

Adapting Covariance Propagation to Account for the Presence of Modeled and Unmodeled Maneuvers

Conrad Schiff*

a.i. solutions, Inc., Lanham, MD, 20706

This paper explores techniques that can be used to adapt the standard linearized propagation of an orbital covariance matrix to the case where there is a maneuver and an associated execution uncertainty. A Monte Carlo technique is used to construct a final orbital covariance matrix for a ‘prop-burn-prop’ process that takes into account initial state uncertainty and execution uncertainties in the maneuver magnitude. This final orbital covariance matrix is regarded as ‘truth’ and comparisons are made with three methods using modified linearized covariance propagation. The first method accounts for the maneuver by modeling its nominal effect within the state transition matrix but excludes the execution uncertainty by omitting a process noise matrix from the computation. The second method does not model the maneuver but includes a process noise matrix to account for the uncertainty in its magnitude. The third method, which is essentially a hybrid of the first two, includes the nominal portion of the maneuver via the state transition matrix and uses a process noise matrix to account for the magnitude uncertainty. The first method is unable to produce the final orbit covariance except in the case of zero maneuver uncertainty. The second method yields good accuracy for the final covariance matrix but fails to model the final orbital state accurately. Agreement between the simulated covariance data produced by this method and the Monte Carlo truth data fell within 0.5-2.5 percent over a range of maneuver sizes that span two orders of magnitude (0.1-20 m/s). The third method, which yields a combination of good accuracy in the computation of the final covariance matrix and correct accounting for the presence of the maneuver in the nominal orbit, is the best method for applications involving the computation of times of closest approach and the corresponding probability of collision, P_c . However, applications for the two other methods exist and are briefly discussed. Although the process model (‘prop-burn-prop’) that was studied is very simple – point-mass gravitational effects due to the Earth combined with an impulsive delta-V in the velocity direction for the maneuver – generalizations to more complex scenarios, including high fidelity force models, finite duration maneuvers, and maneuver pointing errors, are straightforward and are discussed in the conclusion.

Nomenclature

P_c	= collision probability
t, t_a	= generic time, time of an event (e.g., boundary condition) labeled by the index ‘a’
t_b^-, t_b^+	= time of the application of an impulsive maneuver with the superscripts ‘-’ for before and ‘+’ for after
S, S_a	= orbital state (position and velocity), orbital state at time t_a
$\mathcal{P}(t)$	= orbital state covariance at time t
$\Phi(t, t_0)$	= state transition matrix from time t_0 to t
$Q(t)$	= process noise matrix
$Q_{e/m}(t)$	= process noise matrix for environmental forces (‘e’) or maneuvers (‘m’)
λ_i	= i^{th} eigenvalue of an orbital covariance matrix
$\hat{\lambda}_i$	= eigenvector corresponding to λ_i

*Chief Scientist, a. i. solutions, Inc., 10001 Derekwood Lane, Suite 215, Lanham, MD 20706

σ_i	= standard deviation corresponding to λ_i
v_0	= nominal maneuver magnitude
v	= perturbed maneuver magnitude
$[A], A_i$	= i^{th} component of the vector \vec{A}
M_{ij}	= the component of a matrix M from the i^{th} row and j^{th} column
$\ \mathbf{M}\ $	= Matlab matrix norm for a matrix M
$ q $	= absolute value of the real number q
ε_1	= metric for comparing two orbital covariance matrices
$\Delta_1(i, j)$	= metric for comparing the (i, j) component of two orbital covariance matrices
burn	= synonym for maneuver
MC	= used as a superscript to indicate data produced using Monte Carlo
STM	= used as a superscript to indicate data produced by the linearization method
PBP	= prop-burn-prop process model used in this analysis

I. Introduction

A common problem for many low-Earth orbit missions is the need to assess and, when necessary, mitigate the risk of collision with either space debris or other assets, such as the Space Transportation System or the International Space Station. An essential step needed to perform this collision risk assessment is the calculation of the collision probability¹⁻⁶, P_c , which is the likelihood that the trajectories of the two objects in question, referred to as the Primary and Secondary, will overlap given the orbital geometry and uncertainty of each. Most of the technical approaches for computing P_c use a covariance matrix to represent the orbital uncertainties at the time of closest approach between the Primary and Secondary. Clearly, if the computation of P_c is to be useful, it must be computed well in advance of the possible collision to provide sufficient time to plan and execute a set of orbital maneuvers that will reduce the risk. This requirement means that the Primary's and Secondary's covariance matrices must be propagated forward in time from the last available orbit determination solution to the predicted time of closest approach.

