Tracking Maneuvering Targets with Multi-Fidelity Interacting Multiple Model Filters

Enrico M. Zucchelli, Zachary R. McLaughlin, Brandon A. Jones
The University of Texas at Austin

ABSTRACT

With the proliferation of low-thrust propulsion and constellations of small satellites the tracking of maneuvering targets becomes increasingly more important and more challenging. One approach to tracking maneuvering targets is to use Interacting Multiple Model filters, which in turn are computationally expensive as they require the propagation of multiple sets following multiple hypotheses. Identifying small, frequent maneuvers requires accurate propagation methods. This paper presents a way to merge Interacting Multiple Model filters with Multi-Fidelity methods to accurately and tractably track maneuvering space objects. Multi-Fidelity approaches reduce computation time, but introduce a systematic error. The proposed filter outperforms an Equivalent Process Noise filter by 20 to 30% even for targets with accelerations as small as 500 nm/s^2, while providing information on the maneuver, and with a computational cost that is between 5 and 10 times larger. The larger the magnitude of the maneuvers, the larger the performance improvement.

1. INTRODUCTION

Maneuvering space objects can be challenging to accurately track because of possible sudden changes in their dynamics. Given the recent and the forecast growth in the number of active small satellites due to the deployment of large constellations such as SpaceX's Starlink or Amazon's Kuiper, tracking maneuvering spacecraft is more and more important for Space Domain Awareness (SDA). Additionally, maneuvers are expected to be frequent, as well as small in magnitude, because of low-thrust propulsion proliferation. Literature on tracking maneuvering targets is wide and several algorithms have been developed for this goal [9]. Three major families of model-based maneuvering target trackers exist. One approach is maneuver estimation, where the maneuver input is estimated together with the target state. This approach requires first that the maneuver is detected, *e.g.*, by a Chi-square test or a gating approach. Additional assumptions on the maneuver are often required in order to have an observable system. It is common to assume a constant input for a certain number of consecutive observations. In Space Object (SO) tracking for SDA, one usually has fewer than one observation per orbit, having more estimated variables (which are virtually infinite, since the thrust can be modulated in time) than observed states. A common assumption in SDA is that the maneuver is fuel optimal [11], in which case the tracking involves the solution of an optimal control problem. Assumptions on the goal of the maneuver may also be made if historical data is available [16]. A second approach consists of using Interacting Multiple Model (IMM) filters. In this context, the spacecraft is assumed to behave according to a Jump-Markov System (JMS). In a JMS, the dynamics of the target follow one of several possible modes, or models, and the switch from one mode to another is modelled as random, with the distribution described by a transition probability matrix. The last of the three major classes of methods is Equivalent Process Noise (EPN) filtering, where the possible maneuvers are treated as additional random perturbations, and their effect is included in the process noise matrix. Data-adaptive methods exist, such that the residuals are statistically consistent even when the target is not maneuvering, or is maneuvering with time-varying magnitude.

Spacecraft maneuvers are often very small in magnitude, and their occurrence is only observable with large delays in time, because of the integrated effect of the maneuver. Hence, it is difficult (if not sometimes impossible, but also not necessary) for a tracker to detect a maneuver right when it happens. Nonetheless, to have high accuracy, it is important that the maneuver can be detected in retrospective. Because of this, a Multiple Hypothesis (MH) method is beneficial. In such a framework, a Multiple Hypothesis Tracker (MHT) always considers several hypotheses; in the case of maneuvering target tracking, to each hypothesis corresponds a different sequence of possible maneuvers; the various hypotheses are propagated forward in time for a certain number of observations, depending on the observations gathered at subsequent epochs and on the selected hypothesis management method. The approach described in this work consists of an IMM, MH filter [15], which traditionally requires large computational capabilities since every already existing hypothesis needs to be propagated according to all available models. To enable a tractable form of this approach, Multi-Fidelity (MF) methods are used to propagate the hypotheses forward in time. MF approaches such as in [7] leverage models with different levels of accuracy, delivering the precision of the higher fidelity methods (except for minor systematic errors) at the computational speed of the lower fidelity methods. Propagation accuracy is especially important to enable the detection of small maneuvers, whereas computational speed is necessary to propagate multiple hypotheses according to multiple models.

The remainder of the paper is organized as follows. Section 2 describes the mathematics of the MF methods. Section 3 introduces the IMM filter, and Sec. 4 presents the filter proposed in this paper. Section 5 illustrates the architecture of the software, which exploits a Graphic Processing Unit (GPU), and the simulation results are reported and discussed in Sec. 6. Section 7 concludes the paper.

2. MULTI-FIDELITY UNCERTAINTY PROPAGATION

MF methods combine a hierarchy of models with varying fidelity to obtain highly accurate and fast prediction models. These methods make use of a few iterations of the most accurate, most expensive models, and many iterations of the least accurate, but fastest models. For this reason, they are suitable when the same model has to be computed many times, with slightly different inputs, such as in uncertainty propagation problems. This work focuses on the use of the bi-fidelity approaches seen in [13,18]. The efficacy of such an approach for orbit-state propagation has been demonstrated in [7] and for orbit determination with a particle filter in [6]. The previous studies considered two representations of the *a posteriori* Probability Density Function (PDF): a Gaussian Mixture Model (GMM) and a particle ensemble. The computational requirements are shown to be reduced by a factor close to 100, at the expense of an additional systematic error that can be bounded [5]. This section outlines the bi-fidelity approach developed as in [7].

The bi-fidelity approach makes use of evaluations of a low-fidelity solver to identify a subset of points to be propagated with a high-fidelity solver. A correction based on the high-fidelity propagations is then applied to all low-fidelity propagated points using *stochastic collocation*. The process is initiated by generating the low-fidelity samples $\boldsymbol{x}^L(\boldsymbol{\xi}) \in \mathbb{X}$ in the state space $\mathbb{X}$. The vector $\boldsymbol{\xi}$ denotes the inputs to the system that determine the point to be propagated. When randomly sampling from a PDF that is multi-variate Gaussian with mean $\boldsymbol{x}_0$ and covariance $\boldsymbol{P}_0$, then

$$\boldsymbol{x}_0(\boldsymbol{\xi}) = \boldsymbol{x}_0 + \boldsymbol{L}_0\boldsymbol{\xi}, \qquad \boldsymbol{P}_0 = \boldsymbol{L}_0\boldsymbol{L}_0^T, \qquad \boldsymbol{\xi} \sim \mathcal{N}(\boldsymbol{0}, \mathbb{I}), \tag{1}$$

where the subscript 0 indicates a sample at the initial time t_0. In the case of orbit-state uncertainty propagation, $\boldsymbol{x}^L(\boldsymbol{\xi})$ would be the state vector $\boldsymbol{x}_0(\boldsymbol{\xi})$ propagated to the time of interest t using the low-fidelity propagator, augmented, in the case of this work, with the constant thrust acceleration. A surrogate approximation based on stochastic collocation and high-fidelity samples $\boldsymbol{x}^H(\boldsymbol{\xi}) \in \mathbb{X}$ is then generated.

The bi-fidelity propagation begins with the generation of the set Ξ of inputs $\{\boldsymbol{\xi}_i\}_{i=1}^m$, which uniquely defines a set of low-fidelity propagated samples represented by the matrix

$$\boldsymbol{X}^L(\Xi) \equiv \begin{bmatrix}\boldsymbol{x}^L(\boldsymbol{\xi}_1) & \cdots & \boldsymbol{x}^L(\boldsymbol{\xi}_m)\end{bmatrix} \in \mathbb{R}^{n\times m}. \tag{2}$$

The matrix $\boldsymbol{X}^H(\Xi)$ follows a similar definition. The samples then define a subset of $\mathbb{X}$

$$\mathbb{X}^L(\Xi) \equiv \operatorname{span}\left(\boldsymbol{X}^L(\Xi)\right) = \operatorname{span}\begin{bmatrix}\boldsymbol{x}^L(\boldsymbol{\xi}_1) & \cdots & \boldsymbol{x}^L(\boldsymbol{\xi}_m)\end{bmatrix} \subseteq \mathbb{X}, \tag{3}$$

which is a function of Ξ. Using stochastic collocation (e.g., see Chapter 20 of [4]), we approximate the points via the surrogate

$$\boldsymbol{x}^L(\boldsymbol{\xi}) \approx \widehat{\boldsymbol{x}}^L(\boldsymbol{\xi}) = \sum_{\ell=1}^{r} c_\ell(\boldsymbol{\xi})\, \boldsymbol{x}^L(\overline{\boldsymbol{\xi}}_\ell), \tag{4}$$

with coefficients c_ℓ, and $r \ll m$ "important" points defined by $\overline{\boldsymbol{\xi}}_\ell$. We use the low-fidelity samples to identify the points $\overline{\boldsymbol{\xi}}_\ell$ with the corresponding set

$$\overline{\Xi} \equiv \left\{\overline{\boldsymbol{\xi}}_\ell\right\}_{\ell=1}^{r}. \tag{5}$$

In the bi-fidelity approach, we use the set $\overline{\Xi}$ identified using the low-fidelity model, and evaluate the propagator to produce $\boldsymbol{x}^H(\overline{\boldsymbol{\xi}}_\ell)$. We then approximate the high-fidelity solution via

$$\boldsymbol{x}^H(\boldsymbol{\xi}) \approx \widehat{\boldsymbol{x}}^H(\boldsymbol{\xi}) = \sum_{\ell=1}^{r} c_\ell(\boldsymbol{\xi})\, \boldsymbol{x}^H(\overline{\boldsymbol{\xi}}_\ell). \tag{6}$$

In summary, the bi-fidelity approach uses the low-fidelity samples to produce the coefficients c_ℓ and the important points $\overline{\boldsymbol{\xi}}_\ell$, and substitutes exact high-fidelity solutions for the nodes $\boldsymbol{x}(\overline{\boldsymbol{\xi}}_\ell)$. This approach includes three assumptions:

- $\boldsymbol{X}^L(\Xi)$ allows for identifying the r samples required for Eq. (6),
- The r high-fidelity samples produce a sufficiently accurate basis for $\boldsymbol{x}^H(\boldsymbol{\xi}) \approx \widehat{\boldsymbol{x}}^H(\boldsymbol{\xi})$, and
- Coefficients c_ℓ of the expansion in Eq. (4) are sufficiently accurate to be leveraged in Eq. (6).

The remainder of this section outlines the procedure for generating c_ℓ and $\overline{\Xi}$ from $\boldsymbol{X}^L(\Xi)$, which are jointly computed and identified in the same algorithm.

We use a greedy algorithm to generate $\overline{\Xi}$, which is an approximate solution to the problem

$$\overline{\Xi} = \arg\min_{\Xi} \inf_{\boldsymbol{y}\in\mathbb{X}^L(\Xi)} \|\boldsymbol{x}^L(\boldsymbol{\xi}) - \boldsymbol{y}\|. \tag{7}$$

The objective function in Eq. (7) is minimizing the distance between the points in $\boldsymbol{X}^L(\Xi)$ and the space $\mathbb{X}^L(\overline{\Xi})$, which is generally not tractable [18]. The greedy algorithm instead leverages a solution to the Pivoted Cholesky decomposition [18]

$$\left[\boldsymbol{X}^L\right]^T \boldsymbol{G}^L \left[\boldsymbol{X}^L\right] = \boldsymbol{A}^T \boldsymbol{L}\boldsymbol{L}^T \boldsymbol{A}. \tag{8}$$

The pivot matrix $\boldsymbol{A}$ orders the samples based on the distance defined in Eq. (7), and $\boldsymbol{G}^L$ is a Gramian matrix generated via $\boldsymbol{X}^L(\Xi)$. Details on the algorithm may be found in [18]. Coefficients $\boldsymbol{c} = \begin{bmatrix} c_1 & \cdots & c_\ell \end{bmatrix}^T$ are then found via

$$\boldsymbol{L}\boldsymbol{L}^T \boldsymbol{c} = \boldsymbol{g}, \tag{9}$$

where

$$g_\ell = \left\langle \boldsymbol{x}^L(\boldsymbol{\xi}), \boldsymbol{x}^L(\overline{\boldsymbol{\xi}}_\ell) \right\rangle, \qquad \ell = 1, \dots, r, \tag{10}$$

and $< \cdot, \cdot >$ denotes the discrete inner product. This algorithm requires the value r be an input, which we autonomously generate using the procedure described in [7]. For the sake of ease of reference, we also define the surrogate-approximated matrix

$$\widehat{\boldsymbol{X}}^H(\Xi) \equiv \begin{bmatrix} \widehat{\boldsymbol{x}}^H(\boldsymbol{\xi}_1) & \cdots & \widehat{\boldsymbol{x}}^H(\boldsymbol{\xi}_m) \end{bmatrix}. \tag{11}$$

3. INTERACTING MULTIPLE MODEL FILTER

An IMM filter differs from a static Multiple Model (MM) filter [2] because it allows switches between modes, making them suitable for tracking maneuvering targets. IMM filters [10, 12, 15] assume that the target behaves according to a JMS, or a hybrid system, where the target state is characterized by a *base state* that evolves according to stochastic difference equations, and a *modal state*, that is instead discrete, and whose evolution is governed by a discrete stochastic process. The stochastic difference equations that govern the evolution of the base state are a function of the modal state. A JMS with non-additive noise evolves as follows:

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}\left(\boldsymbol{x}_k,\, m_k,\, \boldsymbol{v}\left(k,\, m_k\right)\right), \tag{12}$$

where $\boldsymbol{x}_k$ is the base state at epoch k, m_k is the modal state at epoch k, and $\boldsymbol{v}\left(\cdot\right)$ is a random variable. The mode evolves as a Markov chain:

$$P\left(m_{k+1} = j | m_k = i\right) = p_{i,j}, \quad \forall i, j \in \{1,\, ...,\, |M_k|\}, \tag{13}$$

where M_k is the set of all possible models at epoch k. The entries $p_{i,j}$, and the model set M_k may be a function of both the base state and the epoch. Noisy observations of the form

$$\boldsymbol{y}_k = \boldsymbol{h}\left(\boldsymbol{x}_k,\, m_k,\, \boldsymbol{w}\left(k,\, m_k\right)\right) \tag{14}$$

are provided, where $\boldsymbol{y}_k$ is the observation at epoch k and $\boldsymbol{w}\left(\cdot\right)$ is a random variable.

3.1 Baseline IMM Filter

The baseline IMM filter consists of having a single hypothesis prior at epoch $k-1$, propagated to epoch k according to all modes that have non-zero probability to occur from model m_k.

If one uses an Extended Kalman Filter (EKF) for every pair of modes $i, j \in \{1,\, ...,\, |M_k|\}$:

$$\hat{\boldsymbol{x}}^{ij}_{k|k-1} = \boldsymbol{f}\left(\hat{\boldsymbol{x}}^{i}_{k-1|k-1},\, m^j_k,\, \mathbb{E}\left[\boldsymbol{v}\left(k,\, m^j_k\right)\right]\right), \tag{15}$$

$$\boldsymbol{P}^{ij}_{k|k-1} = \boldsymbol{F}^{ij}_k \boldsymbol{P}^{i}_{k-1|k-1} (\boldsymbol{F}^{ij}_k)^T + \boldsymbol{Q}^{ij}_k, \tag{16}$$

$$\hat{\boldsymbol{y}}^{ij}_{k|k-1} = \boldsymbol{h}\left(\hat{\boldsymbol{x}}^{ij}_{k|k-1},\, m^j_k,\, \mathbb{E}\left[\boldsymbol{w}\left(k,\, m^j_k\right)\right]\right), \tag{17}$$

$$\boldsymbol{S}^{ij}_k = \boldsymbol{H}^{ij}_k \boldsymbol{P}^{ij}_{k|k-1} (\boldsymbol{H}^{ij}_k)^T + \boldsymbol{R}^{j}_k, \tag{18}$$

$$\boldsymbol{K}^{ij}_k = \boldsymbol{P}^{ij}_{k|k-1} (\boldsymbol{H}^{ij}_k)^T (\boldsymbol{S}^{ij}_k)^{-1}, \tag{19}$$

$$\hat{\boldsymbol{x}}^{ij}_{k|k} = \hat{\boldsymbol{x}}^{ij}_{k|k-1} + \boldsymbol{K}^{ij}_k \left(\boldsymbol{y}_k - \hat{\boldsymbol{y}}^{ij}_{k|k-1}\right), \tag{20}$$

$$\boldsymbol{P}^{ij}_{k|k} = \boldsymbol{P}^{ij}_{k|k-1} - \boldsymbol{K}^{ij}_k \boldsymbol{S}^{ij} \left(\boldsymbol{K}^{ij}_k\right)^T, \tag{21}$$

where $\hat{\boldsymbol{x}}^{ij}_{k|k}$ and $\boldsymbol{P}^{ij}_{k|k}$ are mean and covariance of the estimate of the base state where the previous mode was i and the current mode is j, $\boldsymbol{F}^{ij}_k$ and $\boldsymbol{Q}^{ij}_k$ are respectively the state transition matrix and the process noise matrix according to the j^{th} model at epoch k, computed around the estimate of model i at epoch $k-1$, and $\boldsymbol{H}^{ij}_k$ and $\boldsymbol{R}^{ij}_k$ are the linearized observation matrix and the observation noise matrix respectively. The weight $\mu^{ij}_{k|k}$, which describes the posterior probability of the system having evolved according to model j at epoch k and according to model i at epoch $k-1$, is:

$$\Lambda^{ij}_k = \mathcal{N}\left(\boldsymbol{y}_k - \hat{\boldsymbol{y}}^{ij}_{k|k-1};\, 0,\, \boldsymbol{S}^{ij}_k\right), \tag{22}$$

$$\mu^{ij}_{k|k} = \frac{1}{c_k} \Lambda^{ij}_k \sum_i p_{i,j} \mu^{i}_{k-1|k-1}, \tag{23}$$

where c_k is a normalization constant such that:

$$\sum_i \sum_j \mu^{ij}_{k|k} = 1. \tag{24}$$

Finally, the probability of model j to have occurred at epoch k is:

$$\mu^j = \sum_i \mu^{ij}_{k|k}, \tag{25}$$

and the distribution of the base state conditioned on model j to have occurred at epoch k is:

$$\hat{\boldsymbol{x}}^{j}_{k|k} = \sum_{i} \frac{\mu^{ij}_{k|k}}{c^{j}_{k}} \hat{\boldsymbol{x}}^{ij}_{k|k}, \tag{26}$$

$$\boldsymbol{P}^{j}_{k|k} = \sum_{i} \frac{\mu^{ij}_{k|k}}{c^{j}_{k}} \left(\boldsymbol{P}^{ij}_{k|k} + \left[\hat{\boldsymbol{x}}^{j}_{k|k} - \hat{\boldsymbol{x}}^{ij}_{k|k} \right] \left[\hat{\boldsymbol{x}}^{j}_{k|k} - \hat{\boldsymbol{x}}^{ij}_{k|k} \right]^{T} \right), \tag{27}$$

where c^{j}_{k} is a normalization constant:

$$c^{j}_{k} = \sum_{j} \mu^{ij}_{k|k}. \tag{28}$$

This baseline filter requires the use of $|M|^2$ EKFs at each time-step, unless $p_{i,j} = 0$ for some pair i, j. The merging of the hypotheses prevents an exponential growth in time $|M|^k$, while on the other hand causing a loss in information. The merging method outlined in Eqs. (26)-(28) is only one of the many possible ones. If a different hypothesis management technique is used, Eqs. (15)-(24) are the same, but the index i would be the index for the hypotheses at the previous step. Each hypothesis can additionally consist of a weighed some of Gaussian components or a GMM [1]; also in this case, there is no modification in Eqs. (15)-(24), except for the fact that now i would be an index that includes all components of all previous hypotheses. A GMM can indeed be interpreted as a MH filter where the different hypotheses are the initial distributions. From now on, hypothesis and GMM component will be used interchangeably.

3.2 UKF-IMM

It is possible to use a bank of Unscented Kalman Filter (UKF) [8] instead of EKFs. UKFs usually perform better than EKFs in nonlinear problems, because they perform statistical linearization over the uncertainty set instead of just locally at the mean. In a UKF, the prior $\hat{\boldsymbol{x}}_{k|k-1}$ and $\boldsymbol{P}_{k|k-1}$ are obtained by a weighed sum of properly selected samples. UKFs can include the nonlinear effect of perturbing accelerations, but this operation is not considered in this paper. In the IMM, Eqs. (15) and (16) are replaced by the following set of equations:

$$\boldsymbol{\chi}^{ijt}_{k|k-1} = \boldsymbol{f}\left(\boldsymbol{\chi}^{it}_{k-1}, m^{j}_{k}, \mathbb{E}\left[\boldsymbol{v}\left(k, m^{j}_{k} \right) \right] \right), \tag{29}$$

$$\hat{\boldsymbol{x}}^{ij}_{k|k-1} = \sum_{t=0}^{2d} \boldsymbol{\chi}^{ijt}_{k} c_{t0}, \tag{30}$$

$$\boldsymbol{P}^{ij}_{k|k-1} = \sum_{t=0}^{2d} c_{t1} \left(\hat{\boldsymbol{x}}^{ij}_{k|k-1} - \boldsymbol{\chi}^{ijt}_{k|k-1} \right) \left(\hat{\boldsymbol{x}}^{j}_{k|k} - \boldsymbol{\chi}^{ijt}_{k|k-1} \right)^{T} + \boldsymbol{Q}^{ij}_{k}, \tag{31}$$

where $\boldsymbol{\chi}^{it}_{k-1}$ are the $2d+1$ σ-points of the initial distribution i, d is the dimension of the base state, c_{t0} are the corresponding weights for the computation of the mean and c_{t1} are the corresponding weights for the computation of the covariance. The σ-points are obtained as follows:

$$\boldsymbol{\chi}^{i0}_{k-1} = \hat{\boldsymbol{x}}^{i}_{k-1|k-1}, \tag{32}$$

$$\boldsymbol{\chi}^{it}_{k-1} = \hat{\boldsymbol{x}}^{i}_{k-1|k-1} + \left(\sqrt{(d+\lambda)} \boldsymbol{L}^{i}_{k-1} \right)_{t,:} \qquad t = 1, ..., d \tag{33}$$

$$\boldsymbol{\chi}^{it}_{k-1} = \hat{\boldsymbol{x}}^{i}_{k-1|k-1} - \left(\sqrt{(d+\lambda)} \boldsymbol{L}^{i}_{k-1} \right)_{t-d,:} \qquad t = d+1, ..., 2d \tag{34}$$

where $\boldsymbol{L}^{i}_{k-1}$ is the lower triangular matrix obtained by the Cholesky factorization of $\boldsymbol{P}^{i}_{k-1}$, $\lambda = \alpha^2(d+\kappa) - d$, $\alpha = 1$, and $\kappa = -3$. The weights are $c_{00} = \lambda/(d+\lambda)$, $c_{01} = \lambda/(d+\lambda) + 1 - \alpha^2 + \beta$ and $c_{t0} = c_{t1} = 1/(2(d+\lambda)$. A similar set of operations is done to compute the expected measurement $\hat{\boldsymbol{y}}_{k|k-1}$, the innovations covariance matrix $\boldsymbol{S}^{ij}_{k|k-1}$, and the Kalman gain $\boldsymbol{K}^{ij}_{k}$. To do so, the following set of equations substitutes Eqs. (17)-(19):

$$\gamma^{ijt}_{k} = \boldsymbol{h}\left(\boldsymbol{\chi}^{ijt}_{k|k-1},\, m^{j}_{k},\, \mathbb{E}\left[\boldsymbol{w}\left(k,\, m^{j}_{k}\right)\right]\right) \tag{35}$$

$$\hat{\boldsymbol{y}}^{ij}_{k|k-1} = \sum_{t=0}^{2d} \boldsymbol{\gamma}^{ijt}_{k} c_{t0}, \tag{36}$$

$$\boldsymbol{S}^{ij}_{k} = \sum_{t=0}^{2d} \left(\boldsymbol{\gamma}^{ijt}_{k} - \hat{\boldsymbol{y}}^{ij}_{k|k-1}\right)\left(\boldsymbol{\gamma}^{ijt}_{k} - \hat{\boldsymbol{y}}^{ij}_{k|k-1}\right)^{T} c_{t1} + \boldsymbol{R}^{j}_{k}, \tag{37}$$

$$\boldsymbol{C}^{ij}_{k} = \sum_{t=0}^{2d} \left(\boldsymbol{\chi}^{ijt}_{k|k-1} - \hat{\boldsymbol{x}}^{j}_{k|k}\right)\left(\boldsymbol{\gamma}^{j}_{t} - \hat{\boldsymbol{y}}^{ij}_{k|k-1}\right)^{T} c_{t1}, \tag{38}$$

$$\boldsymbol{K}^{ij}_{k} = \boldsymbol{C}^{ij}_{k}\left(\boldsymbol{S}^{ij}_{k}\right)^{-1}. \tag{39}$$

The σ-points $\boldsymbol{\chi}^{ijt}_{k|k-1}$ in Eq. (35) may either be the same ones computed in Eq. (29), or may be generated again as in Eqs. (32)-(34), but substituting $\hat{\boldsymbol{x}}^{i}_{k-1|k-1}$ with $\hat{\boldsymbol{x}}^{ij}_{k|k-1}$, and $\boldsymbol{L}^{i}_{k-1}$ with $\boldsymbol{L}^{ij}_{k|k-1}$, the lower triangular Cholesky factorization of $\boldsymbol{P}^{ij}_{k|k-1}$.

4. MULTI-FIDELITY IMM FILTERING

In this section, we explore the implementation of a Bayesian interactive multiple model filtering algorithm integrating the multi-fidelity approach for orbital propagation. As we introduce a large number of hypotheses and models, the multi-fidelity approach is a key enabler to this approach, reducing the computation requirements by approximately 100 times. At each epoch, the steps for the filter are the following:

1. Random selection of thrust acceleration set, each thrust corresponding to a mode of the IMM
2. Random selection from a prior distribution of points used to generate the MF Gramian.
3. MF propagation of the σ-points of the previous Gaussian components according to all the selected thrust models
4. Estimation of MF systematic error
5. Computation of posterior mean, covariance and weight of each hypothesis
6. Hypotheses management

A spacecraft may maneuver according to an infinite number of models, because their thrust can be modulated according to a continuum spectrum. However, as is often done in filtering, the maneuvers can be discretized, either by picking discrete randomly distributed maneuvers (as is done in particle filters, for example) or by adding to each model propagation an additional dynamic noise term to include uncertainty in the maneuver. This is similar to Gaussian splitting approaches, and involves in this case making use of consider filters. Hence, in step 1, a set of continuous thrust accelerations between the two observations is randomly drawn according to a prior, user-provided, distribution. The selection is randomized so that constant bias in the estimation can be avoided. In step 2, a reduced set of points is randomly drawn from the σ-points and the thrust prior distribution; these points are included in the set Ξ and used to select the set of important points $\bar{\boldsymbol{\xi}}_{\ell}$, and then generate the MF coefficients $c_{\ell}(\boldsymbol{\xi})$ as in Eqs. (7)-(10). A representative subset is required because several operations of the algorithm that generates the MF coefficients scale with n^3, n being the number of samples, and thus would quickly become intractable if all the σ-points were used. The initial conditions of the samples are selected from the prior distribution, and the corresponding thrust is selected from the above-mentioned thrust distribution. After the coefficients for the MF correction are computed, in step 3 all σ-points generated according to Eqs. (32)-(34), which can be indexed by the appropriate inputs i

(indexing the GMM components of the prior) and t, are propagated according to all the filter modes, indexed by j, with a MF approach. The MF propagation substitutes Eq. (29), which is repeated with different inputs $|H_{k-1}|\,|M_k|\,(2d+1)$ times, H_{k-1} being the set of hypotheses at epoch $k-1$. In step 4, a set of m randomly selected σ-points is selected and propagated in high-fidelity; the High-Fidelity (HF) propagation is compared to the corresponding MF approximation, to compute the systematic error matrix $\boldsymbol{Q}_t$ [5]:

$$\boldsymbol{Q}_t = \frac{\tau}{m}\sum_{i=1}^{m}\left(\boldsymbol{x}^H(\widetilde{\boldsymbol{\xi}}_i) - \widehat{\boldsymbol{x}}^H(\widetilde{\boldsymbol{\xi}}_i)\right)\left(\boldsymbol{x}^H(\widetilde{\boldsymbol{\xi}}_i) - \widehat{\boldsymbol{x}}^H(\widetilde{\boldsymbol{\xi}}_i)\right)^T . \tag{40}$$

This matrix is added to the process noise matrix $\boldsymbol{Q}^{ij}$ in Eq. (31). The scale factor τ is selected to be equal to 2^2 based on previous work [5]. In step 5, statistical inference is performed as described in Sec. 3, and finally in step 6 some of the hypotheses are trimmed to keep the problem tractable.

In a way, this approach to IMM can be seen as an automatic Gaussian mixture splitting in the perturbations distribution, with the additional fact that the probability of a certain distribution occurring for a certain component depends on the thrust that was used to propagate the same component at the previous step, making the model an IMM. Such a method is expected to outperform simple process noise inflation methods for maneuvering target tracking for the following reasons:

- Better nonlinearity representation:
    - the nonlinear effect of the thrust is better accounted for if several thrust models with small uncertainty are considered, rather than a single one with larger uncertainty, since a large uncertainty implies a large domain of linearization.
- Dependency on previous maneuver(s):
    - the IMM filter can manage sequences of maneuvers that are not independently distributed in time.
- Possibility to include non Gaussian perturbations:
    - the thrust distribution from which the modes are picked in step 1 of the algorithm does not need to be Gaussian, hence having better generalization than an EPN approach.
- Multiple hypotheses management:
    - multiple hypotheses of different previous thrust accelerations can be propagated forward in time, thus allowing the filter to decide whether a maneuver happened or not after more observations.

As mentioned above, for the current version of the filter, the perturbations are drawn at random from a prescribed distribution to avoid constant biases. An issue of this approach is that many components may end up overlapping one another, thus having means very close to one another, decreasing performance and efficiency of the filter. This problem is caused by the fact that different combinations of initial conditions and thrust profiles may lead to the same final state. This issue may be solved in future versions of the filter by predominantly picking the thrust accelerations along the directions that cause less overlapping in the propagated prior.

4.1 Hypotheses Management

Automatic splitting in the maneuver space causes an exponential growth in time of the number of models. Hence, hypotheses management is required to keep the number of components from becoming intractable. If one attempts to minimize information loss while limiting the number of components below a certain threshold, then merging is required, in addition to pruning. Merging in an IMM results in a loss of information on the previous maneuver, and thus on the probability of the next maneuvers: it is only possible to merge GMM components that have been propagated according to the same last mode. This is inefficient, and causes a loss in information, since components that have been propagated according to the same model are likely to be, on average, as different from one another as the prior were. Pruning may be problematic too in the case of maneuvering target tracking. The weight of the true PDF may be split over overlapping components, which may end up being pruned despite their summed weight may be larger than that of some non-pruned components. This issue is especially true with maneuvering targets, since there is no more a one-to-one relation between initial and final state: any final state can be reached by an infinite family of initial conditions and maneuvers. This fact should be kept in mind when considering the number of thrust models to evaluate at each time-step, since having n thrust models implies discarding $1 - 1/n \times 100\%$ of the components at each time step if no merging is implemented. Because of this, the number of thrust models is limited to 20 in this paper.

4.2 Multi-Fidelity Propagation of Maneuvering Targets

As mentioned above, there is no one-to-one relation between the initial and final state when a spacecraft is maneuvering. MF methods usually only consider the output state, and not the input, when generating the high-fidelity surrogate. As a consequence, if only the final state is considered, the correction is ambiguous, as the final state may have been obtained by an infinite set of possible trajectories, for which the difference between low- and high-fidelity model accelerations differ. Hence, it is important that the MF model can distinguish between two trajectories with similar final state, but different initial conditions and maneuver. In this paper we only consider constant low-thrust maneuvers between two observations. Thus, two states, together with their time-tags, are sufficient to uniquely define the thrust and the full trajectory. In [7], the output used for the MF includes 20 steps after the final time, distanced by 15 s between one another, for a total of 300 s. While this would in theory be sufficient to uniquely determine the maneuver and thus make an accurate correction of the final state, it is in practice not enough to accurately determine thrust values as small as 500 nm/s^2. After some tuning, it is found that the optimal output consists of 20 time-steps taken every 50 s, for a total duration of 1,000 s. The longer the duration, the better the thrust acceleration is discerned; at the same time though, the longer the duration, the more the time steps are separated from one another, and thus the more nonlinear the interpolated trajectory is, making the computation harder. Hence, for thrust values between 500 and 5,000 nm/s^2, 1,000 s is close to the optimal trade-off.

5. UTILIZING GPU PROCESSES

In order to improve the efficiency of the multi-fidelity propagator, an additional option for the low-fidelity state propagation is under continued construction. This section of the code propagates the low-fidelity samples on an NVIDIA Tesla K80 GPU via the the NVIDIA CUDA compute architecture. This code is a minimally invasive addition to the existing bi-fidelity propagation software. The primary objective of constructing a CUDA-based architecture is to reduce the time it takes to generate low-fidelity samples on the CPU.

The CUDA low-fidelity propagator requires only the inputs necessary to predict a set of Monte Carlo samples via a low-fidelity model: the Cholesky decomposition of the covariance, the mean state, a set of times, and a number of low-fidelity states to propagate. The software then chooses a block-thread architecture which will allow for the required number of samples to be generated and propagated. On the GPU, individual threads are responsible for propagating one Cartesian state vector at one explicit end time. This involves the variance of the mean state and 2-body J_2 propagation using a Runge-Kutta 4$^{\text{th}}$ order method before combining data into a large array and returning this information, along with the generated pseudo-random-numbers, to the CPU. Once this process is complete, the remainder of the bi-fidelity code continues as normal.

At this time, propagating low-fidelity samples on the GPU is approximately 30% faster than propagating the samples on the CPU. This is actively being improved and tested with changes to the memory and run architectures such that the runtime may be improved over the CPU-based implementation. A GPU architecture was selected for this project because of the ability to perform massively parallel tasks on the order of millions of processes simultaneously with small increases in overhead. The current area on which improvement is focused is around strided memory access to global CUDA memory, which can cause significant increases in process time. Some amount of strided access may be necessary in order to accomplish highly complex tasks on the card, but methods are under investigation, such as asynchronous booting and data transfer and the creation of multiple shared memory data structures, in order to decrease the time spent on the GPU card.

While the results of the CUDA-based implementation have offered a slight increase in efficiency, a larger speedup is desired. It is important to note that the benefits of a GPU-based implementation are not equal across different graphics cards, with some having more compute capability and some having less. Additionally, some cards may have different software-based architectures or rules, which would make the code written for this process unable to execute properly or compile. Regardless, code may be optimized for specific graphics cards, which will improve performance and should result in at least slight speedup.

6. SIMULATION RESULTS

Table 1. Orbit state initial conditions and 1σ uncertainty.

	a [km]	e [-]	i [deg]	Ω [deg]	ω [deg]	ν [deg]
	6878	7.7×10^{-4}	45	0	90	0
	x [m]	y [m]	z [m]	$\dot{x}$ [mm/s]	$\dot{y}$ [mm/s]	$\dot{z}$ [m/s]
σ	100.0	100.0	100.0	10.0	10.0	10.0

6.1 Scenario Description

Numeric results consider the first detection and follow-up tracking of a low-Earth orbit space object when observed via two ground stations generating radar measurements. The true trajectory depends on the thrust profile, and the initial conditions and uncertainty are shown in Table 1. Table 2 describes the high- and low-fidelity models used for the MF propagation. The high-fidelity model also defines the true dynamics environment. Table 3 defines the parameters used in both force models. At each observation epoch, one of three ground stations generates radar measurements

$$\boldsymbol{y} = [\,\rho,\, \dot{\rho},\, \alpha,\, \delta\,]^T \,, \tag{41}$$

where ρ is the measured range, $\dot{\rho}$ the measured range-rate, and α and δ are measured ascension and declination respectively. The ground station that initially detects the space object, GS1, is located at 44.77° latitude, 83.65° longitude, and a radius of $R_{\oplus}$, with no velocity in the Earth-fixed frame. The second and third stations have the same longitude and radius, but latitudes of 0° and −44.77°, respectively. The ground station locations are selected to (i) produce an initial detection at epoch for the given initial conditions, (ii) a second measurement of the object approximately one orbit after initial detection, and (iii) yield approximately one measurement with each spacecraft orbit. Measurement errors are Gaussian with zero mean, zero correlation, and standard deviations of 30 m in range, 50 arcsec for each of the angles, and 0.03 m/s in range-rate.

Table 2. Force models for low- and high-fidelity fully simulated tests.

Model	Low-Fidelity	High-Fidelity
Central Body Gravity	Two-Body and J_2	70×70
Third-Body Perturbations	None	Sun and Moon
Solar Radiation Pressure	None	Cannonball
Atmospheric Drag	None	Cannonball
Coordinate System Reduction	None	IAU2006 [14]

The simulation begins when the spacecraft is detected by GS1. One measurement is generated with each pass over each station for a total of 14 measurements. The simulated scenario ends 24 hours after the epoch time. The maximum number of components chosen for the IMM-MF filter after update is 300 in all cases. The number of thrust profiles is limited to 20. Hence, the IMM-MF filter propagates 600 Gaussian components at each time-step. Thanks to the MF propagation, this ends up being approximately only 10 times computationally more expensive than a process noise inflation approach. The performance of the IMM-MF filter is evaluated by, at each epoch, merging all hypotheses into a single Gaussian hypothesis, despite better metrics exist for a Gaussian mixture. This should be kept in mind when comparing the results with the EPN filter.

The filter is initialized with mean centered at the truth, and uncertainty with standard deviation of 100.0 m in each position direction and 0.01 m/s in each velocity direction. The corresponding Gaussian is then split into three along the x-axis using the method from [3].

In this paper we do not fully exploit the capabilities of IMM filters, and the probability to transition from mode i to mode j, while it may be a function of j (depending on the thrust scenario), is assumed independent of i. The benchmark is a baseline EPN filter. The process noise in the EPN is computed as follows [17]:

$$\boldsymbol{Q}_k = \sigma_Q^2 \,\boldsymbol{\Gamma}(t_k, t_{k-1})\,\boldsymbol{\Gamma}^T(t_k, t_{k-1}), \tag{42}$$

$$\boldsymbol{\Gamma}(t_k, t_{k-1}) = \left[\frac{(t_k - t_{k-1})^2}{2}\boldsymbol{I}_3 \;\; (t_k - t_{k-1})\boldsymbol{I}_3\right]^T, \tag{43}$$

where $\boldsymbol{I}_3$ is a 3×3 identity matrix. The value for σ_Q is chosen to be 500 nm/s^2 unless otherwise specified. The structure of the process noise matrix is chosen because it is derived by assuming random acceleration with standard deviation σ_Q throughout the propagation time.

Table 3. Force model and satellite parameters for all cases.

Parameter	Value
Satellite Mass	500 kg
Satellite Drag and SRP Area	1 m^2
C_D	2.0
C_R	1.5
Epoch Time	2455200.5 UTC
Earth Radius $R_\oplus$	6378.1363 km
Gravitation Parameter	398600.4415

6.2 Constant Thrust

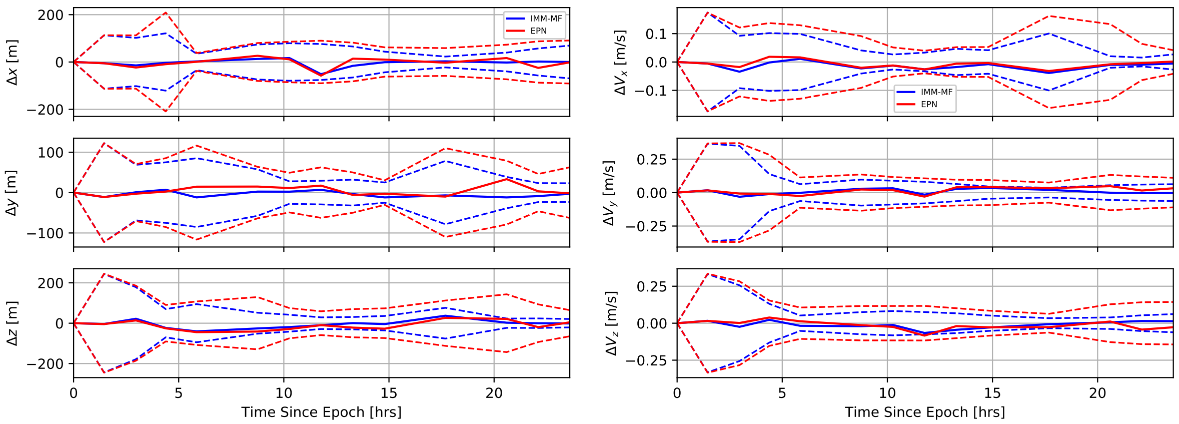

Fig. 1. Position (left) and velocity (right) error for the IMM-MF filter (blue) and for the EPN filter (red) for the case where the spacecraft maneuvers with constant in-track thrust of 500 nm/s^2. The dashed lines correspond to the 3σ uncertainties estimated by the filters.

Figure 1 shows the error in position and velocity for the estimates of the IMM-MF and the EPN filter for the case where the spacecraft accelerates with continuous thrust of magnitude 500 nm/s^2 in the in-track direction only. For the IMM-MF filter, the 20 thrust models are picked randomly from a multivariate normal distribution, with standard deviation of 500 nm/s^2 in each direction (either radial, in-track, or cross-track). The value of σ_Q in Eq. (42) for the EPN filter is 500 nm/s^2. For this case, the IMM-MF filter has an Root Mean Square Error (RMSE) of 28.1 m in position and 40.9 mm/s in velocity, whereas the benchmark filter has an RMSE of 36.9 m in position and 48 mm/s in velocity. Both filters are statistically consistent, and the IMM-MF filter is less conservative in the estimation of the uncertainty. The difference in performance between the two approaches becomes more marked if the thrust is increased. With a 10 times larger thrust magnitude in both truth and filters models, the RMSE for the IMM-MF filter is 40.2 m in position and 43 mm/s in velocity, while the error for the EPN filter is of 111.2 m in position and 132 mm/s in velocity. Using the IMM MF filter thus reduces the error by a factor of three in the latter case. Such a result is expected, since one of the advantages of the IMM-MF filter is that of better accounting for the nonlinear behavior of the perturbations, and nonlinearities are larger for larger accelerations.

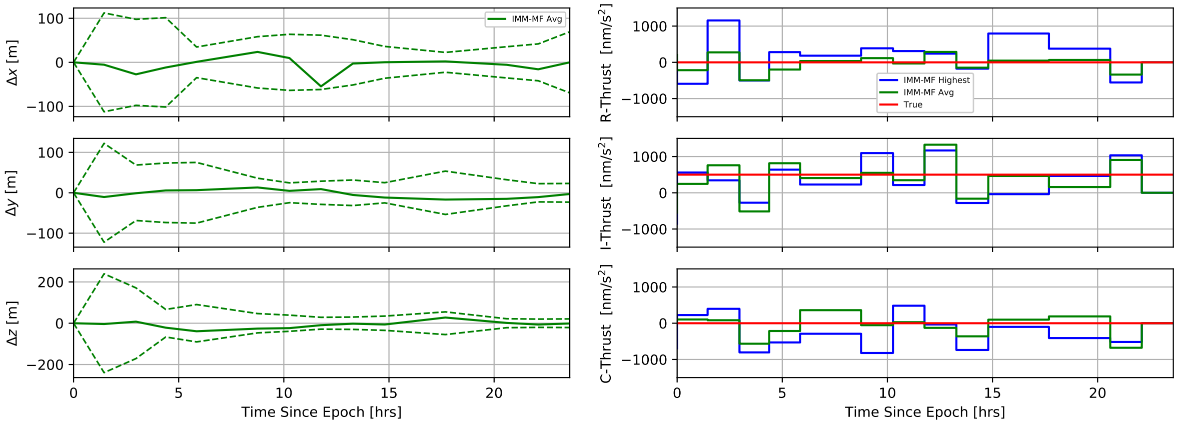

Fig. 2. Position error (left) and thrust models picked (right) for the IMM-MF filter with posterior weights for the case where the spacecraft maneuvers with constant in-track thrust of 500 nm/s^2. The dashed lines correspond to the 3σ uncertainties estimated by the filter.

It is possible to evaluate the performance of the filter by only considering the hypotheses that are kept until the end. This performance is expected to be better than that of the real-time filter, since at each epoch it makes use of information obtained at the following epochs, namely the updated likelihood of each hypothesis. Figure 2 shows the performance of the filter when only the 300 hypotheses that have the highest weight at the end are considered, and, on the left, the corresponding thrust models picked. "Highest" refers to the solution obtained by considering only the highest weighed hypothesis, whereas "Avg" is a weighed sum of the 300 highest weighed hypotheses. The RMSE does not decrease, but the uncertainty of the filter is reduced. This result suggests that a better hypothesis management technique may be used: the pruning only method artificially decreases the entropy at every step, thus making the filter more confident. When evaluating the thrust model plot, one should keep in mind that the direction that most affects the state is the in-track one. While not accurately tracking the in-track thrust component at each time-step, the filter picks models whose average in time is quite close to the true in-track thrust.

6.3 Intermittent, Random Uniformly Distributed Thrust

In this scenario, the thrust is kept constant for intermittent periods of 10,000 s. It is zero with probability 0.85, and uniformly distributed between ± 500 nm/s^2 in the three directions otherwise. The IMM-MF filter gives a probability of 0.85 to a maneuver with zero thrust, and equal probability to all of the remaining models, which are uniformly distributed between ± 500 nm/s^2 in the three directions. σ_Q of Eq. (42) is reduced to 43 nm/s^2, to account for the reduced variance of a uniform distribution, and to account for the intermittent thrust. In this case, the propagation duration is increased to 48 hrs. Figure 3 shows the performance of the filters in this scenario. The RMSE in position of the IMM-MF filter is 21.9 m, against

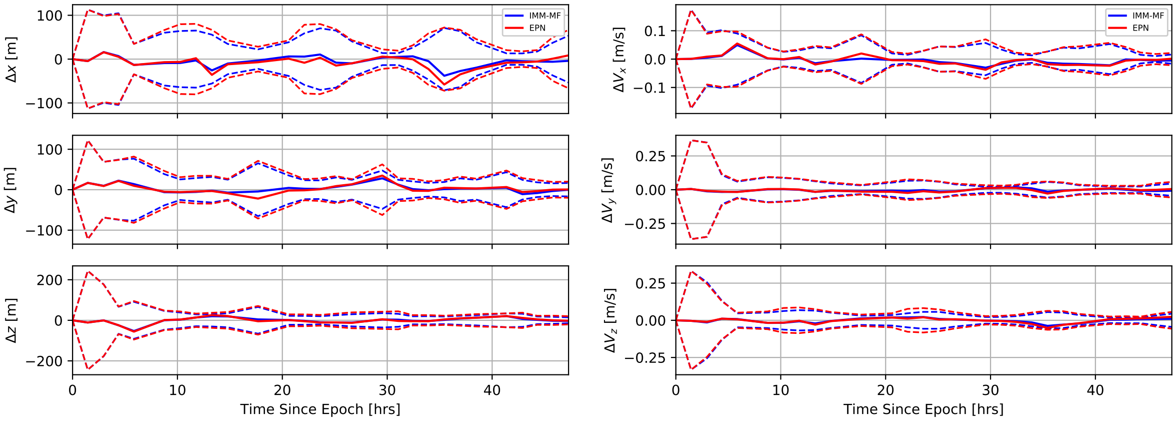

Fig. 3. Position (left) and velocity (right) error for the IMM-MF filter (blue) and for the EPN filter (red) for the case where the spacecraft maneuvers with random, intermittent uniformly distributed thrust between -500 nm/s^2 and 500 nm/s^2. The dashed lines correspond to the 3σ uncertainties estimated by the filters.

the 26.1 m of the EPN. The RMSE in velocity of the IMM-MF filter is 23.8 mm/s, against the 28.3 mm/s of the EPN.

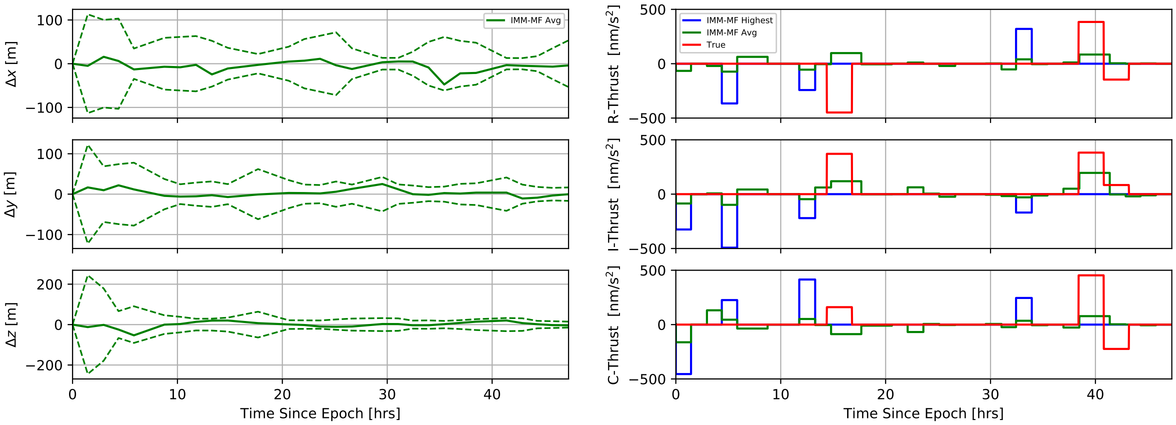

Fig. 4. Position error (left) and thrust models picked (right) for the IMM-MF filter with posterior weights for the case where the spacecraft maneuvers with intermittent, random thrust uniformly distributed in ±500 nm/s^2 in all three directions. The dashed lines correspond to the 3σ uncertainties estimated by the filter.

Figure 4 shows the performance of the filter when only the 300 hypotheses that have the highest weight at the end are considered, and the corresponding thrust models picked. Also in this case, the RMSE does not decrease, but the uncertainty of the filter is reduced. While the highest weighed hypothesis picks several maneuvering models also when there are no maneuvers occurring, the average of all the hypotheses has a behavior that is more consistent with the truth.

6.4 Process Noise Only

Finally we test the situation where there is only process noise. The true model is subject to a randomly distributed acceleration that changes, independently in time, every 60 s. The standard deviation is in this case increased to 5,000 nm/s^2 for each direction, both in the true model and in the IMM-MF model; σ_Q for the EPN filter is also increased to 5,000 nm/s^2. Figure 5 shows the error in position and velocity of

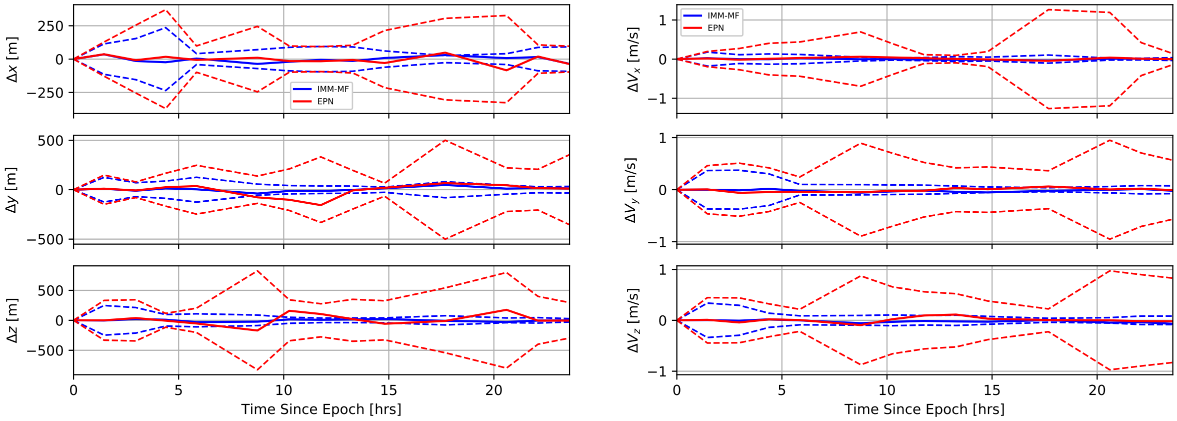

Fig. 5. Position (left) and velocity (right) error for the IMM-MF filter (blue) and for the EPN filter (red) for the case where the spacecraft maneuvers with random, fast-changing, thrust, equivalent to process noise, normally distributed with 5,000 nm/s^2 in standard deviation in each direction. The dashed lines correspond to the 3σ uncertainties estimated by the filters.

the two filters for this scenario. The position RMSE is 34.7 m for the IMM-MF and 118.0 m for the EPN, whereas the velocity RMSE is 48.1 mm/s for the IMM-MF and 71.8 mm/s for the EPN. As expected the IMM-MF filter outperform the EPN filter also when tracking a non-maneuvering target, mostly thanks to the capability of better representing the nonlinear effects of perturbations.

7. CONCLUSIONS

This paper presents a combination of multi-fidelity and multiple model filtering to reduce computation time in uncertainty propagation while accounting for the possibility of a maneuvering target. An IMM-MF filter that tracks 300 hypotheses and propagates them all according to 20 different maneuvering models is tested against an EPN filter approach. While more computationally demanding by a factor between 5 and 10 times, the presented filter provides solutions that always outperform the benchmark. The difference is more marked, the larger the maneuvers, because of the nonlinear effects of the maneuvers, which are lost in an EPN approach. The filter is additionally tested for the case in which there is a non-maneuvering target whose motion is subject to large process noise, and again outperforms the benchmark. Preliminary results suggest that better hypotheses management techniques, that include merging, may improve the performance. Some of the benefits of having an IMM that propagates multiple hypotheses have not been exploited yet in this paper, such as having not independent sequences of maneuvers, and probability of the models dependent on the propagated state (*e.g.*, according to historical data), and will be explored next. In future work we also plan to have state jumps in the filter at epochs that are independent of the observations, together with the use of consider filter to include uncertainty in the thrust in each mode.

8. ACKNOWLEDGEMENTS

This material is based upon work supported by the Air Force Office of Scientific Research under award number FA9550-19-1-0404. Any opinions, finding, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Air Force.

9. REFERENCES

[1] Daniel L. Alspach and Harold W. Sorenson. Nonlinear Bayesian estimation using Gaussian sum approximations. *IEEE Transactions on Automatic Control*, AC-17(4):439–448, August 1972.

[2] Yaakov Bar-Shalom, X Rong Li, and Thiagalingam Kirubarajan. *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2004.

[3] Kyle J. DeMars, Yang Cheng, Robert H. Bishop, and Moriba K. Jah. Methods for splitting gaussian distributions and applications within the AEGIS filter. In *2012 AAS/AIAA Space Flight Mechanics Meeting*, Charleston, SC, Jan. 29 - Feb. 2 2012.

[4] Roger G. Ghanem, David Higdon, and Houman Owhadi, editors. *Handbook of Uncertainty Quantification*. Springer International Publishing, Switzerland, 2017.

[5] Brandon Jones, