominal state $\bar{\mathbf{x}}$ is also an n-dimensional vector that represents the desired or reference state. The covariance of the environment dispersions, $\mathbf{D}$, indicates how precisely the system can follow a desired trajectory.

The navigation dispersions $\delta\hat{\mathbf{x}}$ are defined as the difference between the navigation state $\hat{\mathbf{x}}$ and the nominal state $\bar{\mathbf{x}}$. The navigation state is an $\hat{n}$-dimensional vector ($\hat{n} < n$) that represents the filter's estimated state.

$$\delta\hat{\mathbf{x}} \triangleq \hat{\mathbf{x}} - \mathbf{M}_x\bar{\mathbf{x}} \qquad \hat{\mathbf{D}} = E\left[\delta\hat{\mathbf{x}}\delta\hat{\mathbf{x}}^{\mathrm{T}}\right] \tag{2}$$

The matrix $\mathbf{M}_x$ is an $(\hat{n} \times n)$ matrix that maps the estimated state in terms of the true and nominal state. The covariance of the navigation dispersions, $\hat{\mathbf{D}}$, reflects how precisely the onboard system thinks it can follow a prescribed reference trajectory.

The true navigation error $\delta\mathbf{e}$ is the difference between the environment and navigation states. It is also the difference between the environment and the navigation dispersions.

$$\delta\mathbf{e} \triangleq \mathbf{M}_x\mathbf{x} - \hat{\mathbf{x}} = \mathbf{M}_x\delta\mathbf{x} - \delta\hat{\mathbf{x}} \qquad \mathbf{P} = E\left[\delta\mathbf{e}\delta\mathbf{e}^{\mathrm{T}}\right] \tag{3}$$

The covariance of the true navigation error, $\mathbf{P}$, quantifies how precisely the onboard navigation system can estimate the actual state.

The onboard navigation error $\delta\hat{\mathbf{e}}$ itself is never computed, but it is used to develop the onboard navigation filter equations. It is defined as the difference between the design state, $\mathbf{x}$, and the navigation state $\hat{\mathbf{x}}$.

$$\delta\hat{\mathbf{e}} \triangleq \mathbf{x} - \hat{\mathbf{x}} \qquad \hat{\mathbf{P}} = E\left[\delta\hat{\mathbf{e}}\delta\hat{\mathbf{e}}^{\mathrm{T}}\right] \tag{4}$$

The covariance of the onboard navigation error, $\hat{\mathbf{P}}$, quantifies how precisely the onboard navigation system expects it can determine the actual state. The performance of the onboard navigation system is determined by comparing $\hat{\mathbf{P}}$ to the actual navigation performance $\mathbf{P}$. If the *true* states $\mathbf{x}$ and the *design* states $\mathbf{x}$ are assumed to be the same, then the true navigation covariance will equal the onboard navigation covariance.

The covariances of the true dispersions, navigation dispersions, true navigation error, and the onboard navigation error are ultimately used to analyze and assess the performance of a proposed GN&C system. A common approach to obtain these performance metrics is to use a Monte Carlo simulation as outlined in Figure 2, where the sample statistics of hundreds or thousands of runs, N, are used to numerically compute the desired covariance matrices.

$$\mathbf{D} \approx \frac{1}{N-1}\sum \delta\mathbf{x}\delta\mathbf{x}^{\mathrm{T}} \qquad \hat{\mathbf{D}} \approx \frac{1}{N-1}\sum \delta\hat{\mathbf{x}}\delta\hat{\mathbf{x}}^{\mathrm{T}} \qquad \mathbf{P} \approx \frac{1}{N-1}\sum \delta\mathbf{e}\delta\mathbf{e}^{\mathrm{T}} \tag{5}$$

The onboard navigation error covariance $\hat{\mathbf{P}}$ is the navigation filter covariance for each run. This same statistical information can be obtained using linear covariance analysis techniques.

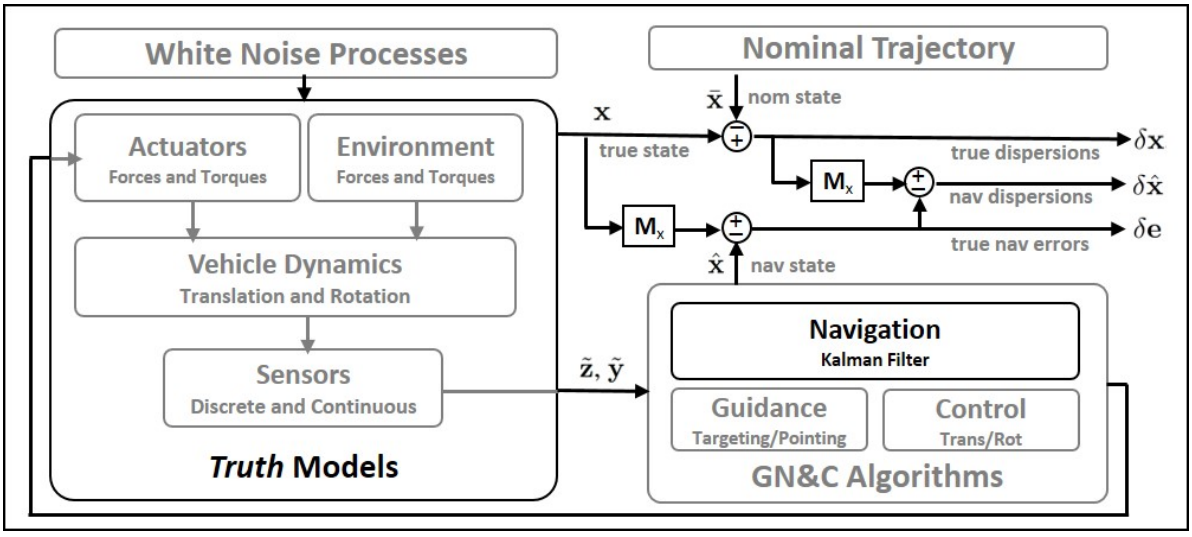


Fig. 2 Extracting GN&C Performance Metrics Using Monte Carlo Techniques

B. Linear Covariance Analysis

In comparison to the hundreds or thousands of runs typically required for Monte Carlo analaysis, LinCov utilizes augmented and onboard state covariance matrices to produce the statistical performance metrics of a closed-loop GN&C system in a single-run. Linear covariance analysis incorporates the non-linear system dynamics models and the non-linear GN&C algorithms to generate a nominal reference trajectory $\bar{\mathbf{x}}$ which is then used to propagate, update, and correct an onboard navigation covariance matrix $\hat{\mathbf{P}}$ and an augmented state covariance matrix $\mathbf{C}$,

$$\mathbf{C} = E\left[\delta\mathbf{X}\delta\mathbf{X}^{\mathrm{T}}\right] \tag{6}$$

where the augmented state $\delta\mathbf{X}^{\mathrm{T}} = [\delta\mathbf{x}^{\mathrm{T}}\ \delta\hat{\mathbf{x}}^{\mathrm{T}}]$ consists of the true dispersions and the navigation dispersions. Pre- and post-multiplying the augmented state covariance matrix by the following mapping matrices results in the covariances for the trajectory dispersions, navigation dispersions, and the navigation error:

$$\begin{aligned}
\mathbf{D} &= [\ \mathbf{I}_{n\times n},\ \mathbf{0}_{n\times\hat{n}}\ ]\,\mathbf{C}\,[\ \mathbf{I}_{n\times n},\ \mathbf{0}_{n\times\hat{n}}\ ]^{\mathrm{T}} \\
\hat{\mathbf{D}} &= [\ \mathbf{0}_{\hat{n}\times n},\ \mathbf{I}_{\hat{n}\times\hat{n}}\ ]\,\mathbf{C}\,[\ \mathbf{0}_{\hat{n}\times n},\ \mathbf{I}_{\hat{n}\times\hat{n}}\ ]^{\mathrm{T}} \\
\mathbf{P} &= [\ \mathbf{I}_{\hat{n}\times n},\ -\mathbf{I}_{\hat{n}\times\hat{n}}\ ]\,\mathbf{C}\,[\ \mathbf{I}_{\hat{n}\times n},\ -\mathbf{I}_{\hat{n}\times\hat{n}}\ ]^{\mathrm{T}}
\end{aligned} \tag{7}$$

For details regarding the development and implementation of the linear covariance simulation, see the following references[3, 16].

C. Sweeping Gradient Method

This subsection provides a brief overview and references to the sweeping gradient methods [1] and how they are applied for this research.