When the orbital uncertainties are small compared to the size of the orbits and the propagation time is relatively short, as is usually the case in practice, the covariance matrix can be evolved using⁷

$$\mathcal{P}(t) = \Phi(t, t_0) \cdot \mathcal{P}(t_0) \cdot \Phi^T(t, t_0) + Q(t), \quad (1)$$

where $\mathcal{P}(t_0)$ is the known covariance matrix at time t_0 , $\Phi(t, t_0)$ is the state transition matrix linking local neighborhoods of the baseline trajectory at time t_0 to those at time t , $Q(t)$ is an additive term accounting for process model error, and $\mathcal{P}(t)$ is the desired covariance matrix at the time t . If neither the Primary nor the Secondary maneuvers during the time span from t_0 to t , then Eq. (1) is easy to apply. The state transition matrix is unambiguously defined by the equations of motion, and the process noise matrix represents an estimate of the small environmental forces omitted from the equations of motion used to model the motion of the Primary and Secondary.

However, in cases where one or both of the objects maneuvers during the prediction time span, application of the covariance propagation equation is not straightforward. In this case, Eq. (1) can be generalized using one of the following three methods depending on the fidelity of the model for the maneuvering object:

- **Method A – ‘burn – no Q’:** Assume that the maneuver is modeled perfectly and that its influence on the covariance matrix can be reflected solely in the state transition matrix $\Phi(t, t_0)$.
- **Method B – ‘no burn – Q’:** Omit the maneuver from the trajectory prediction and schedule the process noise term, $Q(t)$, in such a way that it grows in magnitude during the times that the maneuver is being executed. This

method may be desirable when full information regarding the maneuvering is unavailable as might be the case for when there are momentum unloads⁸.

- **Method C – ‘burn – Q’:** Assume that the maneuver, while being modeled in the process, is not modeled perfectly and that its influence on both $Q(t)$ and $\Phi(t, t_0)$ must be taken into account. This method can be regarded as a hybrid of the first two.

This paper describes the three methods in detail, compares these methods for covariance propagation against a ‘truth model’ as represented by a Monte Carlo simulation, and assesses which method is best for use in low-Earth orbit (i.e., nearly circular orbits). Specifically, Section II lays out the technical approach and Section III documents the results and the recommended method. Section IV concludes the work with an overview of work remaining and logical extensions.

II. Technical Approach

This section discusses the type of orbital evolution modeled in the process, the Monte Carlo method used to propagate an initial orbital covariance into a final ‘truth’ covariance, and the linearized state covariance propagation methods (*Methods A-C*) that were studied. All models were implemented in Matlab⁹.

A. Process Definition

For convenience, the orbital evolution process was distilled down to a simple model that was amenable to simulation. The resulting process is termed the prop-burn-prop (PBP) model for the Keplerian orbit problem. In the PBP process, only the external (i.e., environmental) force due to the gravitational attraction of a point-like Earth is included and the force due to the maneuver is modeled as an impulsive delta-V performed along the velocity direction. As the name implies, the PBP process evolves an initial state (position and velocity) S_0 from time t_0 to the burn time t_b , applies the burn, and then subsequently evolves the state to the final time t_f . In the absence of either errors in the initial state and/or errors in the burn execution, the final state S_f is uniquely determined. Figure 1 shows a schematic representation of the action of the PBP process.

There are no restrictions on the values of either t_b or t_f other than that they must be greater than t_0 . The case where $t_b = t_f$ is interpreted as a propagation followed by an impulsive maneuver with no second propagation. The case where $t_b > t_f$ is interpreted as a propagation from t_0 to t_f with no burn applied. Both of these special cases were used in testing the algorithm with the former case ($t_b = t_f$) specifically used for confirming the functional form of $Q(t)$.

B. Monte Carlo Truth Model

With the basic PBP process in hand, the next step is to take into account uncertainties in the initial conditions and in the maneuver execution. Initial state uncertainties are modeled by attaching a covariance matrix $\mathcal{P}(t_0)$ to the initial state S_0 . The covariance matrix is then diagonalized to obtain a set of eigenvalues $\{\lambda_1, \lambda_2, \dots, \lambda_6\}$ and eigenvectors $\{\hat{\lambda}_1, \hat{\lambda}_2, \dots, \hat{\lambda}_6\}$. The square root of the eigenvalues are then considered to be the standard deviations

$$\begin{aligned} \sigma_1 &= \sqrt{\lambda_1} \\ \sigma_2 &= \sqrt{\lambda_2} \\ &\vdots \\ \sigma_6 &= \sqrt{\lambda_6} \end{aligned} \tag{2}$$

