Precomputing Process Noise Covariance for Onboard Sequential Filters

Corwin G. Olson[1] and Ryan P. Russell[2]
University of Texas at Austin, Austin, Texas, 78712

J. Russell Carpenter[3]
NASA Goddard Space Flight Center, Greenbelt, Maryland, 20771

Process noise is often used in estimation filters to account for unmodeled and mismodeled accelerations in the dynamics. The process noise covariance acts to inflate the state covariance over propagation intervals, increasing the uncertainty in the state. In scenarios where the acceleration errors change significantly over time, the standard process noise covariance approach can fail to provide effective representation of the state and its uncertainty. Consider covariance analysis techniques provide a method to precompute a process noise covariance profile along a reference trajectory, using known model parameter uncertainties. The process noise covariance profile allows significantly improved state estimation and uncertainty representation over the traditional formulation. As a result, estimation performance on par with the consider filter is achieved for trajectories near the reference trajectory without the additional computational cost of the consider filter. The new formulation also has the potential to significantly reduce the trial-and-error tuning currently required of navigation analysts. A linear estimation problem as described in several previous consider covariance analysis publications is used to demonstrate the effectiveness of the precomputed process noise covariance, as well as a nonlinear descent scenario at the asteroid Bennu with optical navigation.

---

[1] Research Associate, Space and Geophysics Laboratory, Applied Research Laboratories, University of Texas at Austin, 10000 Burnet Rd, Austin, TX 78758.
[2] Associate Professor, Department of Aerospace Engineering and Engineering Mechanics, 1 University Station, C0600, University of Texas at Austin, Austin, TX 78712-0235. Senior Member AIAA.
[3] Aerospace Engineer, Navigation and Mission Design Branch, NASA Goddard Space Flight Center, Greenbelt, MD 20771. Associate Fellow AIAA.

I. Introduction

In almost all statistical estimation problems that involve dynamical systems, there are mismodeled or unmodeled forces that act to cause errors in the expected state of the system. Thus measurements are needed to correct the state, and a formal covariance is also computed to provide a measure of the uncertainty of that estimate. Between measurements, both the estimated state and the covariance are propagated, and the covariance is also inflated to account for the increase in uncertainty introduced by the mismodeled and unmodeled forces. This inflation of the state covariance is commonly known as "process noise covariance" (PNC) [1], and should not be confused with the traditional definition of process noise as the white noise process included in the dynamical model.

The standard approach for determining the appropriate amount of PNC is often a lengthy trial-and-error procedure that attempts to match the formal covariance to the covariance computed from the distribution of errors generated by a Monte Carlo simulation. The process noise is typically modeled as an uncorrelated sequence of Brownian increments, which may enter the deterministic plant directly, or via shaping filters that impose a correlation structure. In other words, the stochastic differential equation can become correlated from the homogeneous portion, but the forcing function is uncorrelated in time and thus the process as a whole is Markov (i.e. the future evolution depends only on the current state). However, there are many scenarios where the magnitude of known unmodeled or mismodeled perturbations can change significantly over the course of a reference trajectory, which can make tuning the standard PNC challenging. If the perturbation magnitude increase is significant enough, the standard PNC model may lead to a divergent filter.

An example where significant state-dependent perturbation magnitudes occur is a spacecraft descent scenario to the surface of a newly explored celestial small body such as an asteroid or comet. Onboard spacecraft navigation systems often use truncated spherical harmonic gravity models of the bodies they orbit in order to reduce computational requirements for propagation of the spacecraft position and velocity state about the body. Additionally, higher order gravity terms

may not be observable in the higher orbits before a descent maneuver. These higher order gravity terms, along with errors in the estimated lower order gravity terms, can introduce significant errors in the trajectory as the spacecraft descends to the surface. These errors can lead to degraded filter performance and strongly affect the spacecraft's ability to land accurately and locate landmarks for optical navigation. More generally, Wright [2] has shown that the errors resulting from using imperfect spherical harmonic gravity coefficients and a truncated gravity model are correlated in time. For these autocorrelated noise processes, the standard PNC approach can fail to provide effective representation of the state uncertainty as well as effective estimation of the state. Thus a better method for computing PNC is needed.

The method derived in this paper computes the PNC for each interval between measurements of a reference trajectory by mapping model uncertainty into state uncertainty, as is done in Consider Covariance Analysis (CCA) and consider filters such as the Schmidt-Kalman Filter (SKF) [3]. CCA is covered extensively by Tapley [1] for the linearized method, and Lisano [4] using sigma point transforms. However, in CCA the consider covariance is computed independently and does not affect the performance of the underlying filter.

In the SKF, first introduced by Schmidt in 1966 [3, 5], the uncertainties in the considered model parameters do affect the estimation of the state variables, and can provide significant improvement to the state estimation without separate PNC added. As a result, PNC tuning is only necessary for unmodeled perturbations that are not related to the consider parameters, and any state-dependent perturbations are properly included. However, with the additional considered parameters the SKF becomes more computationally demanding than the standard Kalman Filter (KF) (assuming none of the considered model parameters are included in the KF), and may prove computationally intractable for onboard systems. The additional computational burden can be partially mitigated by avoiding unnecessary calculations associated with consider states that are structurally uncorrelated from the vehicle dynamics, such as measurement biases. However, such an approach does not apply when the consider covariance terms directly relate to the vehicle dynamics, such as the gravity spherical harmonic coefficients. The difference between the SKF and KF becomes more pronounced as the number of considered parameters increases, particularly when considering many

model parameters (e.g. a gravity field with thousands of terms). Thus when considering highly limited computational systems such as those onboard a spacecraft, the question arises: is it possible to obtain most or all of the benefits of the SKF without the higher computational costs?

The answer to this question is in fact yes, though some trade-offs must occur: the PNC must be precomputed along a reference trajectory, which must be available and reasonably approximate the true trajectory, and the precomputed PNC must be stored onboard in the form of a table or interpolation coefficients. The precomputation of the PNC profile along the reference trajectory involves computing the difference of the propagated state covariance for each propagation interval between a covariance propagation including the consider uncertainties (SKF) and a covariance propagation that does not include consider uncertainties (KF), as described in the "Method of Solution" section III. The PNC precomputation can be performed using linearized techniques involving the Extended Kalman Filter (EKF) and Extended Schmidt-Kalman Filter (ESKF), or using sigma point transformations. The most common sigma point transformation is the Unscented Kalman Filter (UKF), the unscaled form of which is applied by Stauch and Jah to the SKF to derive the Unscented Schmidt-Kalman Filter (USKF) [6]. "Unscaled" indicates that no central sigma point value is computed and thus no tuning parameters are necessary. Stauch and Jah obtain the same final form as in Zanetti and DeMars [7], but through a different derivation. Note that, like the ESKF, the USKF does not resolve the issue of onboard computational limits. However, the USKF is a key element of the PNC precomputation algorithm, as explained in more detail in the "Method of Solution" section III.

DeMars and Bishop [8] use the linearized approach to map model uncertainties to estimation state uncertainties for the purposes of precomputing PNC for onboard navigation. They also analyze interpolation methods for precomputed PNC between measurement updates. The current work takes these efforts a step further by incorporating the precomputed PNC into the onboard state and covariance estimation. In addition, we show how a covariance analysis involving covariance corrections is necessary to provide the appropriate cross-covariance contributions to the precomputed PNC profile. Without updates, these cross-covariance contributions grow without bound; without them, the precomputed PNC profile is incomplete following an expected measurement update.

While there are many advantages to precomputing the PNC, the technique assumes the truth trajectory will not deviate too far from a reference trajectory. Another assumption is that the error in the dynamics model parameters is roughly equivalent to the parameter uncertainties used in the precomputation of the PNC. If the error in the dynamics model parameters is significantly different than the assumed uncertainty, or there are totally unmodeled perturbations, then the precomputed PNC might be significantly different than the level of truth perturbations. In this scenario, adding a blindly tuned standard PNC (on top of the precomputed PNC described in this paper) might be necessary to account for the mismodeled and unmodeled perturbations. A limited analysis is performed to investigate differences in the model error versus the assumed model uncertainties.

Also note that this paper does not address another important tuning parameter in an onboard filter: the measurement noise uncertainty. Early work by Bryson and Henrikson [9] investigates filtering, prediction, and smoothing improvements when measurement data includes sequentially correlated noise. A comparison of the algorithms derived by Bryson and Henrikson versus application of the new method using measurement model parameters as consider states in order to compute appropriate measurement noise is saved for future analysis. Other relevant works regarding the treatment of bias in the measurement and dynamical models used by the estimation filter, which may serve as benchmarks for future work, include Friedland [10], Ignagni [11], Hough [12], and Zanetti and Bishop [13].

II. Process Noise

Before the PNC precomputation algorithm is derived, the relevant process noise terms must be defined. Consider a random dynamical process obeying the linear differential equation

$$\Delta\dot{x}(t) = F(t)\Delta x(t) + G(t)u(t) \tag{1}$$

with *a priori* value Δx_0 at time $t = 0$. $\Delta x(t)$ is the deviation from the reference state, $u(t)$ is the system model error, $F(t)$ describing the system dynamics, and $G(t)$ maps the model error to the state error. If linearizing a non-linear system, $F(t)$ and $G(t)$ are partial derivative matrices

evaluated at the nominal state. "Formally" integrating Eq. 1 from $t_0 = 0$ to $t = t_k$ gives

$$\Delta x(t_k) = \Phi(t_k, t_0)\Delta x_0 + \int_0^{t_k} \Phi(t_k, t')G(t')u(t')dt' \tag{2}$$

where $\Phi(t_k, t_0)$ is the state transition matrix. Define the state error covariance at time t_k as

$$P_k \equiv E\left[\Delta x(t_k)\Delta x^T(t_k)\right] \tag{3}$$

and the process noise autocovariance as

$$Q(t, t') \equiv E\left[u(t)u^T(t')\right] \tag{4}$$

where E is the expectation operator (which computes the mean of the argument), and both $\Delta x(t_k)$ and $u(t)$ are zero-mean. There is no assumption that the process noise is uncorrelated at separate times. The covariance at time t_k is

$$P_k = \Phi(t_k, t_0)P_0\Phi^T(t_k, t_0) + \int_0^{t_k}\int_0^{t_k} \Phi(t_k, t')G(t')Q(t', t'')G^T(t'')\Phi^T(t_k, t'')dt'dt'' \tag{5}$$

assuming that $u(t)$ is not correlated with the *a priori* value Δx_0; but *not* assuming that $u(t)$ will be uncorrelated with $\Delta x(t)$ at other times [1].

Assuming that $Q(t, t') = Q(t)\delta(t - t')$ (i.e. the process noise increments are uncorrelated at separate times, otherwise known as a white-noise process when the process noise is also zero-mean), one of the integrals in Eq. 5 is annihilated. More fundamentally, since future noise increments depend in no way on the system's past history, the present moments of $\Delta x(t_k)$ give the only information useful for predicting future values. This characteristic is known as the Markov property.