We assume each component J_i of the trajectory analysis output $\boldsymbol{J}$ is defined by

$$J_i = \phi_i \left(t_f, \boldsymbol{\xi}\left(t_f\right); \boldsymbol{\theta}\right) + \int_{t_0}^{t_f} L_i\left(t, \boldsymbol{\xi}\left(t\right); \boldsymbol{\theta}\right)\, dt$$

for $i = 1, \ldots, n_J$ where ϕ_i is a terminal function and L_i is an on-going function over a trajectory defined by a parameterized initial value problem with events. Every component J_i depends on the same trajectory $\boldsymbol{\xi}\left(t\right)$ with initial condition

$$\boldsymbol{\xi}\left(t_0\right) = \mathbf{h}_0\left(t_0; \boldsymbol{\theta}\right)$$

with the ordinary differential equation governing the continuous evolution of the state is given by

$$\dot{\boldsymbol{\xi}}\left(t\right) = \mathbf{f}\left(t, \boldsymbol{\xi}\left(t\right); \boldsymbol{\theta}\right) \quad .$$

An event i occurs at the zero-crossing of an its event function

$$\gamma_i\left(t_k^-, \boldsymbol{\xi}\left(t_k^-\right); \boldsymbol{\theta}\right) = 0$$

at which point its update equation $\mathbf{h}_i$ is applied to the state so that

$$\boldsymbol{\xi}\left(t_k^+\right) = \mathbf{h}_i\left(t_k^-, \boldsymbol{\xi}\left(t_k^-\right); \boldsymbol{\theta}\right),$$

enabling the modeling of discontinuous changes to the state or dynamics.

The continuous-time SGM method emerged during early analysis of differential equations and was used through the 1960s to solve optimal control problems. The SGM for a trajectory analysis for an initial value problem without events involves a four step process:

1) Evaluate the trajectory by solving IVP with the given set of parameters
2) Evaluate the performance index of interest for the given set of parameters and trajectory.
3) Solve the differential equation adjoint to the trajectory model *and each performance index*, given by

$$\begin{aligned}
\dot{\boldsymbol{\lambda}}\left(t\right) &= -\left[\frac{\partial \mathbf{f}}{\partial \boldsymbol{\xi}}\left(t, \boldsymbol{\xi}\left(t\right); \boldsymbol{\theta}\right)\right]^T \boldsymbol{\lambda}\left(t\right) - \left[\frac{\partial L_i}{\partial \boldsymbol{\xi}}\left(t, \boldsymbol{\xi}\left(t\right); \boldsymbol{\theta}\right)\right]^T \\
\boldsymbol{\lambda}\left(t_f\right) &= \frac{\partial \phi_i}{\partial \boldsymbol{\xi}}\left(t_f, \boldsymbol{\xi}\left(t_f\right); \boldsymbol{\theta}\right)
\end{aligned}$$

   where the forcing function $\frac{\partial L_i}{\partial \xi}$ and terminal condition $\frac{\partial \phi_i}{\partial \xi}$ induce a unique adjoint trajectory for each performance index J_i. Notice also that the adjoint trajectory depends on the particular trajectory determined by the parameters θ as solved in Step 1.
4) Evaluate the gradient of the performance index with respect to the parameters by computing

$$\frac{\partial J_i}{\partial \boldsymbol{\theta}} = \int_{t_0}^{t_f} \boldsymbol{\lambda}^T\left(t\right) \frac{\partial L_i}{\partial \boldsymbol{\theta}}\left(t, \boldsymbol{\xi}\left(t\right); \boldsymbol{\theta}\right)\, dt + \boldsymbol{\lambda}^T\left(t_f\right) \frac{\partial \phi_i}{\partial \boldsymbol{\theta}}\left(t, \boldsymbol{\xi}\left(t\right); \boldsymbol{\theta}\right)$$

This sweeping gradient method was recently extended to trajectories modeled by ODEs with events by inducing an associated adjoint event with update equation

$$\begin{aligned}
\boldsymbol{\lambda}\left(t_e^-\right) = &\left[\frac{\partial}{\partial \boldsymbol{\xi}} \mathbf{h}\left(t_e, \boldsymbol{\xi}\left(t_e^+\right); \boldsymbol{\theta}\right)\right]^T \boldsymbol{\lambda}\left(t_e^+\right) \\
&+ \left[\frac{dt_e}{d\boldsymbol{\xi}}\right]^T \left[\frac{\partial}{\partial \boldsymbol{\xi}} \mathbf{h}\left(t_e, \boldsymbol{\xi}\left(t_e^+\right); \boldsymbol{\theta}\right)\, \mathbf{f}\left(t_e, \boldsymbol{\xi}\left(t_e^-\right); \boldsymbol{\theta}\right) - \mathbf{f}\left(t_e, \boldsymbol{\xi}\left(t_e^+\right); \boldsymbol{\theta}\right)\right]^T \boldsymbol{\lambda}\left(t_e^+\right)
\end{aligned} \tag{8}$$

where

$$\frac{dt_e}{d\boldsymbol{\xi}} = \left(\frac{\partial}{\partial \boldsymbol{\xi}} \gamma\left(t_e, \boldsymbol{\xi}\left(t_e^-\right); \boldsymbol{\theta}\right) \mathbf{f}\left(t_e, \boldsymbol{\xi}\left(t_e^-\right); \boldsymbol{\theta}\right)\right)^{-1} \frac{\partial}{\partial \xi} \gamma\left(t_e, \boldsymbol{\xi}\left(t_e^-\right); \boldsymbol{\theta}\right)$$

represents the sensitivity of the event time with respect to the state. The events in the state and adjoint trajectories contribute additional discontinuous terms to the gradient of the form

$$\begin{aligned}\Delta\frac{\partial J_i}{\partial \boldsymbol{\theta}} = &\boldsymbol{\lambda}^T\left(t_e^+\right)\left[\frac{\partial}{\partial \boldsymbol{\theta}}\mathbf{h}\left(t_e, \boldsymbol{\xi}\left(t_e^-\right);\boldsymbol{\theta}\right)\right] \\ &+\boldsymbol{\lambda}^T\left(t_e^+\right)\left[\frac{\partial}{\partial \boldsymbol{\xi}}\mathbf{h}\left(t_e, \boldsymbol{\xi}\left(t_e^-\right);\boldsymbol{\theta}\right)\, f\left(t_e, \boldsymbol{\xi}\left(t_e^-\right);\boldsymbol{\theta}\right) - \mathbf{f}\left(t_e, \boldsymbol{\xi}\left(t_e^+\right);\boldsymbol{\theta}\right)\right]\frac{dt_e}{d\boldsymbol{\theta}}\end{aligned} \tag{9}$$

where

$$\frac{dt_e}{d\boldsymbol{\theta}} = -\left(\frac{\partial}{\partial t_e}\gamma\left(t_e, \boldsymbol{\xi}\left(t_e^-\right);\boldsymbol{\theta}\right) + \frac{\partial}{\partial \boldsymbol{\xi}}\gamma\left(t_e, \boldsymbol{\xi}\left(t_e^-\right);\boldsymbol{\theta}\right)\mathbf{f}\left(t_e, \boldsymbol{\xi}\left(t_e^-\right);\boldsymbol{\theta}\right)\right)^{-1}\frac{\partial}{\partial \boldsymbol{\theta}}\gamma\left(t_e, \boldsymbol{\xi}\left(t_e^-\right);\boldsymbol{\theta}\right)$$

represents the sensitivity of the event time with respect to the parameters. For additional details on the sensitivity calculation, we refer the interested reader to [1].

To apply SGM to trajectory and system design, it is useful to model a trajectory analysis as a black-box mapping $\mathbf{T}\colon \mathbb{R}^{n_\theta} \mapsto \mathbb{R}^{n_J}$, where n_θ is the dimension of a parameter space and n_J is the dimension of the *trajectory analysis output* space. Then we can construct optimization problems of the form

$$\begin{aligned}\underset{\boldsymbol{\theta}}{\text{minimize}} \quad & \Gamma_0\left(\boldsymbol{J}, \boldsymbol{\theta}\right) \\ \text{subject to} \quad & \Gamma_i\left(\boldsymbol{J}, \boldsymbol{\theta}\right) \le 0, \qquad i = 1, \ldots, m \\ \text{where} \quad & \boldsymbol{J} = \mathbf{T}\left(\boldsymbol{\theta}\right)\end{aligned} \tag{10}$$

where Γ_0 is an objective function and the Γ_i are constraint functions. Since SGM provides an efficient method for computing the gradients of $\boldsymbol{J}$ with respect to the parameters $\boldsymbol{\theta}$, application of the chain rule makes it possible to compute gradients of the objective and constraint functions with respect to the trajectory model parameters. The availability of these gradients permits the use of efficient, off-the-shelf gradient-based solvers to optimal control or system design problems of the form (10). By combining SGM and LinCov, these gradient-based optimizers can be used to solve robust optimal trajectory and navigation system design problems with minimal software development effort. For additional details on model formulation with applications to robust optimal orbital rendezvous trajectory design, we refer the interested reader to our companion paper [17].

III. NRHO Rendezvous and Docking Concept of Operations

The Near-Rectilinear Halo Orbit (NRHO) rendezvous, proximity operations, and docking (RPOD) scenario utilized to demonstrate the co-optimization of the relative trajectory profile and navigation requirements adopts an anticipated NASA Artemis VI and onward missions where a lander on the lunar surface returns to the NRHO to rendezvous with the Gateway vehicle as depicted in 3. This section will provided an overview of the baseline rendezvous trajectory design, the models for the guidance, navigation, and control system, and the assumed analysis input parameters.

A. System Dynamics Model

For the purpose of optimization in the context of the RPOD scenario, we will use a simpler, lower fidelity dynamics model to reduce computation time. Luquette linearizes relative motion dynamics for the restricted three body problem (RTBP) about the target spacecraft in his doctoral dissertation [18]. Assuming the rotation of the Sun-referenced local vertical, local horizontal (LVLH) coordinate frame is negligible, this is expressed as the system of differential equations

$$\dot{\bar{\mathbf{x}}} = \begin{bmatrix}\mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} \\ \Xi\left(t\right) & \mathbf{0}_{3\times3}\end{bmatrix}\bar{\mathbf{x}} = \mathbf{A}_{LR}\bar{\mathbf{x}} \tag{11}$$

where

$$\Xi(\mathrm{t}) = -\left(\frac{\mu_1}{|\mathbf{r}_{1L}|^3} + \frac{\mu_2}{|\mathbf{r}_{2L}|^3}\right)\mathbf{I}_3 + \frac{3\mu_1}{|\mathbf{r}_{1L}|^3}\left[\mathbf{e}_{1L}\cdot\mathbf{e}_{1L}^{\mathrm{T}}\right] + \frac{3\mu_2}{|\mathbf{r}_{2L}|^3}\left[\mathbf{e}_{2L}\cdot\mathbf{e}_{2L}^{\mathrm{T}}\right] \tag{12}$$

Figure 4 shows the vectors referenced in Equation 12: $\mathbf{r}_{1L}$ and $\mathbf{r}_{2L}$ refer to the vectors from the two central masses to the leader (target) satellite, whereas $\mathbf{e}_{1L}$ and $\mathbf{e}_{2L}$ refer to the normalized form of these vectors. μ_1 and μ_2 represent the gravitational parameters of the two central masses. In this paper, the two central masses are the Earth and the Moon.

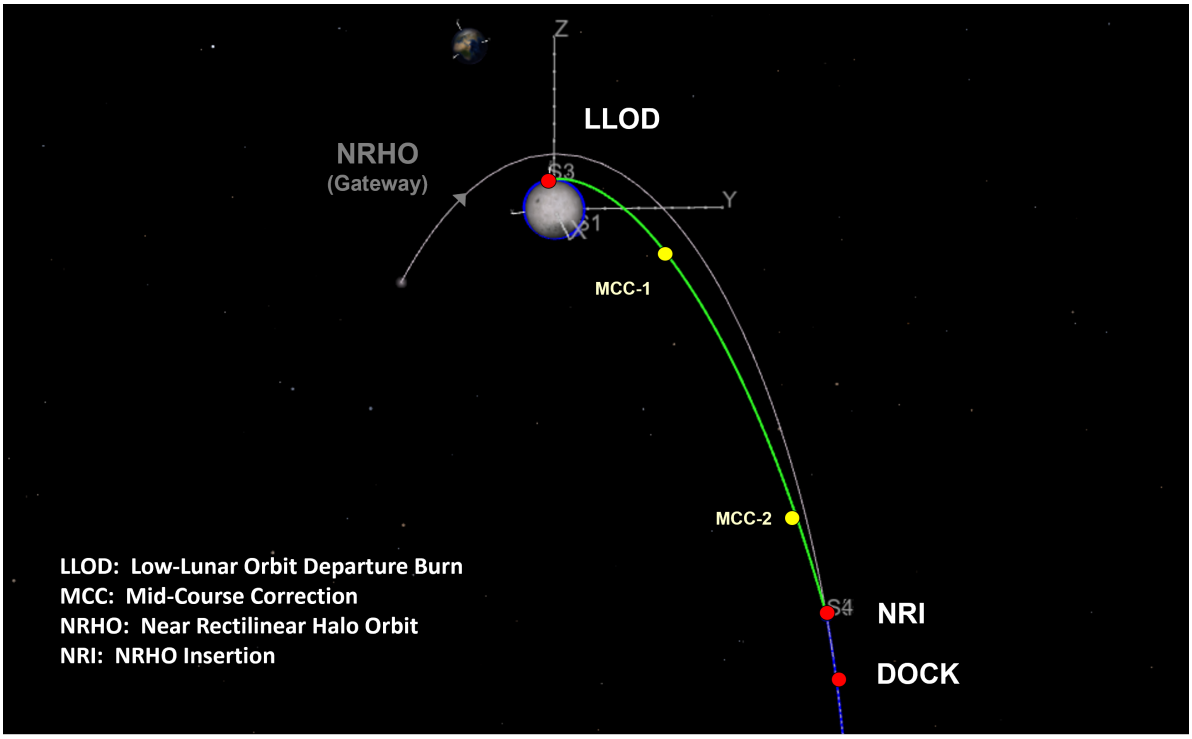


Fig. 3 Low Lunar Orbit to NRHO Rendezvous and Docking Profile

Assignment of the "1" and "2" subscripts to these bodies is arbitrary. During the phase of flight we are interested in, we assume that $|\Xi(t)| \ll 1$ so the dynamics are well-approximated as a double-integrator model in the three translational relative degrees of freedom.

B. Targeting Algorithm

The relative targeting utilizes the linearized RTBP and targets the relative position $\mathbf{r}_{i+1}$ at some specified maneuver end time t_{i+1}. To demonstrate how this is done, the system's state transition matrix Φ is required, which is a function of the state dynamics $\mathbf{A}_{LR}$ defined in equation 11 and the transfer time, $\Delta t_i = t_{i+1} - t_i$, which is the difference between the time of the burn t_i and the maneuver end time t_{i+1}.

$$\Phi = e^{\mathbf{A}_{LR}\Delta \mathrm{t}} \tag{13}$$

The state transition matrix can be divided in terms of the relative position and velocity states,

$$\Phi = \begin{bmatrix} \Phi_{rr} & \Phi_{rv} \\ \Phi_{vr} & \Phi_{vv} \end{bmatrix} \tag{14}$$

Using Φ to determine how an instantaneous change in velocity $\Delta\mathbf{v}$ will influence the relative position at time t_{i+1},

$$\begin{bmatrix} \mathbf{r}_{i+1} \\ \mathbf{v}_{i+1} \end{bmatrix} = \begin{bmatrix} \Phi_{rr} & \Phi_{rv} \\ \Phi_{vr} & \Phi_{vv} \end{bmatrix} \begin{bmatrix} \mathbf{r}_i \\ \mathbf{v}_i + \Delta\mathbf{v} \end{bmatrix} \tag{15}$$

the three rows associated with the position can be expressed as

$$\mathbf{r}_{i+1} = \Phi_{rr}\mathbf{r}_i + \Phi_{rv}\left(\mathbf{v}_i + \Delta\mathbf{v}\right) \tag{16}$$

Targeting a final relative position state $\mathbf{r}_{i+1}$ from the current state requires an instantaneous velocity change of

$$\Delta\mathbf{v} = \Phi_{rv}^{-1}\left(\mathbf{r}_{i+1} - \Phi_{rr}\mathbf{r}_i\right) - \mathbf{v}_i \tag{17}$$

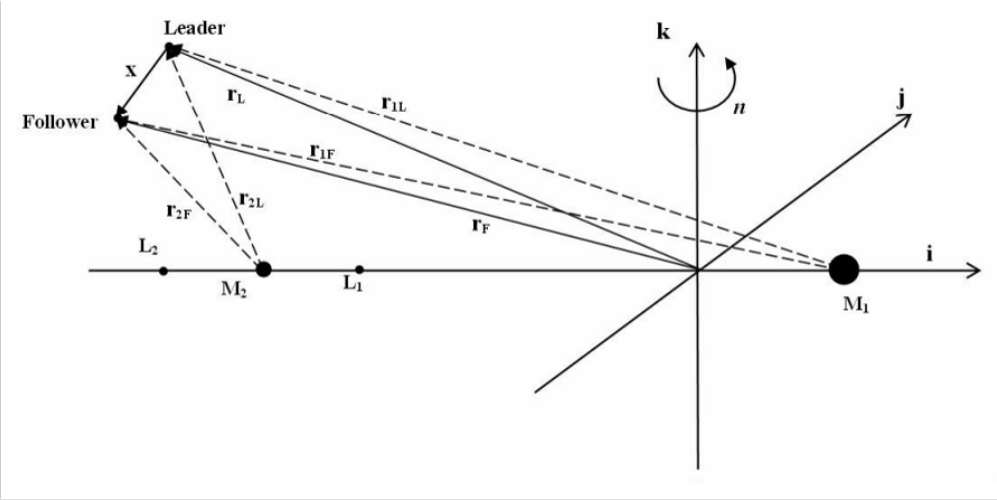


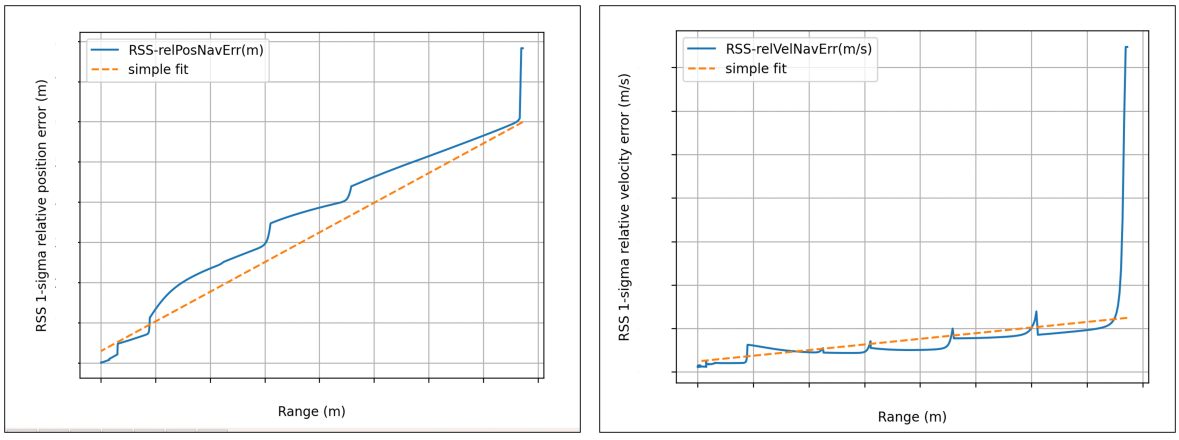
Fig. 4 Diagram of vectors in the RTBP

which is treated as the nominal control model $\Delta\bar{\mathbf{g}}\left(\bar{\mathbf{x}}_j^{-c}, t_j\right)$ for this scenario. This controller has a corresponding mapping from the control $\Delta\bar{\mathbf{u}}$ to the change in the nominal state by