of the random fluctuations that will develop about the initial state. By sampling these random fluctuations, a perturbed initial state can be constructed using

$$\mathbf{S} = \mathbf{S}_0 + \sum_{i=1}^6 \sigma_i \text{randn}(0,1) \hat{\lambda}_i, \quad (3)$$

where $\text{randn}(0,1)$ is a Gaussian random number generator with zero mean and a unit standard deviation.[†] An exactly analogous situation is applied to the maneuver. If the nominal maneuver magnitude is given by v_0 then the perturbed magnitude is given by

$$v = (1 + \sigma_b \text{randn}(0,1))v_0, \quad (4)$$

where σ_b is an expected execution error in the maneuver. The perturbed initial state \mathbf{S} is then propagated from t_0 to t_b where a maneuver of magnitude v is performed and then subsequently propagated to t_f . By repeating this process N times and collecting statistics on the resulting final states, a final covariance matrix can be constructed directly as¹⁰

$$\mathcal{P}^{STM}(t_f) = E \left[(\mathbf{S} - \bar{\mathbf{S}}) \cdot (\mathbf{S} - \bar{\mathbf{S}})^T \right], \quad (5)$$

where $\bar{\mathbf{S}}$ is an average over all of the final states.

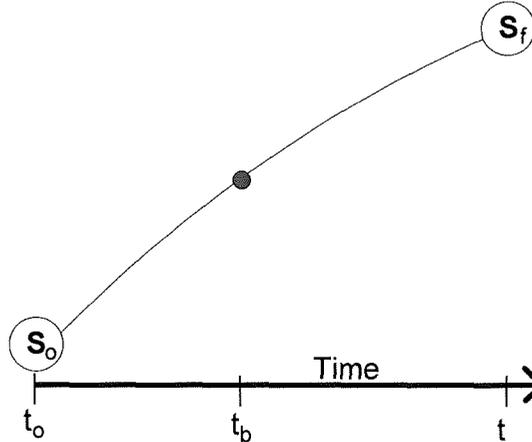


Figure 1: Schematic representation of the 'prop-burn-prop' (PBP) process used in the paper. The gray circles represent the initial state, \mathbf{S}_0 , and the final state, \mathbf{S}_f , and the red circle represents the impulsive maneuver.

Figure 2 shows a schematic of the Monte Carlo method applied to the PBP process. Just as in Figure 1, the gray circles represent initial and final states at t_0 and t_f , respectively. The red circles of different sizes represent different magnitude impulsive burns applied at t_b , all performed along the local velocity direction. The 'khaki' ellipses at t_0 and t_f represent the initial and final covariance matrices. The final covariance matrix, calculated using Eq. (5), is taken as the truth model for the linearized covariance propagation and will, hereafter, be denoted as $\mathcal{P}^{MC}(t_f)$.

[†] There is no restriction to a Gaussian distribution and in fact the methodology has been used with uniform random numbers with similar results.

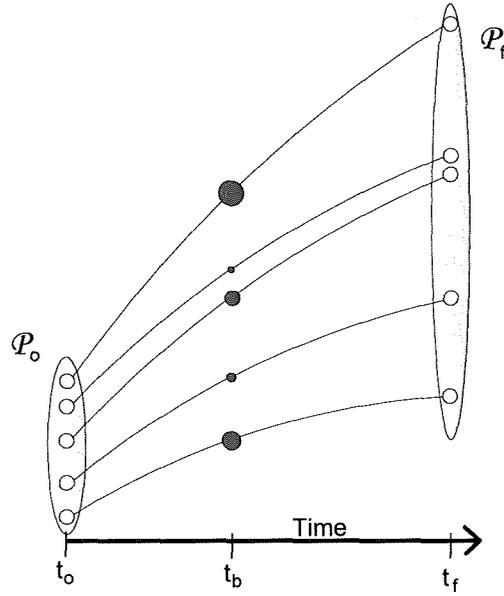


Figure 2: A schematic representation of the Monte Carlo method applied to the PBP process. The gray circles represent initial and final states at t_0 and t_f , respectively. The red circles of different sizes represent different magnitude impulsive burns applied at t_b , all performed along the local velocity direction.