A common application for this assumption of uncorrelated process noise increments is the derivation of a PNC for a state consisting of the spacecraft position and velocity. This derivation is provided in detail in Tapley, et al. [1], which treats the process noise as white noise (i.e. no gravity gradient information is used). With the additional assumption that the time interval $t_0 = 0$ to $t = t_k$ corresponds to the propagation time from the previous measurement time to the current measurement time, Δt_k, the PNC for a spacecraft position and velocity can be expressed as

$$Q_k = q\begin{bmatrix} \frac{1}{3}I_3\Delta t_k^3 & \frac{1}{2}I_3\Delta t_k^2 \\ \\ \frac{1}{2}I_3\Delta t_k^2 & I_3\Delta t_k \end{bmatrix} \tag{6}$$

where I_3 is the 3x3 identity matrix, Δt_k is the propagation time from the previous measurement time to the current measurement time, and q is the single tuning parameter [1]. Equation 6 is a common PNC form, which we shall hereafter refer to as the traditional form.

By contrast, Wright [2] observes that errors in the gravity model used for orbit determination induce a noise process that is correlated in time. In other words, Wright shows that when Eq. 1 describes errors in predicting the evolution of a satellite orbit due to gravity model errors, there is a correlation structure among future and past noise increments that makes Eq. 1 a non-Markov process, which $Q(t,t') = Q(t)\delta(t-t')$ fails to capture. He provides an alternative, autocorrelated, but approximately Markov, model suitable for circular or near-circular orbits, based on projection of the gravity model error covariance onto a sphere. However, there are many scenarios that do not meet the criteria required for Wright's model, such as a descent trajectory to the surface of a small body. Thus an alternative method is needed that can determine PNC for any mission scenario.

III. Method of Solution

Denote the double-integral PNC term of Eq. 5 as

$$Q_d(t_k,t_0) = \int_0^{t_k}\int_0^{t_k} \Phi(t_k,t')G(t')Q(t',t'')G^T(t'')\Phi^T(t_k,t'')dt'dt'' \tag{7}$$

This paper investigates $Q_d(t_k,t_0)$ for arbitrary force model errors, with the aim to develop approximations suitable for use in onboard robotic navigation. In other words, the PNC used to inflate the state uncertainty across propagation intervals in order to appropriately reflect dynamical model perturbations is precomputed along a reference trajectory. To understand how this can be done, it helpful to first describe the components of the covariance propagation.

Let $x(t)$ denote the truth Cartesian state of a satellite, which evolves according to the propagation function ϕ_t:

$$x(t) = \phi_t(t, x_0, c) \tag{8}$$

where c is a vector of uncertain model parameters. Let $\hat{x}(t)$ and $\hat{c}$ respectively be the estimate of $x(t)$ and c maintained by a navigation filter, which assumes the nominal model propagation function

$$\hat{x}(t) = \phi_n(t, \hat{x}_0, \hat{c}) \tag{9}$$

Inherent in the nominal propagation function described by equation 9 is the standard assumption that

$$E[f(x)] = \int f(x)p(x)dx \approx f\left(\int x\, p(x)dx\right) = f(E[x]) \tag{10}$$

where $p(x)$ is the probability density function. Equation 10 is derived by dropping all but the first term of the power series for the nonlinear propagation function about the current state estimate (see Gelb [14]).

The propagation function ϕ_n used for the nominal model is not necessarily the same as the truth propagation function ϕ_t. For example the gravity field maximum degree and order is likely significantly truncated for an onboard dynamical model as compared to the truth dynamical model. The state error can then be written as

$$\Delta x(t) = \phi_t(t, x_0, c) - \phi_n(t, \hat{x}_0, \hat{c}) \tag{11}$$

To isolate the error sources present in the state error, $\phi_t(t, \hat{x}_0, c)$ is added and subtracted from equation 11 to yield

$$\Delta x(t) = [\phi_t(t, x_0, c) - \phi_t(t, \hat{x}_0, c)] + [\phi_t(t, \hat{x}_0, c) - \phi_n(t, \hat{x}_0, \hat{c})] \tag{12}$$

The first bracketed expression is the error due solely to the initial state error, and the second bracketed expression is the error due solely to force model errors. Rewriting these expressions as

$$\Delta x(t) = e_{x0}(t) + e_c(t) \tag{13}$$

the covariance of the final error can be written as

$$\begin{aligned} P(t) &= E\left[e_{x0}(t)e_{x0}^T(t) + e_c(t)e_c^T(t) + e_c(t)e_{x0}^T(t) + e_{x0}(t)e_c^T(t)\right] \\ &= P_{x0}(t) + P_{cc}(t) + P_{cx0}(t) + P_{cx0}^T(t) \end{aligned} \tag{14}$$

Note that there are three contributions to the final mapped covariance: $P_{x0}(t)$ (contribution from the initial state uncertainty), $P_{cc}(t)$ (directly mapped contribution from the model uncertainty), and $P_{cx0}(t) + P_{cx0}^T(t)$ (cross-correlation contribution between the state uncertainty and the model uncertainty). The $P_{x0}(t)$ term is typically computed in the onboard filter, using the limited dynamical models and assumed model values. The onboard filter also adds PNC to the mapped covariance,

under the assumption that the dynamics are not modeled perfectly. Thus it is ideal to use the sum of the $P_{cc}(t)$ and $P_{cx0}(t) + P_{cx0}^T(t)$ terms as this PNC. Comparing equation 14 to equation 5 and 7, the expression for $Q_d(t_k, t_0)$ becomes

$$\begin{aligned} Q_d(t_k, t_0) &= P_{cc}(t_k, t_0) + P_{cx0}(t_k, t_0) + P_{cx0}^T(t_k, t_0) \\ &= P(t) - P_{x0}(t) \end{aligned} \tag{15}$$

However, due to onboard computational limits, these values are precomputed, which requires that they be computed along a known reference trajectory. To precompute $Q_d(t_k, t_0)$ for each interval of a reference trajectory, the new approach makes use of consider covariance and filtering methods such as the ESKF [5], USKF [6], and the newly developed Additive Divided Difference Schmidt Kalman Filter (ADSF) described in the next section. The resulting precomputed PNC can be used in computationally efficient filters such as the EKF. Note that the result of the subtraction in equation 15 has the potential to become negative definite if the reference trajectory used differs greatly from the truth trajectory; for this reason, a small additional standard PNC quantity may become necessary.

These covariance contributions can be mapped via linearized integration or by sigma point transformations. The linearized covariance propagation equations that are numerically integrated are obtained from the SKF, as described by Woodbury [5] and DeMars [8]:

$$\begin{aligned} \dot{\hat{x}}(t) &= F(t)\hat{x}(t) + G(t)c \\ \dot{P}_{xx}(t) &= F(t)P_{xx}(t) + P_{xx}(t)F^T(t) + P_{xc}(t)G^T(t) + G(t)P_{xc}^T(t) \\ \dot{P}_{xc}(t) &= F(t)P_{xc}(t) + G(t)P_{cc} \end{aligned} \tag{16}$$

where c is the nominal vector of considered model parameters (which are in general not equal to the truth model parameters). Note that the final term $G(t)Q(t)G^T(t)$ from Woodbury [5] is not included because no traditional PNC ($Q(t)$) is included when precomputing the PNC profile (the same formulation as in DeMars [8]). An exploration of how adding traditional PNC to the precomputation of the PNC profile affects the resulting PNC profile is saved for future work.

In equation 16, the integration of $P_{xx}(t)$ is dependent on $P_{xc}(t)$. Thus the final two terms of the $\dot{P}_{xx}(t)$ equation cannot be integrated independently due to this coupling, though doing so can provide a reasonable approximation for some systems. Additionally, the dependence on the $P_{xc}(t)$

terms requires that the measurement updates of the $P_{xc}(t)$ be included. The SKF measurement update equations, as described by Woodbury [5], reveal how the estimated state covariance update is dependent on the $P_{xc}(t)$ term (which is also updated):

$$\begin{aligned}
\bar{y} &= H_x\bar{x} + H_c\hat{c} \\
P_{xy} &= P^-_{xx}H_x^T + P^-_{xc}H_c^T \\
P_{yy} &= H_xP^-_{xx}H_x^T + H_xP^-_{xc}H_c^T + H_cP^-_{cx}H_x^T + H_cP^-_{cc}H_c^T + R \\
K &= P_{xy}P_{yy}^{-1} \\
P^+_{xx} &= (I - KH_x)\,P^-_{xx} - KH_cP^-_{cx} \\
P^+_{xc} &= (I - KH_x)\,P^-_{xc} - KH_cP_{cc}
\end{aligned} \tag{17}$$

where H_x and H_c are the measurement partials with respect to the state and consider parameters, R is the measurement noise covariance, and K is the Kalman gain. Note that when the measurement model does not depend on the considered parameters, as is the case when considering gravity terms and instantaneous optical measurements, H_c is zero. When performing these updates as part of the precomputation of the PNC, the state is not updated (as there are no measurements) and the reference trajectory is used to determine expected measurements at each time (as is done in standard covariance analysis). The covariance update expressions in equation 17 can also be formulated using the Joseph form [1], which is more numerically stable and holds true when the Kalman gain K is not optimal. This form can be derived by segmenting the known Joseph covariance update formulation $P^+ = (I-KH)P^-(I-KH)^T + KRK^T$ into state and consider portions for each of the terms. The resulting P^+_{xx} term is

$$\begin{aligned}
P^+_{xx} =& (I - KH_x)\,P^-_{xx}\,(I - KH_x)^T - KH_cP^-_{cx} - P^-_{xc}H_c^TK^T \\
&+ K\left(H_cP^-_{cx}H_x^T + H_xP^-_{xc}H_c^T + H_cP^-_{cc}H_c^T + R\right)K^T
\end{aligned} \tag{18}$$

where K is the portion of the Kalman gain matrix associated with the state values (as is the case in equation 17). The P^+_{xc} term remains the same as in equation 17 after cancellation of terms. The Joseph formulation of the covariance update with consider parameters is also described in Zanetti and DeMars [7]. The original covariance update formulation shown in equation 17 is implemented in all analysis to maintain consistency with Woodbury [5]. The non-Joseph form of the covariance update has been used in onboard filters on such vehicles as the retired Space Shuttle [15].

The final PNC profile is obtained by taking the difference in the prefit P_{xx}^{-} terms between the standard KF and the SKF at each measurement time (as described in equation 15), which are computed by performing covariance propagation and updates along a reference trajectory. This difference provides the full PNC contribution of both the $P_{cc}(t_k, t_0)$ and $P_{cx0}(t_k, t_0) + P_{cx0}^{T}(t_k, t_0)$ terms. Note that the KF and SKF covariance analyses are performed simultaneously, and the computed PNC after each propagation is added to the KF propagated covariance before performing the KF covariance update (keeping the KF covariance and state portion of SKF covariance equal before performing the covariance updates).