$$\bar{\mathbf{x}}\left(t_j^{+c}\right) = \bar{\mathbf{x}}\left(t_j^{-c}\right) + \mathbf{d}\left(\bar{\mathbf{x}}_j^{-c}, \Delta\bar{\mathbf{u}}_j, t_j\right) = \bar{\mathbf{x}}\left(t_j^{-c}\right) + \left[\mathbf{0}_{3\times3}, \mathbf{I}_{3\times3}\right]^T \Delta\bar{\mathbf{u}} \tag{18}$$

C. Navigation System Model

The navigation system is modeled as a linear function of the relative range $\mathbf{N}\left(|\boldsymbol{\rho}|;\theta_N\right)$, implemented using a stochastic navigation approach[4]. Prior to each burn, the navigation error impact on the covariance is applied based on the nominal relative range of the burn location. A baseline navigation system is modeled by fitting the navigation error from a LinCov simulation of a spacecraft equipped with a notional sensor suite flying the baseline trajectory described in the next subsection. The root sum of squares (RSS) navigation error from the simulation and the linear fit is shown in Figure 5. Despite the simplicity of the model, a sampling of the results presented here were verified to sufficient accuracy against a full nonlinear simulation with notional sensor suite models. Navigation system design (or navigation system requirement specification) will be performed rotating the line upwards as parameterized by θ_N. It is assumed that the ratio of the position and velocity navigation error models is constant.



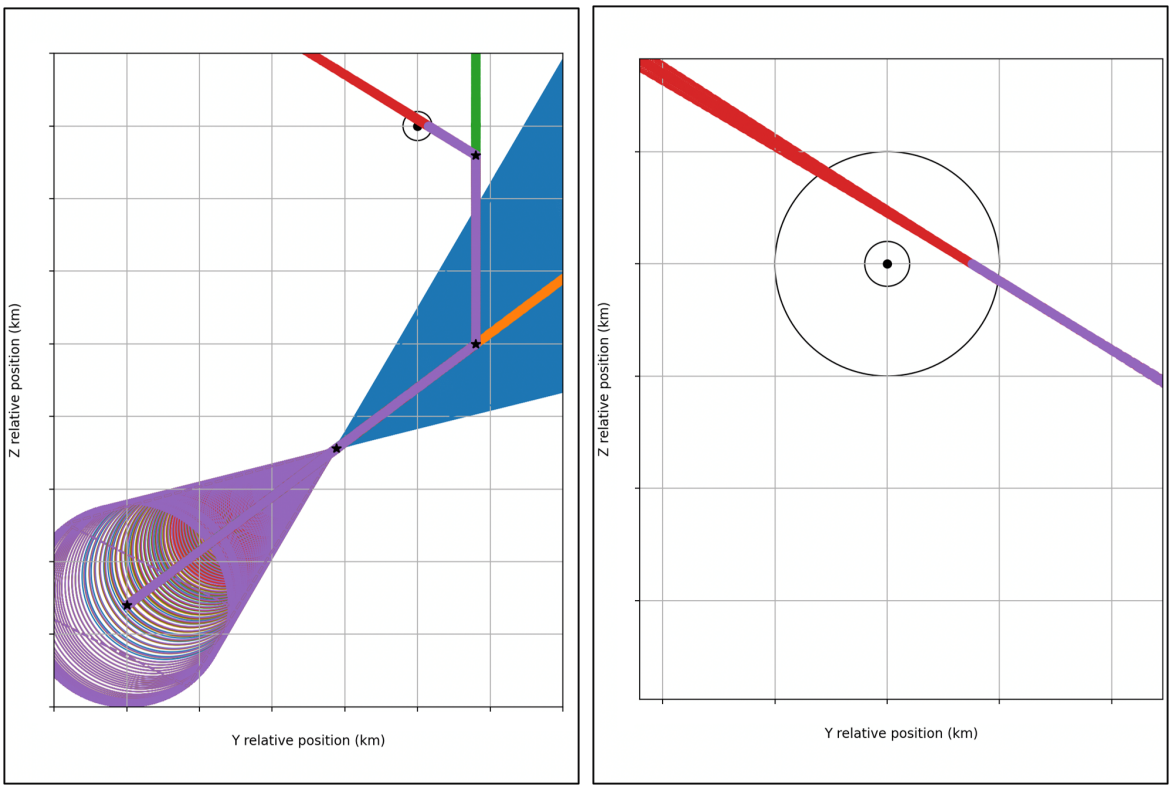
(a) RSS Relative Position Navigation Error

(b) RSS Relative Velocity Navigation Error

Fig. 5 Relative navigation error model as a function of range

D. Baseline Trajectory Design

A baseline trajectory will be used as a comparison for the optimized trajectories and navigation systems to assess the impacts on performance criteria like propellant usage and safety constraints. We create an arbitrary trajectory in-family with those presented in [19], by creating a baseline maneuver plan $\boldsymbol{\theta}_M = \left[\mathbf{r}_1^T, t_2, \mathbf{r}_2^T, \ldots, t_{n_M-1}, \mathbf{r}_{n_M-1}^T, t_{n_M}\right]^T$, which specifies the execution times, durations, and target positions (and implicitly specifies the number of maneuvers, n_M) for the linear targeting algorithm, dynamics, and navigation models described above. The trajectory in the local vertical, local horizontal (LVLH) frame with anticipated trajectory dispersions for both the far-range and close-range rendezvous is highlighted in Figure 6. Figure 6 also shows the free drift trajectory corresponding to each burn in its own color which will be used to assess certain safety requirements.



(a) Far-Range (b) Close-Range

Fig. 6 Notional relative NRHO RPOD trajectory profile with anticipated trajectory dispersions

Trajectory design is performed by modifying the the maneuver plan. For this work, a few key features of the baseline trajectory will be preserved for all trajectory designs. The first burn will represent the insertion to the NRHO relative to the target, and will be referred to as the NRI burn. The second to last burn will be the docking axis acquisition (DAA) burn, and the final burn will be a docking axis hold (DAH) burn to maintain the position on the docking axis prior to the final docking sequence. We also assume that the end-time for each maneuver is the execution time for the next maneuver.

E. Complete state equations

To simulate the nominal dynamics and the associated uncertainty, we form the complete state

$$\boldsymbol{\xi}(t) = \begin{bmatrix} \bar{\mathbf{x}}(t) \\ \operatorname{vec}\mathbf{C}(t) \\ |\Delta\bar{v}|(t) \\ \sigma_{|\Delta v|}(t) \\ \underline{\sigma}_{|\boldsymbol{\rho}|}(t) \\ \boldsymbol{\mathcal{L}}(t) \end{bmatrix}$$

where $\bar{\mathbf{x}}$ is the nominal state, the vec operator un-ravels the argument matrix into a column vector, optionally ignoring repeated elements of a symmetric matrix for computational and storage requirement reduction, $\mathbf{C}$ is the augmented covariance matrix for the system model, and the remaining state components are scalar accumulators used to model the trajectory analysis outputs used in the objectives and constraints of the optimization problem formulation. The onboard navigation error covariance does not need to be modeled since we are using a stochastic navigation error approach. The dynamic rate, measurement update, and correction update for each component of the complete state can be treated independently.

The nominal state follows the dynamics and correction updates defined by (11) and (18), respectively. The stochastic navigation system model does not impact the nominal state.

The augmented covariance has continuous dynamics and control updates given by

$$\begin{aligned} \dot{\mathbf{C}} &= \boldsymbol{\mathcal{F}}\mathbf{C} + \mathbf{C}\boldsymbol{\mathcal{F}}^T \\ \mathbf{C}\left(t_j^0\right) &= \boldsymbol{\mathcal{A}}\mathbf{C}\left(t_k^-\right)\boldsymbol{\mathcal{A}}^T - \boldsymbol{\mathcal{B}}\operatorname{diag}\left(\mathbf{N}\left(|\boldsymbol{\rho}|;\theta_N\right)\right)\boldsymbol{\mathcal{B}}^T \\ \mathbf{C}\left(t_j^{+c}\right) &= \boldsymbol{\mathcal{D}}_j\mathbf{C}^{-c}\left(t_j^0\right)\boldsymbol{\mathcal{D}}_j^T + \boldsymbol{\mathcal{N}}_j\mathbf{S}_{\Delta w}\left(t_j\right)\boldsymbol{\mathcal{N}}_j^T \end{aligned}$$

where $\mathbf{C}\left(t_k^-\right)$ is the value prior to the impulsive maneuver, $\mathbf{C}\left(t_j^0\right)$ is an intermediate value after the stochastic navigation update has been applied, $\mathbf{N}\left(|\boldsymbol{\rho}|;\theta_N\right)$ is the linear navigation error model, and $\mathbf{C}\left(t_j^{+c}\right)$ is after the maneuver has been executed, and $S_{\Delta w}\left(t_j\right)$ is the maneuver execution covariance. The coefficients for these equations are given by

$$\boldsymbol{\mathcal{F}} = \begin{bmatrix} \mathbf{A}_{LR} & 0 \\ 0 & \mathbf{A}_{LR} \end{bmatrix}$$

$$\boldsymbol{\mathcal{A}} = \begin{bmatrix} \mathbf{I}_{6\times6} & \mathbf{0}_{6\times6} \\ \mathbf{I}_{6\times6} & \mathbf{0}_{6\times6} \end{bmatrix} \qquad \boldsymbol{\mathcal{B}} = \begin{bmatrix} \mathbf{0}_{6\times6} \\ -\mathbf{I}_{6\times6} \end{bmatrix}$$

$$\boldsymbol{\mathcal{D}}_j = \begin{bmatrix} 0 & \mathbf{D}_{\Delta\hat{u}}\left(t_j\right)\Delta\hat{\mathbf{G}}_{\hat{x}}\left(t_j\right) \\ 0 & \mathbf{I} + \hat{\mathbf{D}}_{\Delta\hat{u}}\left(t_j\right)\Delta\hat{\mathbf{G}}_{\hat{x}}\left(t_j\right) \end{bmatrix} \quad \mathbf{D}_{\Delta\hat{u}}\left(t_j\right)\Delta\hat{\mathbf{G}}_{\hat{x}}\left(t_j\right) = \begin{bmatrix} \mathbf{0}_{3\times3} \\ \mathbf{I}_{3\times3} \end{bmatrix}\left[-\Phi_{rv}^{-1}\Phi_{rr}, -I_{3\times3}\right] \quad \boldsymbol{\mathcal{N}}_j = \begin{bmatrix} \mathbf{I}_{6\times6} \\ \mathbf{0}_{6\times6} \end{bmatrix}$$

The dynamics for each of the accumulator states are null. The nominal impulse accumulator state $|\Delta\bar{v}|$ and the RSS impulse dispersion state $\sigma_{|\Delta v|}(t)$ have updates during a correction maneuver given by

$$|\Delta\bar{v}|\left(t_j^+\right) = |\Delta\bar{v}|\left(t_j^-\right) + \Delta v, \tag{19}$$

where the second term is computed according to (17), and

$$\sigma_{|\Delta v|}\left(t_j^+\right) = \sigma_{|\Delta v|}\left(t_j^-\right) + \sqrt{\operatorname{tr}\left(\mathbf{M}_c\mathbf{C}\left(t_j^0\right)\mathbf{M}_c^T\right)},$$

where

$$\mathbf{M}_c = \left[\hat{\mathbf{D}}_{\Delta\hat{u}}\left(t_j\right)\Delta\hat{\mathbf{G}}_{\hat{x}}\left(t_j\right)\right]\left[\mathbf{0}_{6\times6}, \mathbf{I}_{6\times6}\right].$$

The dispersed point of closest approach (DPCA) accumulator state $\underline{\sigma}_{|\boldsymbol{\rho}|}(t)$ and relative angular momentum at DPCA accumulator state $\boldsymbol{\mathcal{L}}$ are used for the free drift and under burn constraints. The DPCA is the minimum value of the 3σ

dispersed closest distance (DCD), denoted by $\sigma_{|\boldsymbol{\rho}|}$, over the trajectory. The DCD can be computed as a function of the complete state,

$$\sigma_{|\boldsymbol{\rho}|}\left(\boldsymbol{\xi}\left(t\right)\right) = \left|\boldsymbol{\rho}\left(t\right)\right| - 3\sqrt{\hat{\boldsymbol{\rho}}^{T}\left(t\right)\mathbf{C}\left(t\right)\hat{\boldsymbol{\rho}}\left(t\right)}$$

where $\hat{\boldsymbol{\rho}}\left(t\right) = \left[\mathbf{0}_{3\times1}^{T}, \boldsymbol{\rho}^{T}(t)/|\boldsymbol{\rho}(t)|, \mathbf{0}_{6\times1}^{T}\right]^{T}$ captures the position dispersion along the vector from the chaser to target. Then we define a new event that occurs when the relative velocity of the DCD changes sign,

$$\gamma_{\mathrm{DCPA}}\left(t, \boldsymbol{\xi}\left(t\right); \boldsymbol{\theta}\right) = \frac{\partial \sigma_{|\boldsymbol{\rho}|}\left(\boldsymbol{\xi}\left(t\right)\right)}{\partial \boldsymbol{\xi}}\mathbf{f}\left(t, \boldsymbol{\xi}\left(t\right); \boldsymbol{\theta}\right)$$

which has a corresponding update

$$\underline{\sigma}_{|\boldsymbol{\rho}|}\left(t^{+}_{\mathrm{DCPA}}\right) = \begin{cases} \sigma_{|\boldsymbol{\rho}|}\left(\boldsymbol{\xi}\left(t^{-}_{\mathrm{DCPA}}\right)\right) & \underline{\sigma}_{|\boldsymbol{\rho}|}\left(t^{-}_{\mathrm{DCPA}}\right) > \sigma_{|\boldsymbol{\rho}|}\left(\boldsymbol{\xi}\left(t^{-}_{\mathrm{DCPA}}\right)\right) \\ \underline{\sigma}_{|\boldsymbol{\rho}|}\left(t^{-}_{\mathrm{DCPA}}\right) & \underline{\sigma}_{|\boldsymbol{\rho}|}\left(t^{-}_{\mathrm{DCPA}}\right) \leq \sigma_{|\boldsymbol{\rho}|}\left(\boldsymbol{\xi}\left(t^{-}_{\mathrm{DCPA}}\right)\right) \end{cases}$$

Similarly, the relative angular momentum at DPCA accumulator $\boldsymbol{\mathcal{L}}$ has an update

$$\boldsymbol{\mathcal{L}}\left(t^{+}_{\mathrm{DCPA}}\right) = \begin{cases} \boldsymbol{\rho}\times\dot{\boldsymbol{\rho}} & \underline{\sigma}_{|\boldsymbol{\rho}|}\left(t^{-}_{\mathrm{DCPA}}\right) > \sigma_{|\boldsymbol{\rho}|}\left(\boldsymbol{\xi}\left(t^{-}_{\mathrm{DCPA}}\right)\right) \\ 0 & \underline{\sigma}_{|\boldsymbol{\rho}|}\left(t^{-}_{\mathrm{DCPA}}\right) \leq \sigma_{|\boldsymbol{\rho}|}\left(\boldsymbol{\xi}\left(t^{-}_{\mathrm{DCPA}}\right)\right) \end{cases}$$

to capture which "side" of the target the free-drift is passing.

F. Notional Mission Constraints

For both navigation requirement specification and trajectory design, we will consider several notional mission constraints that must be satisfied. First, we can consider several notional timing requirements. We may consider a minimum time $\underline{t}_2$ between the NRI burn and the next burn which is longer than the minimum time $\underline{\Delta t}$ between subsequent burns. The minimum time between burns generally reflects operational requirements such as navigation filter convergence, target acquisition, and communication considerations. The increased minimum time after NRI represents the impact of the larger maneuver for insertion. The maximum time $\bar{t}_f$ for this sequence of maneuvers is also specified so it can fit within the larger mission plan.

We will also consider approach sphere (AS) and under burn requirements. Both of these constraints require that we perform a trajectory analysis for a free drift corresponding to each burn. The free drift trajectories model the behavior if the thrusters fail after the corresponding burn. The AS requirement that the DPCA is greater than the AS radius R_{AS} reduces the chance of collision after thruster failure to an acceptable level. The free-drift trajectory for the DAA burn is allowed to enter the AS but must respect a keep-out sphere (KOS) with radius R_{KOS} reflecting the intermediate stage of the mission between approach and docking. The under burn constraint requires that the relative angular velocity at the DPCA for each free drift has the same sign. This ensures that the chaser will drift into free space if an under burn occurs.

IV. Application of Sweeping Gradient Method with Linear Covariance Analysis

In this section, we describe the process of applying SGM and LinCov to design the navigation system and trajectory. We first mirror the heritage process of designing either the navigation system or the trajectory, independently. As we shall see, this approach leaves potential margin that can be exploited by the system design. To take advantage of this, we perform a co-optimization where both the trajectory and navigation system are optimized together. Finally we fully explore this space by sweeping out a Pareto front over the navigation system error and propellant requirements.

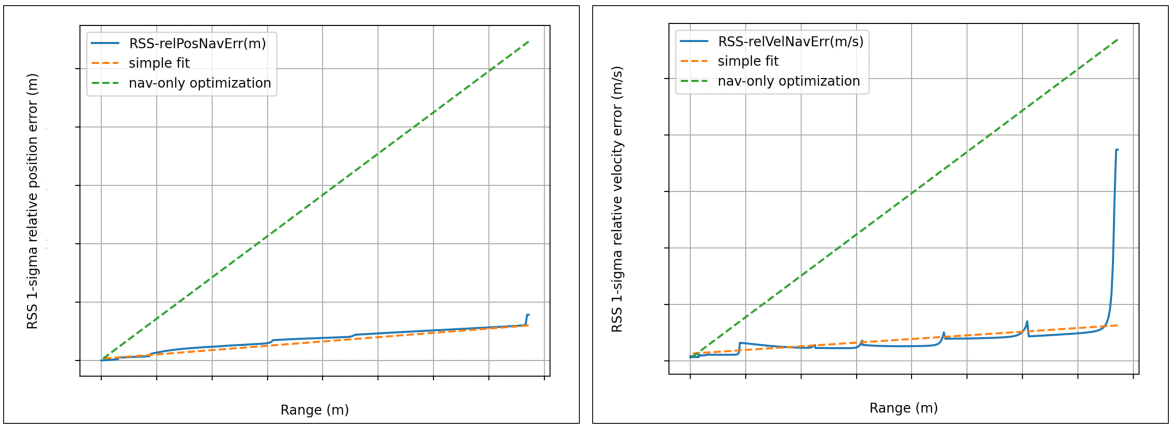
A. Navigation requirements optimized (maximized) for a fixed notional trajectory

First we consider the design stage where the trajectory is assumed to be fixed and the maximum allowable navigation errors that still meet core trajectory and vehicle constraints are identified. The basic question of this design stage is this: How large can the navigation errors be and still satisfy the free-drift and other mission constraints? To answer this

question numerically, we formulate the optimization problem

$$
\begin{aligned}
\underset{\theta_N}{\text{minimize}} \quad & -\theta_N \\
\text{subject to} \quad & \bar{\sigma}_{|\boldsymbol{\rho}|,i}\left(t_f\right) \geq R_{\text{AS}}, \qquad i = 1, \ldots, n_M - 2 \\
& \bar{\sigma}_{|\boldsymbol{\rho}|,n_M-1}\left(t_f\right) \geq R_{\text{KOS}} \\
\text{where} \quad & T_i : \begin{bmatrix} \boldsymbol{\theta}_M \\ \theta_N \end{bmatrix} \mapsto \begin{bmatrix} \underline{\sigma}_{|\boldsymbol{\rho}|,i}\left(t_f\right) \\ \boldsymbol{\mathcal{L}}_i\left(t_f\right) \end{bmatrix}, \qquad i = 1, \ldots, n_M - 1 \\
& T_{n_M} : \begin{bmatrix} \boldsymbol{\theta}_M \\ \theta_N \end{bmatrix} \mapsto \begin{bmatrix} |\Delta\bar{v}|\left(t_f\right) \\ \sigma_{|\Delta v|}\left(t_f\right) \end{bmatrix}
\end{aligned}
\tag{20}
$$

where θ_N, the navigation system model parameter, is maximized while still satisfying the free drift constraint, the only constraint that will be impacted by this design problem. The resulting navigation system performance is shown in Figure 7 and the resulting trajectory dispersions for the notional relative trajectory with free drifts is shown in Figure 8. The optimizer is able to increase the navigation error parameter θ_N until the DAA free-drift is tangent with the KOS.

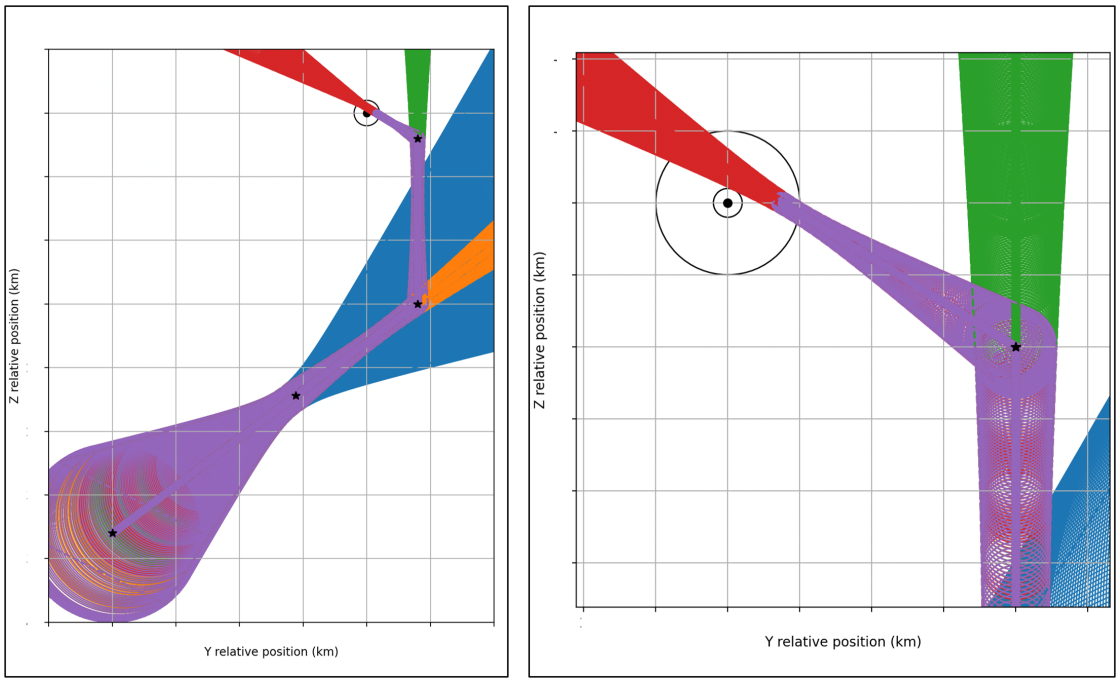


(a) RSS Position Error (b) RSS velocity Error

Fig. 7 Optimized navigation requirements as a function of range for a fixed notional trajectory

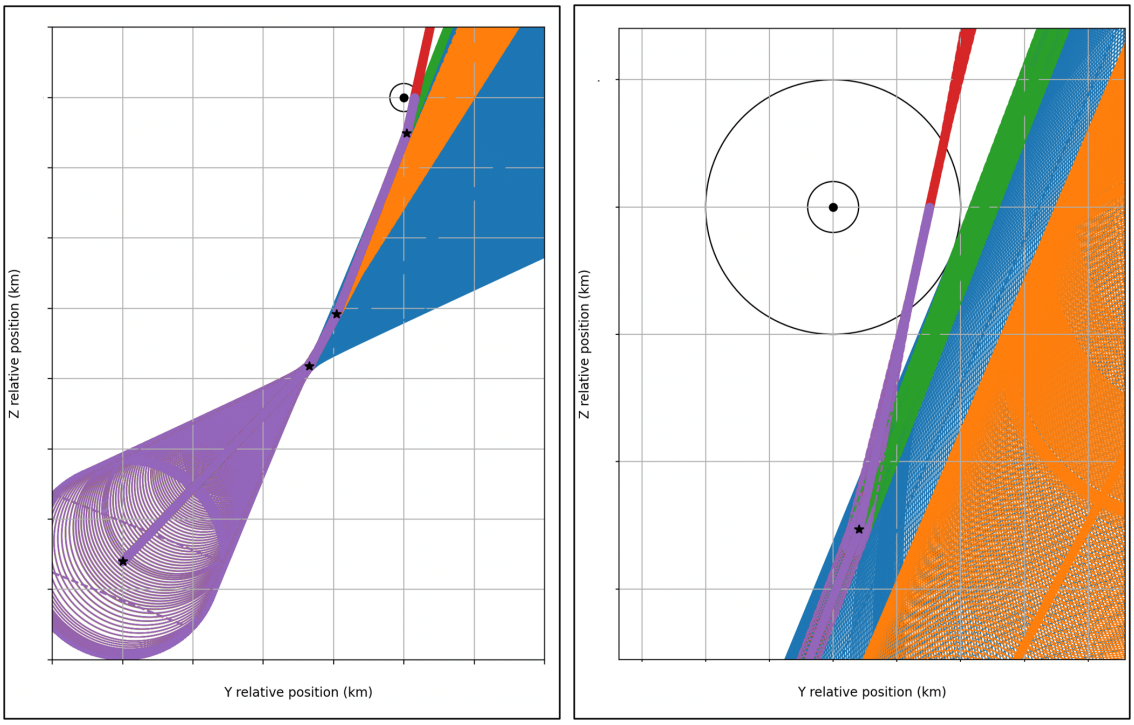
B. Trajectory profile optimized for fixed relative navigation requirements

Next, we consider the design stage where the notional navigation errors are fixed and the trajectory design can be altered to reduce the total Δv requirement which is the sum of the nominal and 3σ dispersion of the impulse magnitudes.



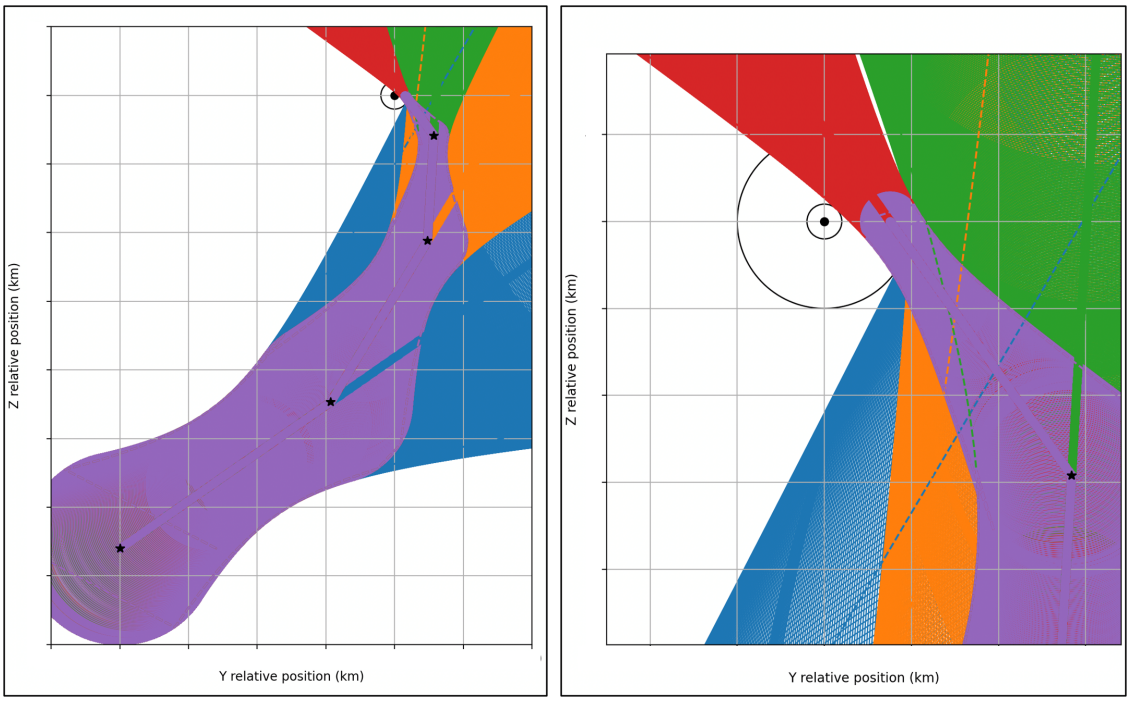
(a) Far-Range (b) Close-Range

Fig. 8 Trajectory dispersions for an optimized navigation system given a fixed notional trajectory



(a) Far-Range (b) Close-Range

Fig. 9 Trajectory optimization for notional navigation system



(a) Far-Range (b) Close-Range

Fig. 10 Trajectory profile and dispersions for a co-optimized trajectory design and navigation system.

To solve this problem numerically, we solve the optimization problem

$$
\begin{aligned}
\underset{\boldsymbol{\theta}_M}{\text{minimize}} \quad & |\Delta\bar{v}|\left(t_f\right) + 3\sigma_{|\Delta v|}\left(t_f\right) \\
\text{subject to} \quad & \bar{\sigma}_{|\boldsymbol{\rho}|,i}\left(t_f\right) \geq R_{\text{AS}}, \qquad i = 1, \ldots, n_M - 2 \\
& \bar{\sigma}_{|\boldsymbol{\rho}|,n_M-1}\left(t_f\right) \geq R_{\text{KOS}} \\
& \operatorname{sgn} \boldsymbol{\mathcal{L}}_{i-1}\left(t_f\right) = \operatorname{sgn} \boldsymbol{\mathcal{L}}_i\left(t_f\right), \qquad i = 2, \ldots, n_M - 1 \\
& \underline{t}_2 \leq t_2 \\
& \underline{\Delta t} \leq t_i - t_{i-1}, \qquad i = 3, \ldots, n_M \\
& t_{n_M} \leq \bar{t}_f \\
\text{where} \quad & T_i : \begin{bmatrix} \boldsymbol{\theta}_M \\ \theta_N \end{bmatrix} \mapsto \begin{bmatrix} \underline{\sigma}_{|\boldsymbol{\rho}|,i}\left(t_f\right) \\ \boldsymbol{\mathcal{L}}_i\left(t_f\right) \end{bmatrix}, \qquad i = 1, \ldots, n_M - 1 \\
& T_{n_b} : \begin{bmatrix} \boldsymbol{\theta}_M \\ \theta_N \end{bmatrix} \mapsto \begin{bmatrix} |\Delta\bar{v}|\left(t_f\right) \\ \sigma_{|\Delta v|}\left(t_f\right) \end{bmatrix}
\end{aligned}
\tag{21}
$$

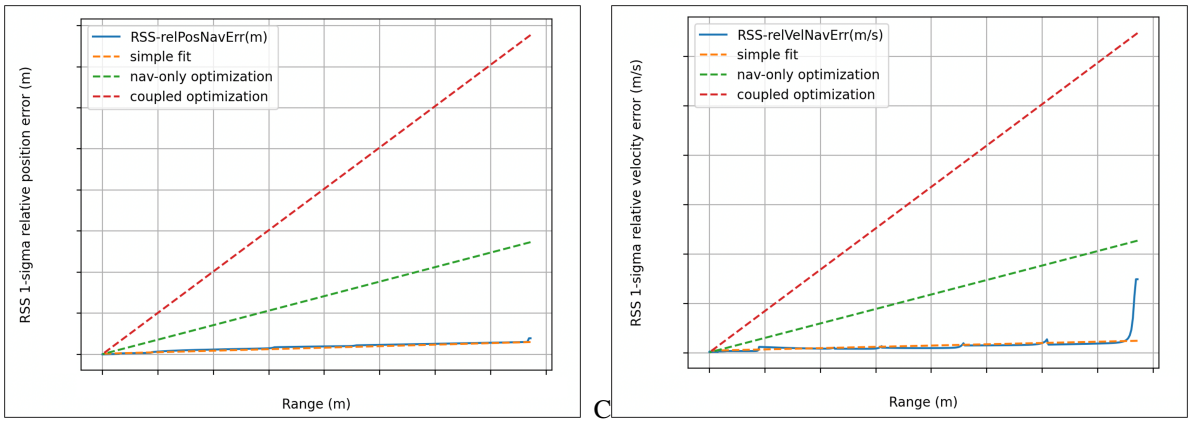
We find that the total Δv requirement can be reduced by half by modifying the trajectory profile to account for the anticipated navigation errors. The resulting optimized trajectory given the notional, yet fixed, navigation requirements is shown in Figure 9. This optimized trajectory has the first $n_M - 2$ free-drift dispersion tubes tangent to the KOS, which suggests more performance is being fully exploited for these maneuvers compared to the navigation-only optimization. However, we note that the optimizer does not have enough degrees of freedom to also make the DAA free drift dispersion tube tangent to the AS, suggesting more performance can be achieved.

C. Co-optimization of relative trajectory profile and navigation requirements

In this section, we consider the possibility of combining the trajectory design with navigation system requirement specification. To do so, we consider the optimization problem

$$
\begin{aligned}
\underset{\boldsymbol{\theta}_M, \theta_N}{\text{minimize}} \quad & |\Delta\bar{v}|\left(t_f\right) + 3\sigma_{|\Delta v|}\left(t_f\right) - w_N\theta_N \\
\text{subject to} \quad & \bar{\sigma}_{|\boldsymbol{\rho}|,i}\left(t_f\right) \geq R_{\text{AS}}, \qquad i = 1, \ldots, n_M - 2 \\
& \bar{\sigma}_{|\boldsymbol{\rho}|,n_M-1}\left(t_f\right) \geq R_{\text{KOS}} \\
& \operatorname{sgn}\boldsymbol{\mathcal{L}}_{i-1}\left(t_f\right) = \operatorname{sgn}\boldsymbol{\mathcal{L}}_i\left(t_f\right), \qquad i = 2, \ldots, n_M - 1 \\
& \underline{t_2} \leq t_2 \\
& \underline{\Delta t} \leq t_i - t_{i-1}, \qquad i = 3, \ldots, n_M \\
& t_{n_M} \leq \bar{t}_f \\
\text{where} \quad & T_i : \begin{bmatrix} \boldsymbol{\theta}_M \\ \theta_N \end{bmatrix} \mapsto \begin{bmatrix} \underline{\sigma}_{|\boldsymbol{\rho}|,i}\left(t_f\right) \\ \boldsymbol{\mathcal{L}}_i\left(t_f\right) \end{bmatrix}, \qquad i = 1, \ldots, n_M - 1 \\
& T_{n_b} : \begin{bmatrix} \boldsymbol{\theta}_M \\ \theta_N \end{bmatrix} \mapsto \begin{bmatrix} |\Delta\bar{v}|\left(t_f\right) \\ \sigma_{|\Delta v|}\left(t_f\right) \end{bmatrix}
\end{aligned} \tag{22}
$$

where the objective of the previous two optimization problems is combined with a navigation weighting factor w_N. The resulting trajectory profile and free drifts with dispersions are show in Figure 10 while the resulting navigation error requirements for both relative position and velocity are provided in Figure 11. A key observation is that not only can the allowable navigation errors increase, requiring a lower quality navigation system performance, but the total Δv required to complete the rendezvous and proximity operations requires less propellant than the original design which is only marginally worse than trajectory-only optimization.



(a) RSS Position Error (b) RSS velocity Error

Fig. 11 Navigation requirements as a function of range for co-optimized trajectory

D. Pareto front

To understand the trade-off between the navigation system requirements and propellant requirements, we can sweep out a Pareto front. This can be done by solving (22) for a variety of values of w_N. However, in order to ensure an even spacing of solutions in the design space, we equivalently solve (21) for a variety of values for θ_N. We can also consider different numbers of maneuvers n_M to ensure the best possible trajectory design is identified for the given navigation system model. Finally, we can also consider different initial dispersions to add another degree of freedom in allocating the error budget.

Figure 12 shows the Pareto front of the total Δv requirement and navigation system model parameter θ_N for a variety of burn numbers and several initial dispersion scenarios. The solid, dotted, and dashed linestyle indicates the three

increasing initial dispersions modeled. The colors blue, orange, and green indicate the number of maneuvers in the trajectory for 2, 3, or 4. The black curves indicate the best performance over the number of burns for a given navigation error and initial dispersion. The stars in the upper left corner indicate the performance of baseline trajectories for each maneuver count. As the initial dispersions increase, the overall performance must get worse (increasing Δv requirement for the same navigation error). Interestingly, the number of burns with the best performance increases as the navigation error increases and the cross-over from the lower number of burns occurs at a lower navigation error as the initial dispersions increase. This suggests that more burns increase the robustness to error; since each burn uses targeting, each maneuver "cleans up" the error by a certain amount.

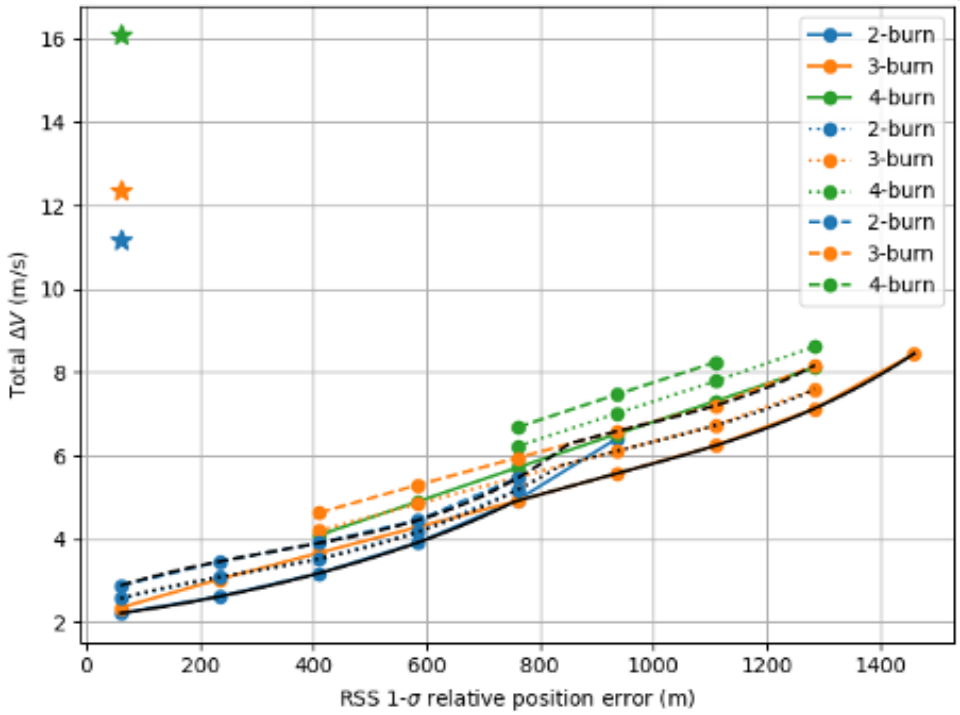


Fig. 12 Pareto front for navigation system requirement and trajectory design

V. Conclusion

In this work, we developed nominal and linear covariance analysis (LinCov) models for a rendezvous and docking mission in a Near-Rectilinear Halo Orbit (NRHO). Using a sweeping gradient method (SGM) for trajectory analysis and gradient-based optimizers, we performed navigation system requirement design, trajectory design, and co-optimization of the navigation system and trajectory by formulating the appropriate numerical optimization problem. We also used these models to generate a Pareto front of the design space, assessing the total Δv requirement as the navigation system performance, number of burns, and initial dispersions were varied. Using optimization as a dimensionality reduction tool as performed here allows engineers to allocate the error budget of the overall mission design by exploiting as many design degrees of freedom as possible.

References

[1] Margolis, B. W. L., "A Sweeping Gradient Method for Ordinary Differential Equations with Events," *Journal of Optimization Theory and Applications*, Vol. 199, No. 2, 2023, pp. 600–638. doi:10.1007/s10957-023-02303-3.

[2] Maybeck, P. S., *Stochastic models, estimation, and control*, Vol. 1, Academic Press, New York, 1979.

[3] Geller, D. K., "Linear Covariance Techniques for Orbital Rendezvous Analysis and Autonomous Onboard Mission Planning," *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 6, 2006, pp. 1404–1414.

[4] Woffinden, D., and Breger, L., "Automated Derivation and Verification of Navigation Requirements for On-orbit Rendezvous," *AIAA Guidance, Navigation, and Control Conference*, AIAA 2013-4964, Boston, MA, 2013.

[5] Woffinden, D., Bhatt, S., Kirkpatrick, D., and Spanos, P., "Optimal Multi-variable Multi-constraint Spacecraft GN&C Requirement Derivation," *AAS Guidance and Control Conference*, AAS 18-095, Breckenridge, CO, 2018.

[6] Jin, K., Geller, D. K., and Luo, J., "Robust Trajectory Design for Rendezvous and Proximity Operations with Uncertainties," *Journal of Guidance, Control, and Dynamics*, Vol. 43, No. 4, 2020, pp. 741–753.

[7] Geller, D. K., Shuster, S., Woffinden, D., and Bieniawski, S., "Robust Cislunar Trajectory Optimization via Midcourse Correction and Optical Navigation Scheduling," *44th Annual AAS Guidance, Navigaiton and Control Conference*, AAS 22-065, Breckenridge, CO, 2022.

[8] Woffinden, D., Shuster, S., and Geller, S., David Kand Bieniawski, "Robust Trajectory Optimization and GN&C Performance Analysis For NRHO Rendezvous," *2022 AAS/AIAA Astrodynamics Specialist Conference*, 22-564, Charlotte, North Carolina, 2022.

[9] Geller, D., Woffinden, D., and Bieniawski, S., "Sensitivity of Optimal Midcourse Correction Scheduling for Robust Cislunar Trajectory Design," AAS 23-061, Breckenridge, CO, 2023.

[10] Goulet, T., Woffinden, D., Collins, N., and Andrews, B., "Robust Trajectory Design for Rendezvous in a Near Rectilinear Halo Orbit," AAS 23-066, Breckenridge, CO, 2023.

[11] Cavesmith, T., Woffinden, D., and Collins, N., "Angles-Only Robust Trajectory Optimization for NRHO Rendezvous," AAS 24-168, Breckenridge, CO, 2024.

[12] Calkins, G., Woffinden, D., and Putnam, Z., "Robust Trajectory Optimization for Guided Powered Descent and Landing," *2022 AAS/AIAA Astrodynamics Specialist Conference*, AAS 22-660, Charlotte, NC, 2022.

[13] Joshi, J., Woffinden, D., and Putnam, Z., "End-to-End Mars Aerocapture Analysis Using Linear Covariance Techniques and Robust Trajectory Optimization," *2022 AAS/AIAA Astrodynamics Specialist Conference*, AAS 22-678, Charlotte, NC, 2022.

[14] Woffinden, D., Eckman, R., and Robinson, S., "Optimized Trajectory Correction Burn Placement for the NASA Artemis II Mission," AAS 23-062, Breckenridge, CO, 2023.

[15] Woffinden, D., and Barton, B., "Optimized Trajectory Correction Burn Placement for NRHO Orbit Maintenance," *33rd AAS/AIAA Space Flight Mechanics Meeting*, AAS 23-364, Austin, TX, 2023.

[16] Woffinden, D., Robinson, S., Williams, J., and Putnam, Z., "Linear Covariance Analysis Techniques to Generate Navigation and Sensor Requirements for the Safe and Precise Landing - Integrated Capabilities Evolution (SPLICE) Project," *AIAA Scitech 2019 Forum*, AIAA 2019-0662, San Diego, CA, 2019.

[17] Margolis, B. W. L., and Woffinden, D., "Robust Trajectory Optimization Techniques Using a Sweeping Gradient Method and Linear Covariance Analysis," *AIAA and AAS Astrodynamics Specialist Conference*, Broomfield, CO, 2024.

[18] Luquette, R. J., "Nonlinear control design techniques for precision formation flying at Lagrange points," Ph.D. thesis, University of Maryland, College Park, Jan. 2006.

[19] Mand, K., Woffinden, D., Spanos, P., and Zanetti, R., "Rendezvous and Proximity Operations at the Earth-Moon L2 Lagrange Point: Navigation Analysis for Preliminary Trajectory Design," *AAS/AIAA Space Flight Mechanic Meeting*, AAS 14-376, Santa Fe, New Mexico, 2014.