C. Linearized Covariance Propagation

The next step is to construct the final covariance $\mathcal{P}(t)$ using linearized covariance propagation. The standard formula for doing this is⁷

$$\mathcal{P}(t) = \Phi(t, t_0) \cdot \mathcal{P}(t_0) \cdot \Phi^T(t, t_0) + Q(t), \quad (6)$$

where $\mathcal{P}(t_0)$ is the same initial covariance matrix that was used in the Monte Carlo, $\Phi(t, t_0)$ is the state transition matrix, and $Q(t)$ is an additive term accounting for process model error. The formal definition of the state transition matrix is

$$\Phi(t, t_0) = \left. \frac{\partial \mathbf{S}(t)}{\partial \mathbf{S}(t_0)} \right|_{\mathbf{S}^*(t)}, \quad (7)$$

where $\mathbf{S}^*(t)$ is the reference trajectory obtained evolving \mathbf{S}_0 with the desired process model. In this analysis, the PBP process includes an impulsive maneuver and Eq. 7 is computed using the finite central-difference approximation

$$\Phi(t, t_0)_{i,j} \cong \frac{S^*_i(t_f; \mathbf{S}_0 + h \hat{e}_j) - S^*_i(t_f; \mathbf{S}_0 - h \hat{e}_j)}{2h}, \quad (8)$$

where the indices $i, j = 1 \dots 6$ determine the row and column of the state transition matrix, \hat{e}_j is a unit vector with components $[\hat{e}_j]_i = \delta_{ji}$, h is a small number, and $\mathbf{S}^*(t; \mathbf{S}_0)$ is the state at time t that results from applying the PBP process with an error-free maneuver (i.e., an nominal maneuver magnitude of v_0) to the state \mathbf{S}_0 . It is worth

emphasizing that the reason for using Eq. (8) rather than a process equation of the form¹¹ $\dot{\Phi} = A\Phi$ is because the finite-difference method ensures that the influence of the nominal impulsive maneuver is reflected in the state transition matrix. This observation is what justifies the use of the term ‘burn’ in the definitions of *Method A* and *Method C*. The form of $Q(t)$ in Eq. (6) is decomposed as follows

$$Q(t) = Q_e(t) + Q_m(t), \quad (9)$$

where $Q_e(t)$ accounts for the process noise in all environmental forces and $Q_m(t)$ accounts for noise in the maneuver and is non-zero only during the execution of the maneuver. Usually, the form of $Q_e(t)$ is chosen to account for errors in the environmental forces. Since the Monte Carlo is regarded as truth and since the linearized covariance propagation uses the same process, $Q_e(t) = 0$, in this example. In addition, because $Q_m(t)$ is associated with the maneuver uncertainty, it will be a non-zero term only during the execution of the maneuver.

The specific implementations of the three methods can now be discussed. The implementation of *Method A*, burn – no Q, is straightforward. The initial state covariance is propagated until the final time t_f using

$$\mathcal{P}(t_f) = \Phi(t_f, t_0) \cdot \mathcal{P}(t_0) \cdot \Phi^T(t_f, t_0), \quad (10)$$

where $\Phi(t_f, t_0)$ is constructed from Eq. (8) and includes the impulsive maneuver.

The implementation of *Method B*, no burn – Q, requires a little more work. First the covariance matrix is propagated until the burn time using

$$\mathcal{P}(t_b^-) = \Phi(t_b^-, t_0) \cdot \mathcal{P}(t_0) \cdot \Phi^T(t_b^-, t_0), \quad (11)$$

where t_b^- is the burn time just before the application of the maneuver. The process noise is then added in to yield

$$\mathcal{P}(t_b^+) = \mathcal{P}(t_b^-) + Q_m(t_b), \quad (12)$$

which is then propagated to the final time using

$$\mathcal{P}(t_f) = \Phi(t_f, t_b^+) \cdot \mathcal{P}(t_b^+) \cdot \Phi^T(t_f, t_b^+). \quad (13)$$

The specific form of $Q_m(t)$ will be specified below. By construction, none of the state transition matrices in Eqs. (11) and (13) have the maneuver's effects included.

The implementation of *Method C*, burn – Q, is nearly the same as for *Method B*. The specific change that occurs in Eq. (11), which now becomes

$$\mathcal{P}(t_b^+) = \Phi(t_b^+, t_0) \cdot \mathcal{P}(t_0) \cdot \Phi^T(t_b^+, t_0), \quad (14)$$

is designed so that the state transition matrix includes the influence of the impulsive maneuver. To emphasize that the covariance matrix calculated in Eqs. (10), (13), and (14) are calculated using linear theory, they will be referred to as \mathcal{P}^{STM} .

III. Results

Before presenting the results, it is important to discuss two points. The first is that perfect agreement between a final covariance matrix produced using Monte Carlo (MC) and one produced using the linearized propagation (STM) is not expected. This observation is based on the fact that there will always be random fluctuations in any parameter computed using Monte Carlo due to the finite sample size and because the Monte Carlo trials capture nonlinearities that the linearized propagation cannot capture. The second point is that, in order to decide when one matrix is sufficiently close to another, some sort of metric is required. Two such metrics are employed in this work. The first metric, which is based on using the default matrix norm in Matlab, is defined as

$$\varepsilon_1 = \frac{\|\mathcal{P}^{MC}(t_f) - \mathcal{P}^{STM}(t_f)\|}{\|\mathcal{P}^{MC}(t_f)\|} \cdot 100, \quad (15)$$

where the matrix norm $\|\bullet\|$ is defined as the maximum singular value obtained from a singular value decomposition. The second metric employed is one defined as

$$\Delta_1(i, j) = \mathcal{P}_{ij}^{MC}(t_f) - \mathcal{P}_{ij}^{STM}(t_f). \quad (16)$$

This latter metric is also used in the computation of $Q_m(t)$, which has only one component in a local frame in which the velocity direction is a primary axis.

The first simulation run is for **Method A, burn – no Q**. The values for t_b and t_f were chosen to be 60 seconds and 600 seconds, respectively, and the impulsive maneuver magnitude was varied from 0.1 m/s to 20 m/s. A Monte Carlo simulation with 10,000 trials was used to construct a representation of the final covariance for each value of the maneuver for magnitude (execution) uncertainties of 0.00 and 0.05. Figure 3 shows the results for the ε_1 metric for both values of burn uncertainties. The results clearly show that as long as the maneuver is deterministic (i.e., has no execution uncertainty) then the agreement between the MC and STM representations of the final covariance matrices is excellent. However, the introduction of a maneuver uncertainty causes the deviations between the two representations to grow rapidly.

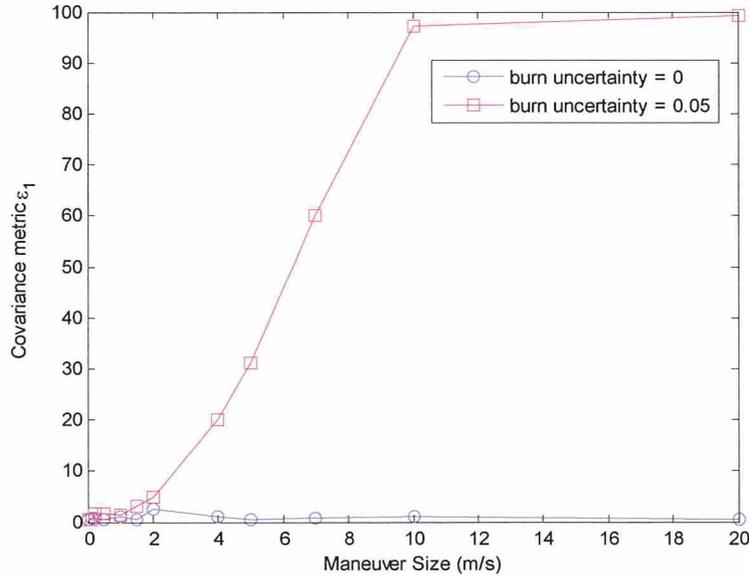


Figure 3: Evaluation of the covariance metric ε_1 for **Method A, burn – no Q**, for maneuver magnitude uncertainties of 0 and 0.05.

Clearly, the application of a process noise matrix is needed to account for the maneuver uncertainty. A reasonable guess for the form of $Q_m(t_b)$ would be

$$Q_m(t_b) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & (\sigma_b)^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (16)$$

expressed in the inertial VNBi coordinate system and where σ_b is defined in Eq. (4). The inertial VNBi coordinate system is defined by the inertial set of basis vectors

$$\hat{V} = \frac{\bar{v}}{|\bar{v}|}, \quad (17a)$$

$$\hat{N} = \frac{\bar{r} \times \bar{v}}{|\bar{r} \times \bar{v}|}, \quad (17b)$$

and

$$\hat{B} = \hat{N} \times \hat{V}, \quad (17c)$$

where \bar{r} and \bar{v} are the spacecraft's position and velocity just prior to the application of the maneuver, respectively.

To confirm that the choice in Eq. 16 is correct, a 'reduced' simulation using the burn – no Q method was made in which $t_f = t_b$. The resulting MC and STM covariance matrices were transformed to VNBi coordinates and then subjected to the $\Delta_1(i, j)$ metric. The only non-zero result for $\Delta_1(i, j)$ was for the 4-4 component corresponding to the velocity-velocity portion of the covariance matrix and in the direction of the spacecraft's velocity. The comparison between this result and the one predicted by Eq. (16) is shown in Figure 4. The conclusion from this result is that the functional form of $Q_m(t_b)$ in Eq. (16) is correct.

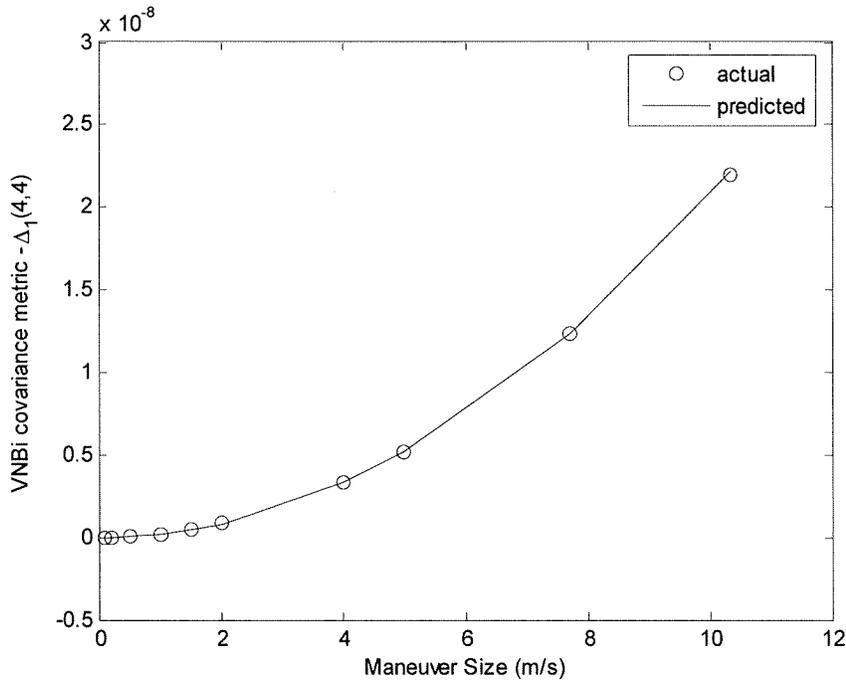


Figure 4: The value of the $\Delta_1(i, j)$ metric for the 4-4 component for simulation data compared to the value of $Q_m(t_b)$ predicted by Eq. (16).

Finally, simulations were run for **Method B** (no burn – Q) and **Method C** (burn – Q) with the parameters set to be the same as for **Method A**. Namely, the values for t_b and t_f were chosen to be 60 seconds and 600 seconds, respectively, the impulsive maneuver magnitude was varied from 0.1 m/s to 20 m/s, and a 10,000-trial Monte Carlo data set was used to construct a representation of the final covariance for a magnitude (execution) uncertainty of 0.05. Figure 5 shows the results for the ε_1 metric for both methods. Statistically, there is no difference between the methods in reproducing the final covariance matrix (the statistics on the ε_1 metric data in Figure 5 are $\mu_B = 0.488$ and $\sigma_B = 0.242$ for **Method B**, no burn – Q and $\mu_C = 0.495$ and $\sigma_C = 0.239$ for **Method C**, burn – Q). It is worth noting that the reason for the statistical equivalence is that differences in the state transition matrices with and without the maneuver are small, even for maneuver magnitudes of 10 and 20 m/s, and are insignificant compared to the process noise and the noise of a Monte Carlo simulation. As t_f becomes much greater than t_b , these differences will become more pronounced, and the expectation is that the results from **Method B** will get progressively worse. In addition, as seen in Figure 6, the final state is significantly off when the maneuver is not modeled in **Method B**. Since the primary goal of this analysis was its application to the computation of P_c , anything that fails to model the maneuver would cause a substantial error in the computation of miss distance and time of closest approach, each of which is used in determining the probability of collision. From this consideration alone, **Method C**, burn – Q, is preferable.

However, there are some practical advantages afforded by **Method A** and **Method B** beyond the goal of computing P_c . **Method A** can be employed in pre-launch delta-V budget analysis when the maneuver uncertainty is considered to be negligible in comparison to the other perturbations in the problem, or when it is desired to see how much of the delta-V is required to solely accommodate the orbit determination error. **Method B**, on the other hand, is well-suited for use in orbit determination using a Kalman filter. In this case, knowledge of the maneuver may not be available or may be too difficult to model. As long as the maneuver uncertainty can be properly reflected in the process noise, the measurements should accurately update the state to reflect the unmodeled force.

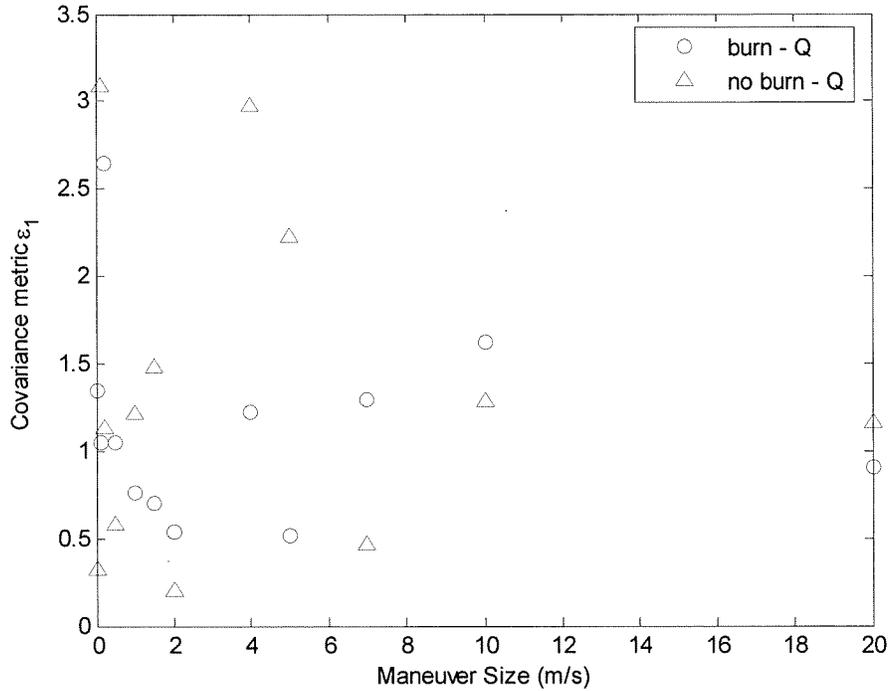


Figure 5: Evaluation of the covariance metric ϵ_1 for *Method B*, no burn - Q, and *Method C*, burn - Q, as a function of maneuver magnitudes with an uncertainty of 0.05.

IV. Conclusion

The aim of this analysis was to examine three methods for adapting linear covariance propagation to account for the presence of maneuvers with associated uncertainties. The method that was most applicable to calculating the probability of collision was that which took into account the presence of the maneuver in the computation of the state and the corresponding STM while accounting for the maneuver uncertainty by applying process noise (**Method C**, the burn - Q method).

Although the process modeled in this analysis is based on simple point-mass gravity for the Earth and impulsive maneuvers performed along the velocity direction, generalizations to higher fidelity models is relatively straightforward. The definition in Eq. (7) applies equally well for computation of a state transition matrix with complex force models and impulsive or finite maneuvers. Even in the case where analytic partial derivatives cannot be derived, a numerical approximation is always available using Eq. (8). Somewhat more difficult is the computation of the process noise matrix for more complex maneuvers. Clearly, the form of $Q_m(t)$ must be non-zero during the times when the maneuver is occurring, regardless of whether the maneuver is modeled as an impulse or a finite burn. An arbitrary thrust direction is easily modeled by adopting a coordinate system with a primary axis pointing along this direction. Most maneuvers are well modeled by associating any uncertainty with the magnitude (i.e, performance). In this case, the process noise then takes the form as in Eq. (16) and a standard transformation is all that is needed to convert to the coordinate system used to express the covariance matrix (see Ref. 8 for numerous examples). In the event that pointing uncertainties are also important, it is likely that the functional form of the process noise must change. Specifically, $Q_m(t)$ will take on a block diagonal form in the lower 3x3 sub-matrix.