As a result, the filter performance is approximately equal (and exactly equal for linear systems) to the SKF, without the need to compute onboard the effect of the consider parameter uncertainty on the state uncertainty. The overall algorithm is as follows (and is depicted in Figure 1):

- Input Arguments: initial state x_0, initial state covariance P_{xx0}, nominal consider parameters c, initial state-consider cross covariance P_{xc0} (typically set to zero), consider parameter covariance P_{cc}, measurement times, and expected measurement model
- Covariance Analysis (propagation and update of the formal covariance only) along reference trajectory is performed along reference trajectory: Both Kalman Filter (with no consider parameters in state) and Schmidt-Kalman Filter (with consider parameters included) covariance analyses performed simultaneously
- The covariance is propagated forward in time via the KF and SKF formulations
- The covariance associated with the estimated state is extracted from both propagation results
- The propagated covariance from the KF covariance analysis is subtracted from the SKF propagated covariance (producing a positive definite matrix because the SKF propagated covariance includes consider parameter uncertainties)
- This covariance difference is saved to form a single entry of the precomputed PNC profile

- The covariance difference is added to the KF propagated covariance, in order to keep the two covariance analyses in sync (alternatively the SKF propagated covariance can be used instead of the KF propagated covariance plus the covariance difference)
- The covariance analysis is run forward for all measurement times, to produce the complete precomputed process noise profile (which is employed in the onboard filter)

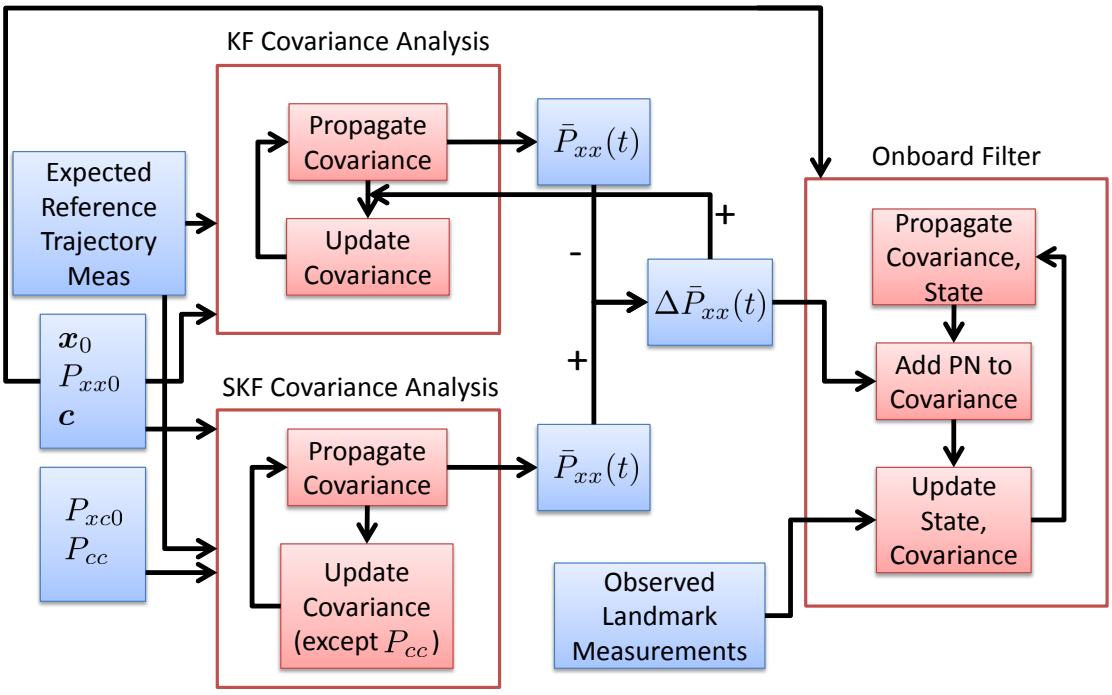


Fig. 1: Process Noise Covariance Algorithm

Note how the precomputed PNC is dependent not only on the model parameter uncertainty, but also the initial state uncertainty (due to the measurement updates of the cross-covariance terms), the measurement model and nominal expected measurements along the reference trajectory, and the time between measurements.

A. The Additive Divided Difference Sigma Point Schmidt-Kalman Filter

While the PNC profile can be precomputed with a linearized approach using the EKF and ESKF, as shown in equations 16 and 17, it is also possible to accomplish this precomputation using sigma point transformations. Additionally, the use of sigma point transformations allows

the separate computation of the $P_{cc}(t_k, t_0)$ term from equation 15, which can be valuable if it sufficiently captures the PNC needed (eliminating the need to perform the more computationally expensive covariance analysis described in Figure 1). In this study, a sigma point formulation called the Additive Divided Difference Sigma Point Filter (ADF) is used to precompute the PNC profile [16–18]. The derivation of the Consider Filter form of the ADF, the ADSF, is provided here.

The Divided Difference Sigma Point Transformation (DDT) is the heart of the ADF, and is what makes the ADF unique from other sigma point filters. In linearized filters it is common to approximate a nonlinear function $f(x)$ with a Taylor series truncation, such as

$$f(x) \approx f(\hat{x}) + f'(x)(x - \hat{x}) \tag{19}$$

where $f'(x)$ is an exact gradient. By contrast, the divided difference transformation uses a second-order truncation along with numerical differencing formulas for the derivatives:

$$f(x) \approx f(\hat{x}) + \tilde{D}^{(1)}_{\Delta x} f(\hat{x}) + \tilde{D}^{(2)}_{\Delta x} f(\hat{x}) \tag{20}$$

where the divided difference operators $\tilde{D}^{(i)}_{\Delta x} f(\hat{x})$ approximate the coefficients of the multidimensional Taylor series expansion using Stirling interpolations. These interpolators difference perturbations of $f(\hat{x})$ across an interval, h, over a spanning basis set of sigma points, which are formed from the columns of the Cholesky decomposition of the covariance of the domain vector, x. For additional details, refer to Nørgaard, et al [16, 17].

Let $\hat{\mathcal{X}}$ denote the array whose columns are a particular ordering of the sigma points derived from the domain mean $\hat{x}$ and its corresponding covariance, P_x. Then

$$\hat{\mathcal{X}} = \left[\hat{x},\, \hat{x} + h\sqrt[c]{P_x}(:,1),\, \hat{x} + h\sqrt[c]{P_x}(:,2), ..., \hat{x} - h\sqrt[c]{P_x}(:,1), \hat{x} - h\sqrt[c]{P_x}(:,2), ...\right] \tag{21}$$

where the index $(:, i)$ denotes column i of $\sqrt[c]{P_x}$, and $P = \sqrt[c]{P}\sqrt[c]{P}^T$ denotes a lower-triangular Cholesky factorization. There are a total of $2n + 1$ sigma point columns, where n is the dimension of the state x. Let sigma points in the range of the function be given by

$$\mathcal{Y} = f(\mathcal{X}) \tag{22}$$

where each column of $\mathcal{Y}$ corresponds to the mapped column of $\mathcal{X}$. These range sigma points may be merged to form the mean of the range of the function using

$$\hat{y} = \mu_h(\mathcal{Y}) = \frac{h^2 - n}{h^2}\mathcal{Y}_{(:,1)} + \frac{1}{2h^2}\sum_{i=2}^{2n+1}\mathcal{Y}_{(:,i)} \tag{23}$$

where the subscript $(:, i)$ denotes column i of $\mathcal{Y}$. To form an associated covariance, the following divided-differences are next computed for values of i from 1 to n:

$$\begin{aligned}\tilde{D}^{(1)}_{\Delta x} f(\hat{x})_{(:,i)} &= \frac{1}{2h}\left[\mathcal{Y}_{(:,i+1)} - \mathcal{Y}_{(:,i+1+n)}\right] \\ \tilde{D}^{(2)}_{\Delta x} f(\hat{x})_{(:,i)} &= \frac{\sqrt{h^2-1}}{2h^2}\left[\mathcal{Y}_{(:,i+1)} + \mathcal{Y}_{(:,i+1+n)} - 2\mathcal{Y}_{(:,1)}\right]\end{aligned} \tag{24}$$

The covariance is then computed from

$$P_y = \left[\tilde{D}^{(1)}_{\Delta x} f(\hat{x}),\, \tilde{D}^{(2)}_{\Delta x} f(\hat{x})\right]\left[\tilde{D}^{(1)}_{\Delta x} f(\hat{x}),\, \tilde{D}^{(2)}_{\Delta x} f(\hat{x})\right]^T \tag{25}$$

The DDT is applied to the nonlinear dynamics function to compute the propagated mean state $\bar{x}$ and covariance P^-_{xx}, and then to the nonlinear measurement function to compute the expected measurement $\bar{y}$ and associated innovation covariance matrix P_{yy} (along with the measurement uncertainty matrix R that is added to the mapped covariance). We employ the standard approach of using augmented state and covariance matrices for these transformations to incorporate traditional process and measurement noise [18, 19]. The tuning parameter h is typically set to the value of $\sqrt{3}$, but can be modified for each of the two sigma point transformations. The cross correlation matrix is then obtained by using the Cholesky decomposition of the mapped state covariance and the forward difference term from the measurement function DDT:

$$P_{xy} = \sqrt[c]{P^-_{xx}}\left[\tilde{D}^{(1)}_{\Delta x} f(\hat{x})_{(:,1:n_x)}\right]^T \tag{26}$$

where n_x is the final index value of the state portion of the augmented state vector. The filter gain matrix K is the same as in equation 17, and the state update has the same form as in the KF. The updated covariance can be computed via either the traditional formulation

$$P^+_{xx} = P^-_{xx} - KP_{yy}K^T, \tag{27}$$

or through the direct use of the difference values as

$$P^+_{xx} = \left[\sqrt[c]{P^-_{xx}} - K\mathcal{S}^{vv}_x \quad K\mathcal{S}^{vv}_{xv}\right]\left[\sqrt[c]{P^-_{xx}} - K\mathcal{S}^{vv}_x \quad K\mathcal{S}^{vv}_{xv}\right]^T \tag{28}$$

where

$$\begin{aligned}\mathcal{S}_x^{vv} &= \tilde{D}_{\Delta x}^{(1)} f(\hat{x})_{(:,1:n_x)} \\ \mathcal{S}_{xv}^{vv} &= \left[\tilde{D}_{\Delta x}^{(1)} f(\hat{x})_{(:,n_x+1)}, ..., \tilde{D}_{\Delta x}^{(1)} f(\hat{x})_{(:,n_{xv})}, \tilde{D}_{\Delta x}^{(2)} f(\hat{x})_{(:,n_{xv})}\right]\end{aligned} \tag{29}$$

where n_{xv} is the length of the augmented state vector for the measurement update (i.e. the number of states plus the number of measurement values).

Reformulation of the ADF into the ADSF consider filter starts with the augmentation of the estimated state parameters with the consider parameters into a single vector. The state and covariance propagation formulation remains the same, employing the DDT described above for the augmented vector. For each sigma point used in the propagation step, the consider parameters are used in the propagation of the state parameters and the consider parameters remain constant (which includes the deterministic deviations of the sigma points). For the state and covariance update steps, the update of the consider parameters is set to zero, and the portion of the covariance directly corresponding to the consider parameters is not updated (though the state and cross-covariance terms are updated) [6]. The augmented covariance update is shown by

$$P_{zz}^{+} = \begin{bmatrix} P_{xx}^{-} & P_{xc}^{-} \\ P_{cx}^{-} & P_{cc} \end{bmatrix} - \begin{bmatrix} K_x P_{yy} K_x^T & K_x P_{yy} K_c^T \\ K_c P_{yy} K_x^T & 0 \end{bmatrix} \tag{30}$$

where $K = \begin{bmatrix} K_x \\ K_c \end{bmatrix}$, or using the difference method as described in equation 28 and setting the P_{cc} update to zero by reseting it to the original value. The difference in state covariance matrices for the ADF and ADSF are computed to generate the PNC profile that is used in the onboard filter, as described in Figure 1.

As noted at the beginning of this section, the DDT described here can also be used to intuitively map the uncertainty in the consider parameters into the space space. The resulting covariance is the $P_{cc}(t_k, t_0)$ term from equation 15. If the $P_{cc}(t_k, t_0)$ term computed for each propagation interval between measurements is used to generate the PNC profile, and the resulting filter performance meets requirements, the more computationally expensive covariance analysis described in Figure 1 may not be required. However, since the PNC profile is computed in advance, and likely by ground-

based resources, the additional computation required by the algorithm shown Figure 1 is not likely to prove burdensome.

IV. Linear Problem: Falling Object Scenario

A simple, linear estimation problem is used throughout the literature for demonstration of consider covariance analysis and filtering techniques, and proves effective in showing the utility of the new PNC computation method. Additionally it is used to verify correct implementation of the algorithms. The problem is a one dimensional falling object scenario, with an uncertain gravity model value. This problem is employed in Stauch [6], Tapley [1], Lisano [4], and Woodbury [5]. The point mass in freefall is shown in Figure 2. The state consists of the position x and velocity $\dot{x}$, while the gravitational parameter g is a consider parameter. The position range y is directly measured.

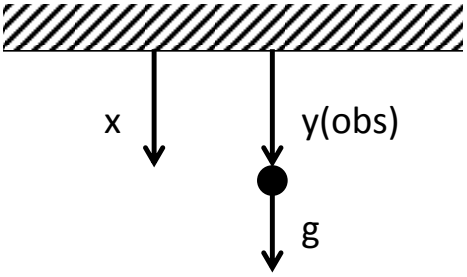


Fig. 2: Free Falling Point Mass

A. Simulation Scenario

The nominal initial conditions are

$$x_{nom}(t_0) = 1\ m, \quad \dot{x}_{nom}(t_0) = 0\ m/s, \quad g_{nom} = 10\ m/s^2 \tag{31}$$

while the truth initial conditions are

$$x_{tru}(t_0) = 0.8\ m, \quad \dot{x}_{tru}(t_0) = 0.3\ m/s, \quad g_{tru} = 9.8\ m/s^2 \tag{32}$$

The initial state covariance, state-consider cross-covariance, consider parameter g uncertainty, and measurement uncertainty are

$$P_{xx0} = \begin{bmatrix} 1\ m^2 & 0 \\ 0 & 1\ (m/s)^2 \end{bmatrix}, \quad P_{xc0} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad P_{cc} = \left[1\ \left(m/s^2\right)^2\right], \quad R = [1\ pixels] \tag{33}$$

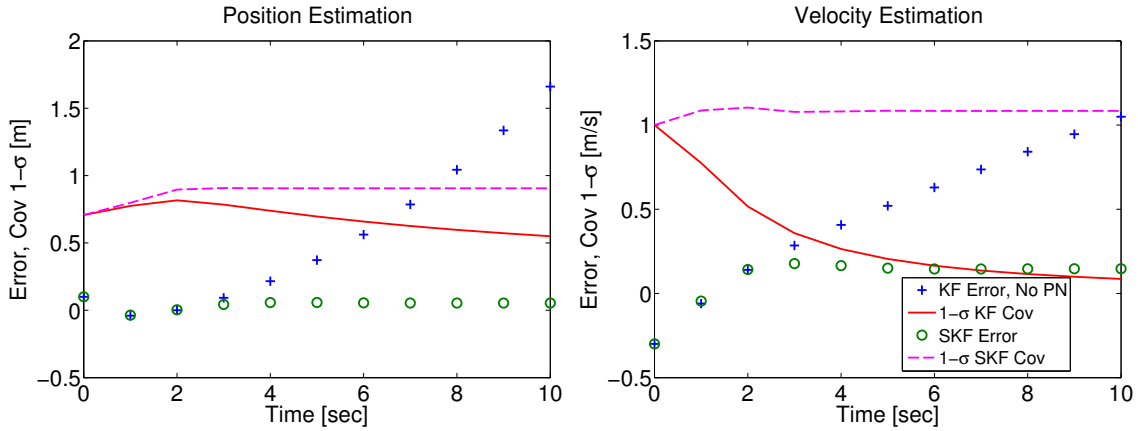
Note that the initial error for this single simulation is well within the initial state and consider parameter uncertainty distributions. The governing equations of motion are

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \end{bmatrix} = \begin{bmatrix} \dot{x} \\ g \end{bmatrix} \tag{34}$$

The position range y is directly measured every second from 0 to 10 seconds, and the measurement error is set to zero for this simple linear problem (though importantly the measurement uncertainty is non-zero for each measurement). The PNC for this linear problem is computed as described in the "Method of Solution" section, taking the difference between the prefit KF state covariance and prefit SKF state covariance at each measurement time. The precomputed PNC is added to the state covariance at the end of each propagation interval between measurements.

B. Results

First the linear scenario is run with both the KF and SKF, with no PNC added to the KF after each propagation of the state uncertainty. The estimated states for the KF and SKF are the position and velocity, and the considered parameter in the SKF is the gravity term. The resulting position and velocity errors and 1-σ covariance values are shown in Figure 3. These results are identical to those shown in Stauch [6]. The KF without PNC added does not account at all for the

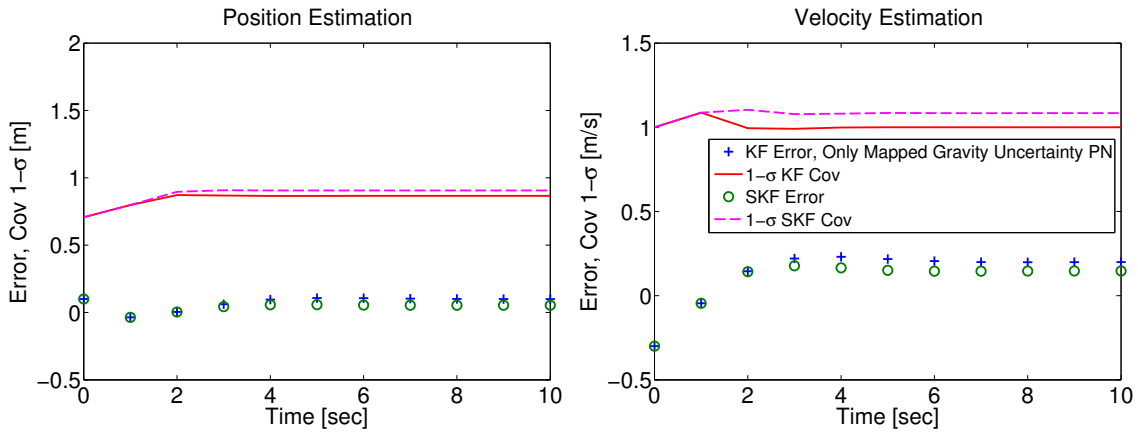


(a) Position **(b) Velocity**

Fig. 3: SKF versus KF without Process Noise Covariance

uncertainty in the gravity value, and as a result the estimated covariance artificially converges and the state error diverges.

To account for the uncertainty in the gravity term, PNC must be added. To start, the consider state (gravity) uncertainty is mapped into the state space over each propagation interval using the DDT, as described in section III A. This mapping provides the $P_{cc}(t)$ term as shown in equation 14. Using only this term as the PNC, the KF performance significantly improves, as shown in Figure 4. Most of the performance gain of the SKF over the KF is achieved by adding this $P_{cc}(t)$ PNC term.

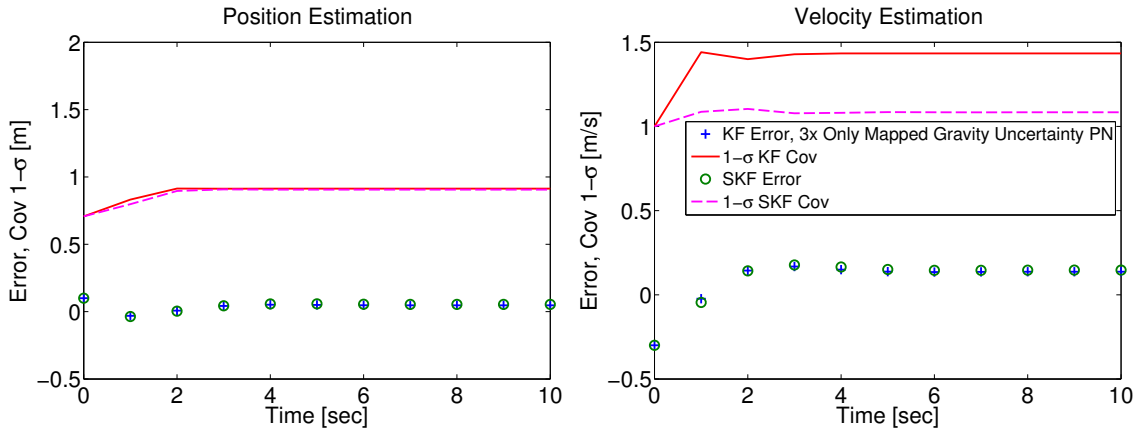


(a) Position **(b) Velocity**

Fig. 4: SKF versus KF with Only Mapped Gravity Uncertainty Process Noise Covariance

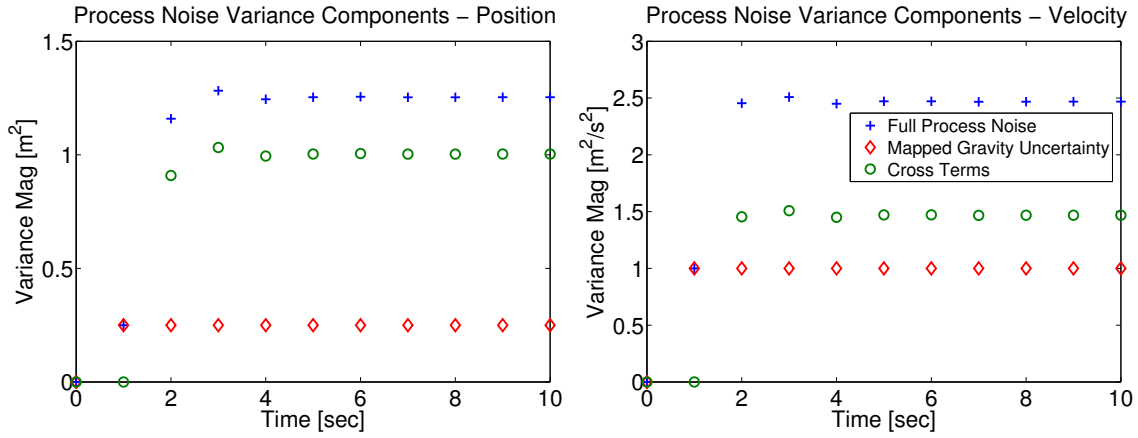
However, the KF performance does not fully match the performance of the SKF, with slightly lower uncertainty values and slightly higher error values. In an attempt to achieve better performance with the KF, the PNC values $P_{cc}(t)$ derived from the gravity uncertainty mapping are multiplied by a factor of three. This simple multiplication of the covariance is equivalent to how standard filter tuning operates (except the standard PNC doesn't have the $P_{cc}(t)$ starting point). The result is shown in Figure 5. A factor of three was chosen (by trial and error) to obtain KF errors on par with the SKF, but as a result the velocity uncertainty is greatly inflated.

To understand why the KF doesn't match the SKF, and why no amount of modifying the $P_{cc}(t)$ term will produce matching results, is it helpful to plot the magnitude of the $P_{cc}(t)$ variance values versus the $2P_{cx0}(t)$ term variance values and the full process noise $P(t)$ variance values as described in equation 14. This plot is shown in Figure 6. Because of the linear nature of the problem, the $P_{cc}(t)$



(a) Position **(b) Velocity**

Fig. 5: SKF versus KF with 3x Only Mapped Gravity Uncertainty Process Noise Covariance

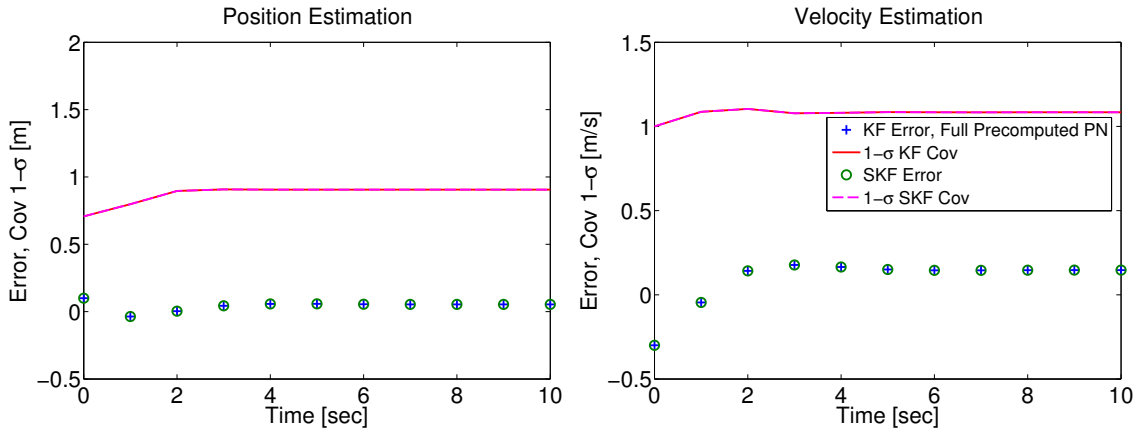


(a) Position **(b) Velocity**

Fig. 6: Process Noise Component Variance Magnitudes (1-σ)^2

contribution is identical for each segment, but the $2P_{cx0}(t)$ cross contributions change significantly and are greater in magnitude than the $P_{cc}(t)$ contribution starting at the time of two seconds. Note that the variance values are plotted instead of the 1-σ standard deviations values to show how the $P_{cc}(t)$ and $2P_{cx0}(t)$ cross contribution values added together produce the full process noise variance values. If the full PNC is used in the KF, the resulting performance matches exactly the SKF, while using the computationally much less expensive KF, as shown in Figure 7.

Note that identical precomputed PNC values and overall simulation results are achieved using three different filter forms: the EKF and ESKF, the UKF and USKF, and the ADF and ADSF.



(a) Position **(b) Velocity**

Fig. 7: SKF versus KF with Full Precomputed Process Noise Covariance

Achieving identical results for these different non-linear filters in a linear problem provides a strong sanity check for the algorithm implementation.

V. Nonlinear Problem: Asteroid Descent Scenario

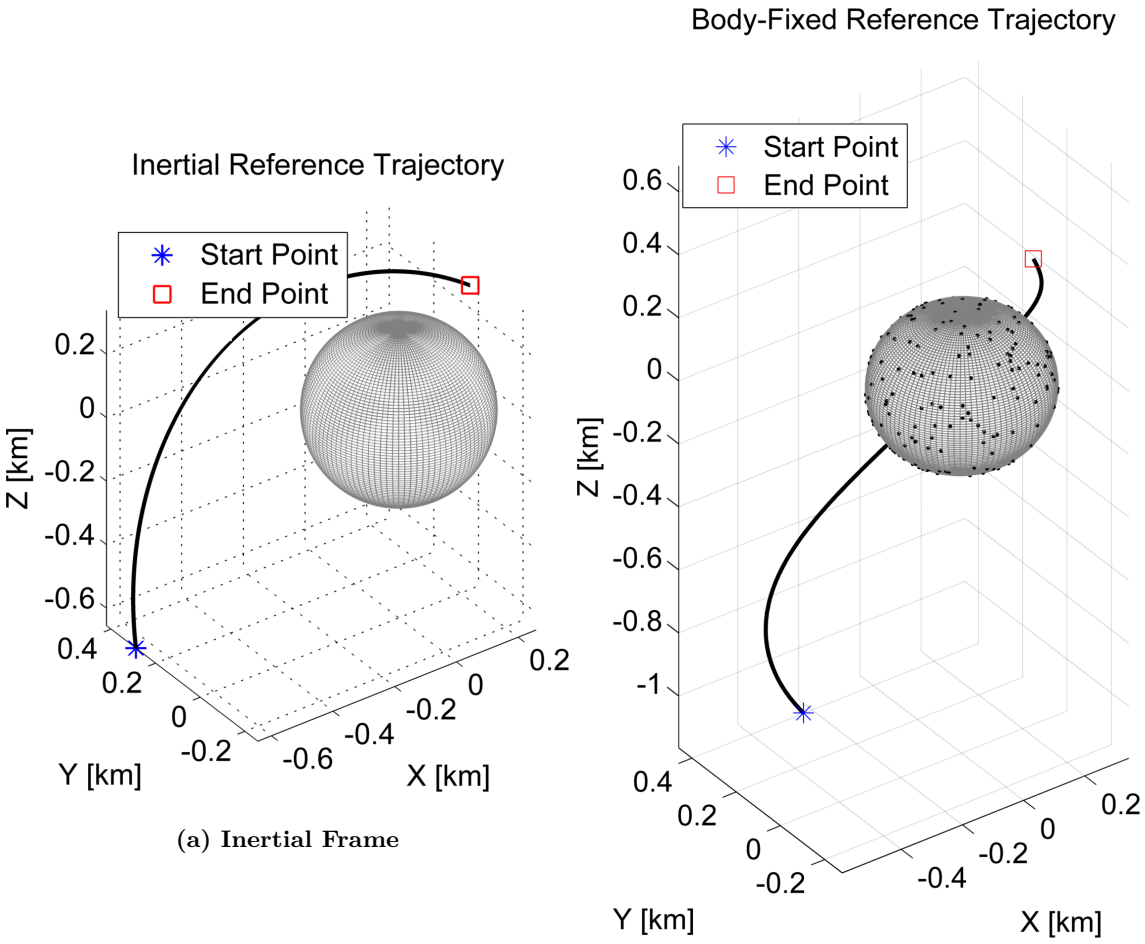
Precomputed PNC is next generated for a nonlinear estimation problem: a spacecraft descent scenario to a checkpoint above the surface of an asteroid, using surface landmarks for optical navigation. The descent scenario simulated matches the currently planned descent trajectory of NASA's OSIRIS-REx asteroid sample return mission during the sample collection phase of the mission, ending at a predetermined checkpoint above the surface. The simulated camera also matches the specifications of the NavCam camera intended for optical navigation during the descent. There are several challenges in this scenario which make use of the new PNC precomputation method:

- Onboard computational limits
- Prior to the descent trajectory, only limited gravity field estimates are available
- The magnitude of the central body force and gravity harmonics perturbations increase significantly during the descent trajectory, which exacerbate dynamical errors resulting from lower order gravity term errors and truncation of higher order gravity terms

A. Simulation Scenario

The reference descent trajectory employed in this simulation is shown in the inertial and body-fixed frames in Figure 8. Note that the object plotted in the Figure 8a is the Brillouin Sphere (i.e. a sphere with radius equal to the largest radius of the asteroid), while the object in Figure 8b is a triaxial ellipsoid with minimum, median, and maximum radii corresponding to the current best estimated dimensions of the asteroid Bennu (the intended target of the OSIRIS-REx mission). The orientation of the body-fixed frame with respect to the inertial frame is provided in Table 2. Figure 8b also shows a portion of the 300 predetermined and randomly located surface landmarks assumed known for this simulation. Remaining above the Brillouin Sphere also allows spherical harmonics to be used for the entire trajectory.

The simulation setup, including timing and initial spacecraft states, is described in Table 1. Note that for the onboard optical navigation, photos are taken every two minutes during the descent trajectory. The descent trajectory is also chosen such that lit landmarks will be visible for the entirety of the trajectory.



(a) Inertial Frame

(b) Body-Fixed Frame

Fig. 8: Descent Trajectory

Table 1: Simulation Setup

Simulation Parameter	Values
Epoch Time	11-Oct-2019 12:17:04.953
End Time	11-Oct-2019 16:17:04.953
Minimum Time Before First Update	1 minute after epoch
Time Between Photos	2 minutes
Spacecraft Initial Inertial Position	[-0.644, 0.288, -0.655] km
Spacecraft Initial Inertial Velocity	[1.182x10^{-5}, 3.009x10^{-5}, 6.368x10^{-5}] km/s

The properties of the small body used for the simulations are given in Table 2. Note that the triaxial ellipsoid from Figure 8b is used as the shape model in the simulations. While a higher fidelity polyhedron shape model is available, it is not necessary to evaluate the new PNC algorithm. The

Table 2: Asteroid Properties

Simulation Parameter	**Nominal Values**
Initial Pole Right Ascension, Declination	0, 10 deg
Pole Right Ascension, Declination Rates	2, 3 deg/Julian century
Longitude of the Prime Meridian at Epoch	0 deg
Rotation Rate	2010.489449 deg/day
GM	$5.2\text{x}10^{-9}$ km^3/s^2
GM Uncertainty (1- σ)	$5.2\text{x}10^{-11}$ km^3/s^2 (1%)
Radius Values	$0.259 \times 0.250 \times 0.230$ km
Number of Landmarks on surface	300
Nominal Gravity Harmonics	4 x 4 (Bennu Estimate) [20]
Truth Gravity Harmonics	12 x 12 (Bennu, Perturbed)
Gravity Harmonic Uncertainties	Modified Kaula Rule [20]

asteroid GM and nominal spherical harmonic coefficients for Bennu are provided in McMahon [20]. McMahon also provides a Modified Kaula Rule that has been derived for Bennu, which provides realistic maximum uncertainties for the gravity spherical harmonic coefficients. The upper bound expressions of both the zonal coefficients ($0.183/n^2$) and non-zonal coefficients ($0.043/n^2$) are used as the uncertainties in the harmonic coefficients up to degree and order 12. Note that the uncertainties for the GM and lower order terms are likely to be significantly lower from estimation of the terms in the mapping orbits of the preceding mission phases, but a more strenuous scenario is sought to evaluate the new precomputed PNC method.

B. Dynamical Model

The spacecraft position and velocity are propagated via numerical integration between picture times using fixed step RK8 propagation. The covariance is numerically integrated to the same

times, also using fixed step RK8 propagation. The simulation includes central body acceleration and perturbations from a spherical harmonic gravity field, for both the truth and filter dynamics. Standard perturbations such as third body perturbations (TBP) and solar radiation pressure (SRP) are not included for simplicity, as the focus of this analysis is to evaluate the new PNC algorithm in the context of gravity field uncertainty. These SRP and TBP terms also do not change significantly in magnitude over the descent trajectory, unlike the direct gravity forces and perturbations.

A 4x4 degree and order spherical harmonic gravity model of the Bennu asteroid that is employed for mission design analysis of the OSIRIS-REx mission is used for the nominal onboard dynamical model. A higher order 12x12 gravity field is used for the truth dynamical model in order to emulate the higher order terms of a true body that are truncated in the onboard model. These higher order terms introduce unmodeled perturbations in the trajectory, and it is desired to know if the filter performs sufficiently well when experiencing these perturbations. An analysis investigating the effect of this truncation on the filter performance for different levels of nominal and truth gravity fields is saved for future work.

To obtain a higher order 12x12 gravity field for each of the truth trajectories, first the uncertainties for the 12x12 field are sampled to generate errors for each of the 12x12 field coefficients. For the lower order 4x4 field values, these errors are added to the non-zero nominal 4x4 field values (which are expected to be estimated in the prior mapping orbit) to obtain the truth 4x4 coefficients. For the higher order terms up to 12x12, these errors are added to the zero value (i.e. truncated) nominal coefficients: the error values become the coefficient values. As described in the Simulation Scenario section above, these coefficient uncertainties are obtained from the Modified Kaula Rule [20], which provides maximum deviations from zero of the spherical harmonic coefficients. Thus using these uncertainties for the higher order terms allows us to account for the "errors of omission" introduced by the truncation of the gravity field.

C. Measurement Model

The navigation simulation uses optical line-of-sight direction vector measurements from the spacecraft to the landmarks on the surface of the small body, whose locations are assumed to be

known. Landmarks are defined as the central positions of surface maplets computed using the stereophotoclinometry method, as described in Gaskell [21], which have numerous advantages over the use of natural surface features such as craters. It is assumed that the spacecraft is commanded to point its camera at the center of mass of Bennu using onboard position and attitude knowledge at every picture time using reaction wheels or thrusters. A similar simulation can be performed using a predetermined inertial attitude profile.

The navigation camera parameters are described in Table 3. These parameters match the NavCam camera planned for navigation use in the OSIRIS-REx mission. Camera distortion effects that are typically calibrated in flight are not included in these simulations, nor are the remaining errors after calibration at the FOV edges that are more likely for a wide FOV lens. The landmarks observation model is provided in Bhaskaran [22].

Table 3: Navigation Camera Properties

Simulation Parameter	Nominal Values
Focal Length	7.68 mm
Sensor Array	2592×1944 pixels
Pixel Density	454.54 pixels/mm
Field of View (FOV)	$40.74^\circ \times 31.12^\circ$

D. Estimation Filter Architecture

A sequential EKF is employed to evaluate the effectiveness of the new PNC model. The estimated states are the spacecraft position and velocity. It is assumed that the small body orientation, rotational velocity, shape model, gravity field, and set of surface landmarks listed in Table 2 have been previously estimated. These quantities are held as constants in the "onboard" filter. A summary of onboard filter parameters is shown in Table 4 [23, 24].

The spacecraft attitude is not estimated, nor are knowledge attitude errors added for each photo, because it is assumed that a highly accurate separate attitude determination system (ADS) consisting of gyroscopes, star cameras, and an attitude determination filter will provide attitude estimates

Table 4: Onboard Filter Parameters

Simulation Parameter	Nominal Values
Initial Filter Position Covariance (Radial, In-Track, Cross-Track) (1-σ)	[0.012, 0.053, 0.004] km
Initial Filter Velocity Covariance (Radial, In-Track, Cross-Track) (1-σ)	[3.92x10^{-6}, 5.06x10^{-7}, 3.10x10^{-8}] km/s
Measurement Noise Covariance (1-σ)	[29.5, 29.5] arcsec ([0.5, 0.5] pixels)
Traditional Process Noise q for Position and Velocity	5x10^{-16} km^2/s^3

at each photo time. Should ADS knowledge attitude errors prove significant (as is predicted for the actual OSIRIS-REx mission due to expected thermal variations in the spacecraft structure), it is possible to further correct the attitude using landmark observations [18]. It is also not the focus of this analysis to investigate spacecraft attitude estimation performance, but rather how the new PNC algorithm affects spacecraft position and velocity estimation performance.

E. Monte Carlo Simulation Design

Monte Carlo analyses are employed to compare the standard and precomputed PNC techniques. A single estimated state and covariance is assumed known at the start of the descent trajectory. Also assumed known are the nominal estimated coefficients for a 4x4 gravity field, as well as a covariance matrix that includes uncertainties up to degree and order 12.

The errors for the initial state, GM, and gravity field coefficients are sampled assuming a Gaussian distribution with the 1-σ values provided in Table 5. Note that modern image processing capabilities can generate landmark center values at sub-pixel precision [22, 25] (depending on the relative resolution of the landmark maplet data and the image pixel scale). Thus 0.5 pixel errors, equivalent to 29.5 arc seconds angular error for this camera, are used in this simulation. In higher fidelity simulations a resolution-dependent term is used for filter measurement noise value, which is neglected in this example for simplicity.

The initial state errors are used to distribute the initial truth states about the nominal. As a result, a different truth trajectory and set of observations exists for each Monte Carlo trial. A small body mission will have a single estimated nominal state at the beginning of the descent trajectory,

Table 5: 1-σ Monte Carlo Error Parameters

Simulation Parameter	1-σ Values
Spacecraft Initial Position (Radial, In-Track, Cross-Track)	[0.012, 0.053, 0.004] km
Spacecraft Initial Velocity (Radial, In-Track, Cross-Track)	[3.92x10^{-6}, 5.06x10^{-7}, 3.10x10^{-8}] km/s
Observation	[29.5, 29.5] arcsec ([0.5, 0.5] pixels)
GM	5.2x10^{-11} km^3/s^2 (1% of nominal)
Asteroid Gravity Coefficients	Modified Kaula Rule [20]

which will be used as the reference trajectory for precomputing the PNC. The truth state will not be known, but the Monte Carlo analysis will reveal the performance of the filter with the precomputed PNC method for many different possible truth trajectories. The same logic extends to the asteroid gravity coefficients: only estimates will be known, about which the precomputation of the PNC will occur. As a result, the landmark measurements are at least slightly different for each Monte Carlo trial. The variation in the landmarks detected and used is valuable as an evaluation of the filter performance using precomputed PNC.

F. Application of Precomputed Process Noise Covariance Method

For this scenario, the model parameters of interest that are used to generate precomputed PNC are the GM and the gravitational spherical harmonic coefficients through degree and order 8. The gravity model is higher order than the 4x4 gravity field used in the "nominal" propagation function $\hat{\phi}$ in order to capture the effect of these higher order terms on the trajectory. The gravity model is lower order than the 12x12 gravity field used in the "truth" propagation function ϕ to represent the fact that no finite gravity model will capture the infinite degree and order truth gravity field.

The precomputed PNC profile is computed via either the linearized or sigma point approaches described in the "Method of Solution" section. These different approaches deliver approximately the same PNC profile in this problem. When using the DDT, the h tuning parameters described in equations 21, 23, and 24 are set equal to standard value of $\sqrt{3}$ [18].

The resulting precomputed 6x6 PNC matrices for each picture time are stored as a 6x6x121 three dimensional array (the 6x6 covariance matrix at 121 time steps), totaling 4356 double precision

numbers (34 kB). If the symmetry of the PNC matrices is accounted for, the storage required is reduced by 5/12. A table containing these values is possible to use directly onboard if fixed step integration and set measurement times are used. If greater amounts of PNC data are needed, variable step integration is employed, or photos don't occur at predictable times, the PNC sigma and correlation coefficient values can be fit to polynomials. See DeMars [8] for a detailed study of interpolation methods for precomputed process noise between measurements (which would likely need only minimal adjustment to account for measurement updates).

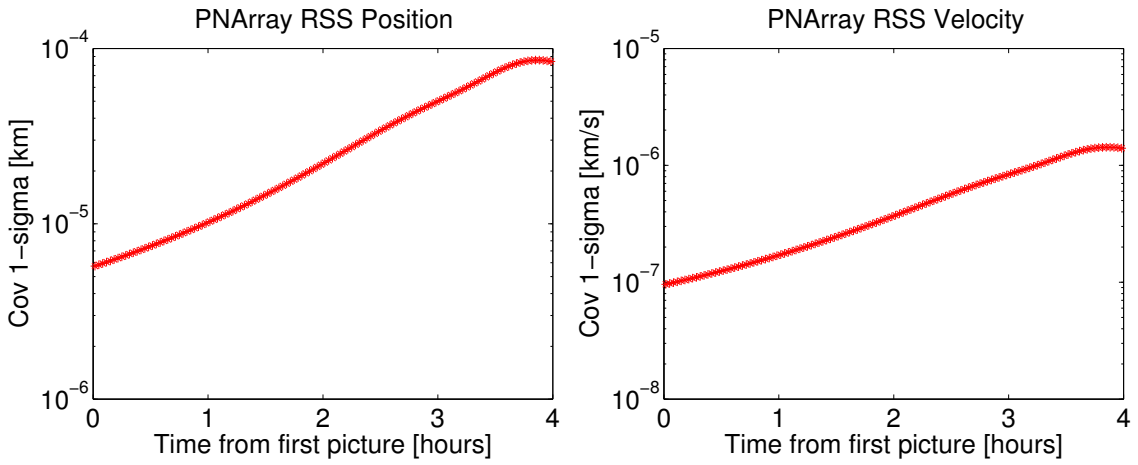
The spacecraft position and velocity covariance is propagated using both the precomputed PNC model and standard PNC model. The resulting filter performance and spacecraft state uncertainty representation is evaluated for both approaches.

G. Results

The precomputed PNC position and velocity magnitudes for each two minute interval between measurements for this scenario is plotted in Figure 9. Note how the PNC changes significantly in magnitude over the simulation timeline. The cross-covariance terms $2P_{cx0}$ are found to be many orders of magnitude smaller than the $P_{cc}(t)$ terms in this scenario (unlike for the linear estimation problem). In such cases, the required computation can be reduced by calculating only the $P_{cc}(t)$ PNC component for each interval. However, in this analysis, all terms of the PNC are precomputed and utilized.

To determine the improvement from using the precomputed PNC shown in Figure 9, a 100 trial Monte Carlo simulation is run using standard PNC as defined in equation 6 and using the q value from Table 4. The resulting errors and state uncertainty are shown for position and velocity in Figure 10. Note that the Max and RMS statistics provided are for the post-initial-convergence period, starting at the sixth picture time. Next the same Monte Carlo simulation is run using the precomputed PNC profile, with results shown in Figure 11.

The precomputed PNC profile results in a 8% reduction of the position error RMS and 21% reduction of the velocity error RMS. Additionally the velocity formal uncertainty now better matches the error distribution. The "velocity smugness" (i.e. overconfidence in the accuracy of the velocity



(a) Position **(b) Velocity**

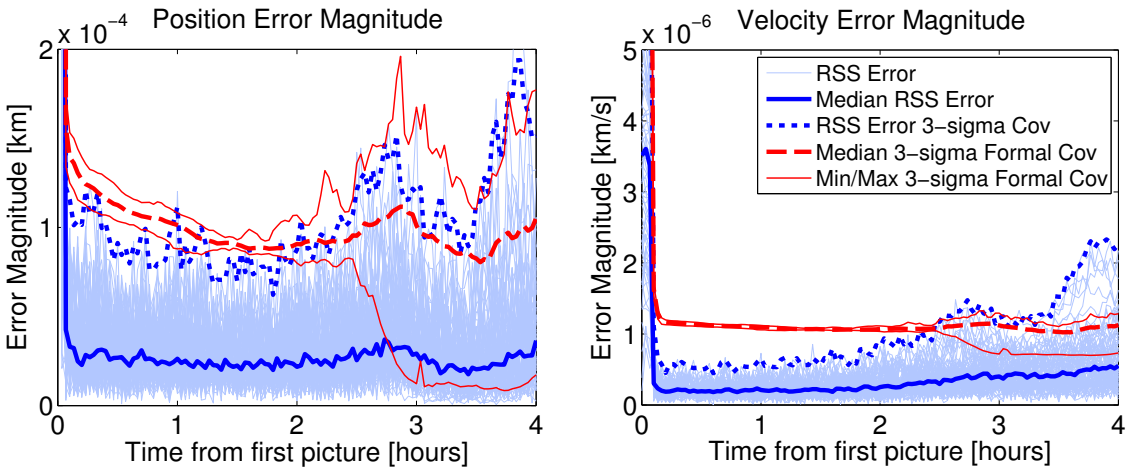
Fig. 9: Precomputed Process Noise Covariance Profile

errors evident from lower formal covariance values) has special significance for the OSIRIS-REx mission due to an abort trigger based on "time of touch" that is sensitive to velocity errors. Note that the position errors and uncertainty are less affected by the PNC chosen than the velocity errors due to the strong position information obtained from low optical measurement errors and uncertainty.

For the standard PNC method, a q value of $5\text{x}10^{-16}$ km^2/s^3 is chosen as the best balance of performance between the start and end of the simulation, an unsatisfactory compromise no longer necessary with the new PNC method. Also unnecessary is the extensive time that it takes to find such a compromise.

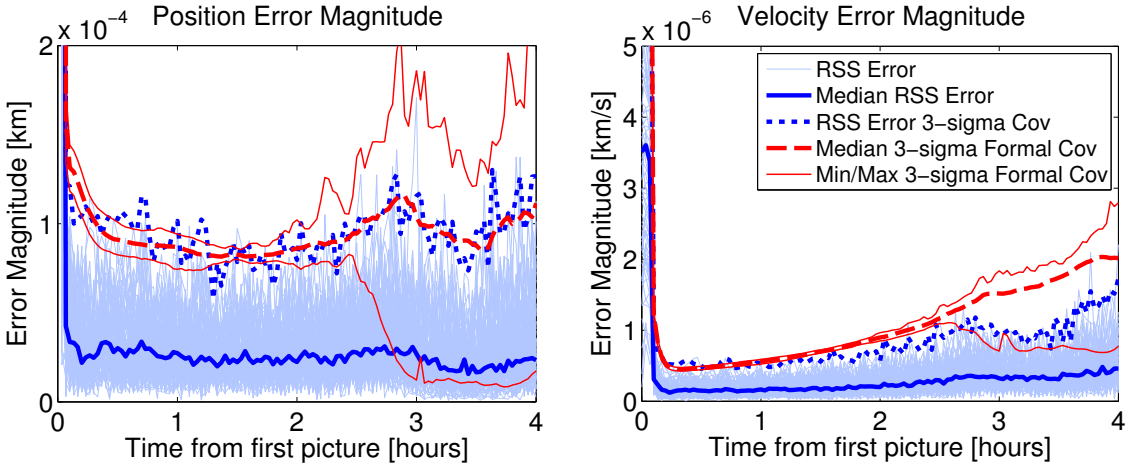
In Figures 10 and 11, there is significant disparity in the magnitude of the minimum and maximum 3-σ formal covariance values for the 100 different Monte Carlo trials. This disparity, particularly for the spacecraft position, is a result of the significant variation in the number (and distribution) of the visible landmarks between trials at the later photo times. This variation occurs because small differences in the trajectory can lead to significant differences in the specific landmarks visible at each time.

To determine if the error in each trial is consistent with the formal covariance, the position and velocity error values from each of the 100 Monte Carlo trials are divided by the 1-σ formal



(a) Position, Max 2.5E-4, RMS 3.7E-5 km **(b) Velocity, Max 2.3E-6, RMS 4.3E-7 km/s**

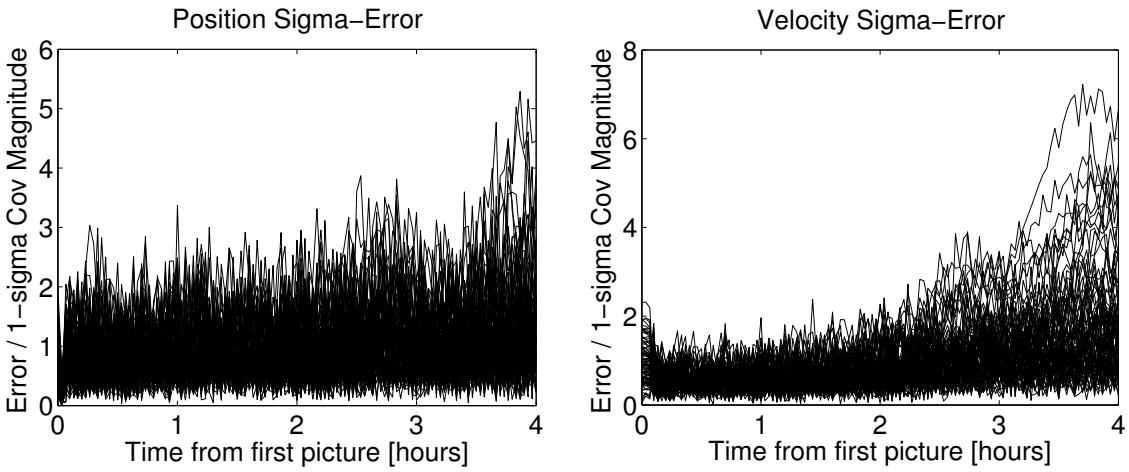
Fig. 10: Descent Scenario with Traditional Process Noise Covariance



(a) Position, Max 1.7E-4, RMS 3.4E-5 km **(b) Velocity, Max 2.2E-6, RMS 3.4E-7 km/s**

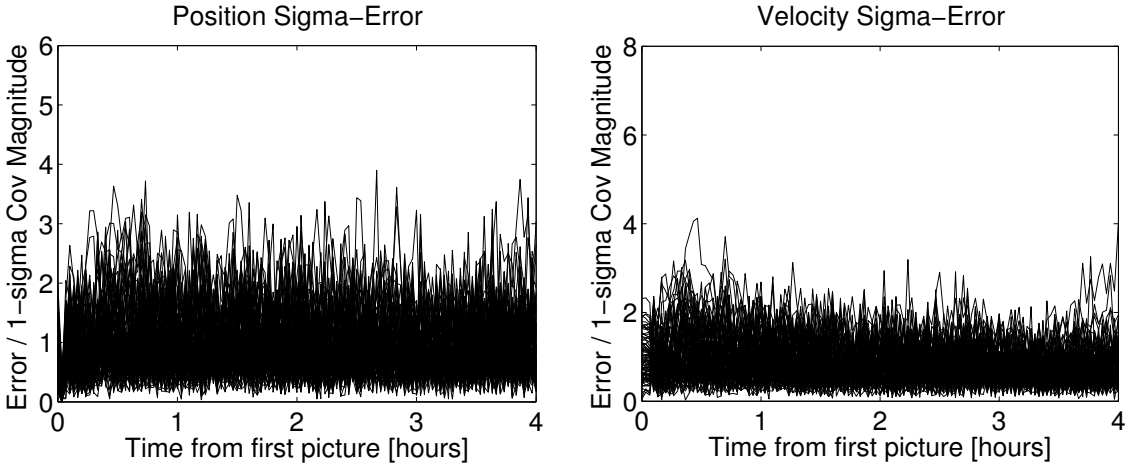
Fig. 11: Descent Scenario with Precomputed Process Noise Covariance

standard deviation values at each picture time, and the resulting values are provided in Figures 12 and 13. These plots reveal the the density of the errors relative to the covariance for the traditional and precomputed PNC scenarios. For the Monte Carlo trials that employ the traditional PNC, the error-to-formal-covariance ratio is reasonable for the position prior to hour 3, but after hour 3 the ratio becomes larger than desired (with many trials having errors that significantly exceed the 3-σ formal covariance). The corresponding ratios for the velocity are consistently very low prior to hour 2 (indicating greater than desired conservatism in the formal covariance), while after hour



(a) Position **(b) Velocity**

Fig. 12: Error Divided By 1-σ Formal Covariance, Traditional Process Noise Covariance



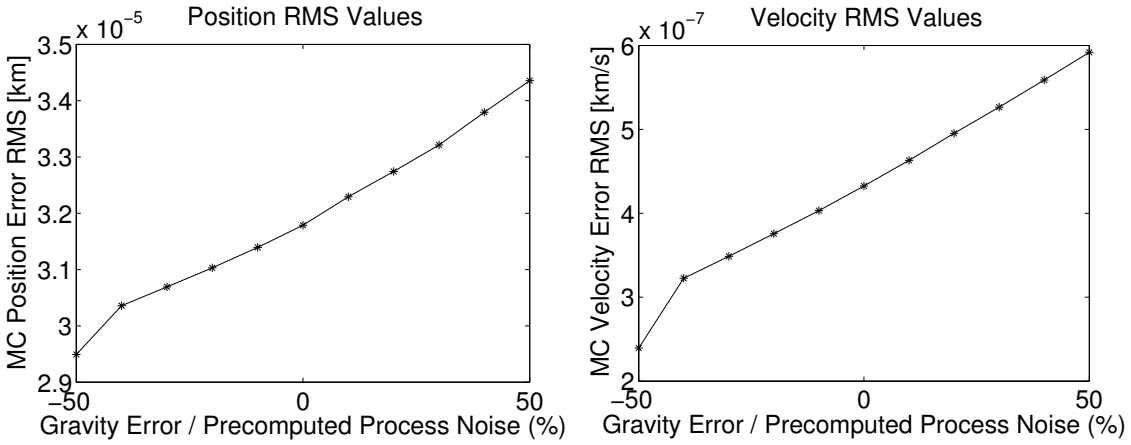
(a) Position **(b) Velocity**

Fig. 13: Error Divided By 1-σ Formal Covariance, Precomputed Process Noise Covariance

2 the ratios become much larger. For the Monte Carlo trials that employ the precomputed PNC, the error-to-formal-covariance ratio is reasonable for the position and velocity at all times, showing significantly improved consistency between the error and formal covariance over the results using the traditional PNC.

To determine how the performance of the precomputed PNC changes when the gravity coefficient errors do not match the uncertainty values used to generate the precomputed PNC, the gravity coefficient error 1-σ values are deterministically varied from -50% to +50% of the 1-σ values used

for the PNC precomputation. A 100 trial Monte Carlo simulation is performed for each each 10% interval, and the position and velocity error RMS for the post-initial-convergence period (starting at the sixth picture time) of each interval are plotted in Figure 14. Note that lower errors are obtained when the uncertainty values for the precomputed PNC are higher than the error 1-σ values. However, the resulting formal covariance is conservative relative to the error values. This result suggests that using conservative higher-than-expected gravity uncertainty values in the PNC precomputation may provide lower state errors in an actual mission (though with slightly less realistic uncertainties). A more complete sensitivity analysis of mismatching distributions of the model parameter error versus uncertainty used in the PNC precomputation, including how these different distributions affect the earlier versus later portions of the trajectory, is saved for future work.



(a) Position **(b) Velocity**

Fig. 14: Error RMS values for Gravity Error 1-σ deterministic variations

It is not the intent of this paper to provide detailed timing comparisons, but some approximate timing values can give the reader a sense of the computational savings achieved by precomputing the PNC before using it in a computationally efficient KF. All code is written in Fortran, using the latest 1995 and 2003 standards, and compiled using standard release mode settings in 2011 Intel Visual Fortran. All simulations are run on a single processor of a quad-core Intel Xeon CPU running at 3.60 GHz. The precomputation of the PNC using the DDT takes approximately 371 seconds (6.2 minutes), while a single trial of the Monte Carlo simulation takes 13 seconds to perform the onboard estimation steps (approximately 3.5% of the PNC precomputation time). To achieve

a comparable level of estimation accuracy and covariance realism without precomputing the PNC, a full SKF would be necessary to run onboard the spacecraft, which would take approximately 200 seconds instead of 13 seconds (increasing the required onboard computation by 15x). Note that for higher order gravitational fields, or if other force model parameters are considered, the onboard computational savings increase approximately with $O(N^2)$, where N is the number of state and consider parameters [26].

Note that the filter type used for the PNC precomputation and the onboard filter are interchangeable: it is entirely possible to use other filters such as the UKF or ADF onboard the spacecraft, with no requirement for the type of filter used in the PNC precomputation. The only stipulation is that the onboard filter must be able to access the additive PNC profile, in addition to the traditional tuned PNC.

H. Future Work

Future analysis involving the new precomputed PNC method includes application to other estimation problems that possess mismodeled or unmodeled perturbations, particularly for large, state-dependent perturbations. For the specific problem of spacecraft estimation, the spacecraft mass, area, and reflectivity coefficient could be included as consider parameters when precomputing the PNC, which would better account for SRP and TBP errors. Another possibility would be to apply this PNC technique to other perturbations that change with time, such as atmospheric drag. For challenging perturbations such as atmospheric drag, the proposed PNC precomputation may serve as an initial starting point that can be fine-tuned with traditional PNC values.

Another possibility for future research is the generation of functions that provide averaged PNC as a function only of the spacecraft radius magnitude, or other temporal/spatial variations. Such functions could prove useful for mission design and analysis of trajectories about small bodies, eliminating the need to recompute the PNC for each reference trajectory.

It is also possible to employ different levels of traditional PNC via a staged approach for different time intervals, or based on an exponential model of the radius of the spacecraft position (derived manually instead of the averaged PNC approach from the precomputed PNC proposed above).

Both of these approaches involve adding more manual tuning parameters to the PNC method. As a result, obtaining initial tuning parameters can be even more challenging. Thus, a future analysis comparing the multi-parameter traditional PNC model to the precomputed PNC approach is likely valuable, as is a study of how the use of the precomputed PNC approach might assist in initially tuning the multi-parameter traditional PNC model.

To go a step further in precomputing elements of the onboard filter, if the nominal trajectory and nominal measurement inclusion is fairly well known a priori, the full formal covariance and Kalman gain can be precomputed for the measurement times. Instead of storing onboard the PNC, the Kalman gain could be stored. An analysis investigating the effectiveness of this approach, both overall and in comparison to the process noise precomputation method described in this paper, is saved for future work.

Another possible method to account for mismodeled and unmodeled perturbations in the dynamics is the inclusion of unmodeled accelerations in the filter. Estimating these three "unmodeled" states allows for the introduction of correlated process noise without modeling it precisely. While the estimation of unmodeled accelerations is more ad hoc than the inclusion of model uncertainties via process noise as described in this paper, the significant heritage of this approach [27] suggests a comparison of the effectiveness of the two different approaches would be valuable for future work.

Other items for future analysis include a study of how the measurement cadence and the measurement error magnitude changes the effectiveness of the precomputed PNC, as well as an investigation of the effect on the precomputed PNC from including higher order gravity coefficient uncertainties. From this investigation of how higher order gravity coefficient uncertainties affect the precomputed PNC, an analysis of how to choose the parameters included in the process noise precomputation can be pursued.

VI. Conclusions

The primary contribution of this paper is an approach that applies known consider covariance analysis tools and filters to precompute a PNC profile along a reference trajectory, and then employ that PNC in an onboard estimation filter. As a result, the need for traditional extensive

manual tuning of PNC is reduced, and robotic systems that experience significant state-dependent perturbations can employ appropriate levels of PNC in onboard navigation filters.

In addition to providing better estimation performance, the new PNC method also allows the onboard navigation filter to better represent the uncertainty in the system state. The example considered here is a better representation of the position and velocity uncertainty of a spacecraft during a descent trajectory that contains gravity perturbations that vary dramatically in magnitude across the reference trajectory.

A secondary contribution of this paper is the translation of the ADF to a consider filter form. The derivation is almost identical to the USKF, where the considered parameters are added to the "state" and then simply not updated in the measurement update portion of the filter (for both the state and covariance updates). The DDT that forms the heart of the ADF can also be used to intuitively map the uncertainty in the consider parameters into the space space. The resulting covariance used as the PNC lacks the cross-covariance contributions, but may prove sufficiently accurate in some applications.

There are limitations of the new precomputed state-dependent PNC method. A reference trajectory must exist, and the vehicle must not deviate too far from the reference path for the precomputed PNC to be effective. The errors in the considered model parameters must also approximately match the model parameter uncertainties used for the PNC precomputation. However, it is noted through a limited analysis that conservative uncertainties in the PNC precomputation may prove more effective than less conservative uncertainties (particularly if there is risk of underestimating the truth gravity errors). While the new precomputed PNC method is shown to be effective for the two examples described in this paper, the method may prove less reliable and effective for scenarios that do not meet these conditions.

VII. Acknowledgment

The work described in this paper was funded by NASA's Chief Technology Office through a NASA Space Technology Research Fellowship (NSTRF) grant. The authors would also like to thank Kevin Berry and Kenny Getzandanner for their assistance on this work.

References

[1] B. Tapley, B. Schutz, and G. Born, *Statistical Orbit Determination.* Elsevier Academic Press, 2004. p. 199–212, 220–229, 499–504.

[2] J. R. Wright, "Sequential Orbit Determination with Auto-Correlated Gravity Modeling Errors," *Journal of Guidance, Control, and Dynamics*, Vol. 4, No. 3, 1981, pp. 304–309.

[3] S. F. Schmidt, "Applications of state space methods to navigation problems," *Advances in Control Systems*, Vol. 3, 1966, pp. 293–340.

[4] M. E. Lisano, "Nonlinear consider covariance analysis using a sigma-point filter formulation," 2006. AAS 06-035.

[5] D. Woodbury and J. Junkins, "On the consider Kalman filter," *AIAA Guidance, Navigation, and Control Conference*, 2010, p. 7752.

[6] J. Stauch and M. Jah, "Unscented Schmidt–Kalman Filter Algorithm," *Journal of Guidance, Control, and Dynamics*, Vol. 38, No. 1, 2014, pp. 117–123.

[7] R. Zanetti and K. J. DeMars, "Joseph Formulation of Unscented and Quadrature Filters with Application to Consider States," *Journal of Guidance, Control, and Dynamics*, Vol. 36, No. 6, 2013, pp. 1860–1864.

[8] K. J. DeMars and R. H. Bishop, "Projecting High-Dimensional Parametric Uncertainties for Improved State Estimation Error Confidence," *Journal of Guidance, Control, and Dynamics*, Vol. 38, No. 9, 2015, pp. 1659–1672.

[9] A. BRYSON, JR and L. Henrikson, "Estimation using sampled data containing sequentially correlated noise.," *Journal of Spacecraft and Rockets*, Vol. 5, No. 6, 1968, pp. 662–665.

[10] B. Friedland, "Treatment of bias in recursive filtering," *IEEE Transactions on Automatic Control*, Vol. 14, No. 4, 1969, pp. 359–367.

[11] M. B. Ignagni, "Separate bias Kalman estimator with bias state noise," *IEEE Transactions on Automatic Control*, Vol. 35, No. 3, 1990, pp. 338–341.

[12] M. E. Hough, "Orbit determination with improved covariance fidelity, including sensor measurement biases," *Journal of Guidance, Control, and Dynamics*, Vol. 34, No. 3, 2011, pp. 903–911.

[13] R. Zanetti and R. H. Bishop, "Kalman filters with uncompensated biases," *Journal of Guidance, Control, and Dynamics*, Vol. 35, No. 1, 2012, pp. 327–335.

[14] A. Gelb, *Applied optimal estimation.* MIT press, 1974. p. 182–184.

[15] E. S. Muller and P. M. Kachmar, "A New Approach to On-Board Orbit Navigation," *Navigation*, Vol. 18, No. 4, 1971, pp. 369–385.

[16] M. Nørgaard, N. K. Poulsen, and O. Ravn, "Advances in derivative-free state estimation for nonlinear systems," Tech. Rep. IMM REP 1998 15, Technical University of Denmark, 2800 Lyngby, Denmark, 2000.

[17] M. Nørgaard, N. K. Poulsen, and O. Ravn, "New developments in state estimation for nonlinear systems," *Automatica*, Vol. 36, No. 11, 2000, pp. 1627–1638.

[18] C. Olson, R. Russell, and J. Carpenter, "Small Body Optical Navigation Using The Additive Divided Difference Sigma Point Filter," *Journal of Guidance, Control, and Dynamics*, 2015. Accepted Sept 2015.

[19] D.-J. Lee and K. T. Alfriend, "Additive Divided Difference Filtering for Attitude Estimation using Modified Rodrigues Parameters," *Journal of the Astronautical Sciences*, Vol. 57, No. 1-2, 2009, pp. 93–111, 10.1007/bf03321496.

[20] J. McMahon, D. Scheeres, D. Farnocchiaz, and S. Chesley, "Optimizing Small Body Gravity Field Estimation Over Short Arcs," *AAS/AIAA Astrodynamics Specialist Conference*, 2015. AAS 15-669.

[21] R. Gaskell, O. Barnouin-Jha, D. Scheeres, A. Konopliv, T. Mukai, S. Abe, J. Saito, M. Ishiguro, T. Kubota, T. Hashimoto, *et al.*, "Characterizing and navigating small bodies with imaging data," *Meteoritics & Planetary Science*, Vol. 43, No. 6, 2008, pp. 1049–1061, 10.1111/j.1945-5100.2008.tb00692.x.

[22] S. Bhaskaran, S. Nandi, S. Broschart, M. Wallace, L. Cangahuala, and C. Olson, "Small Body Landings Using Autonomous Onboard Optical Navigation," *The Journal of the Astronautical Sciences*, Vol. 58, No. 3, 2011, pp. 409–427, 10.1007/bf03321177.

[23] K. Berry, P. Antreasian, M. C. Moreau, A. May, and B. Sutter, "OSIRIS-REx Touch and Go (TAG) Navigation Performance," *38th Annual AAS Guidance and Control Conference*, 2015. AAS 15-125.

[24] K. Berry, B. Sutter, A. May, K. Williams, B. Barbee, M. Beckman, and B. Williams, "OSIRIS-REX TOUCH-AND-GO (TAG) MISSION DESIGN AND ANALYSIS," *Advances in the Astronautical Sciences*, Vol. 149, 2013, pp. 667–678.

[25] A. Konopliv *et al.*, "The Vesta gravity field, spin pole and rotation period, landmark positions, and ephemeris from the Dawn tracking and optical data," *Icarus*, Vol. 240, 2014, pp. 103–117.

[26] J. Civera, A. J. Davison, J. M. M. Montiel, and J. M. M. Montiel, *Structure from motion using the extended Kalman filter*, Vol. 75. Springer Science & Business Media, 2011. p.31.

[27] B. Tapley and D. Ingram, "Orbit determination in the presence of unmodeled accelerations," *IEEE Transactions on Automatic Control*, Vol. 18, No. 4, 1973, pp. 369–373.