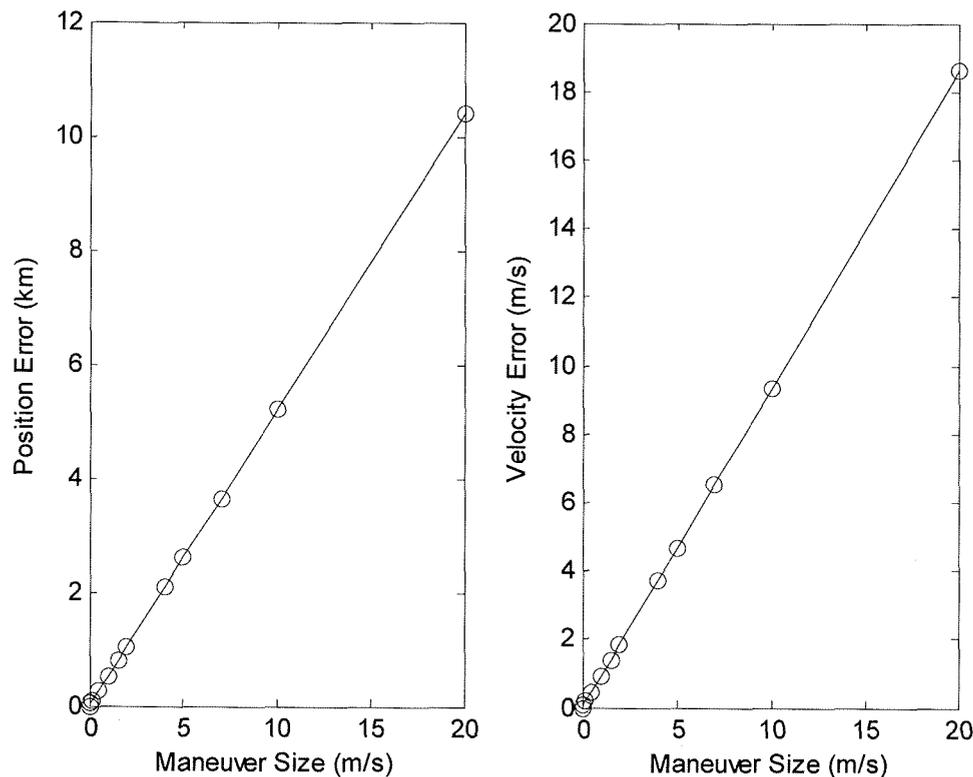


Figure 6: The magnitude of the position and velocity differences as a function of maneuver size for **Method B, no burn - Q.**

These generalizations and their applications to the EOS constellation will be explored in the coming year. The intent is to build upon **Method C, the burn - Q method,** in a way that allows an EOS mission, faced with the need to avoid a close approach by performing a maneuver, to evaluate the maneuver plan based on a calculation of P_c post-burn.

References

- ¹Akella, Maruthi, R, and Alfriend, Kyle T., "Probability of Collision Between Space Objects," *Journal of Guidance, Control, and Dynamics*, Vol. 23, No. 5, September-October 2000, pp. 769-772.
- ²Patera, Russell, "General Method for Calculating Satellite Collision Probability," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 4, July-August 2001, pp. 716-722.
- ³Patera, Russell, "Satellite Collision Probability for Nonlinear Relative Motion," *Journal of Guidance, Control, and Dynamics*, Vol. 26, No. 5, September-October 2003, pp. 728-733.
- ⁴Chan, Ken, "Short-Term vs Long-Term Spacecraft Encounters," *AIAA-AAS Astrodynamics Specialist Conference and Exhibit*, 16-19 August 2004, Providence, Rhode Island, AIAA 2004-5460
- ⁵Chan, Ken, "Improved Analytical Expressions for Computing Spacecraft Collision Probabilities," *AIAA-AAS Space Flight Mechanics Meeting*, 9-13 February 2003, Ponce, Puerto Rico, AAS 03-184
- ⁶Patera, Russell, "Collision Probability for Larger Bodies Having Non-Linear Relative Motion," *AAS/AIAA Astrodynamics Specialist Conference*, 7-11 August 2005, Lake Tahoe, California, AAS 05-309
- ⁷Gelb, Arthur (ed.), *Applied Optimal Estimation*, The M.I.T. Press, Cambridge, Massachusetts, 1974, p.76.
- ⁸Long, Anne C., Lee, Taesul, "Flight Dynamics (FD) Global Positioning System (GPS) Enhanced Onboard Navigation System (GEONS) Mathematical Specifications, Version 2, Release 2.5," NASA FDF-59-005, December 20, 2005.
- ⁹Matlab, Software Package, Ver. 7.0.1.24704 (R14) Service Pack 1, The MathWorks, Natick, Massachusetts, 1984.
- ¹⁰Bevington, Philip R., and Robinson, D. Keith, *Data Reduction and Error Analysis for the Physical Sciences*, 2nd ed., McGraw-Hill, New York, 1992, Chap. 11.
- ¹¹Gelb, Arthur (ed.), *Applied Optimal Estimation*, The M.I.T. Press, Cambridge, Massachusetts, 1974, p.58.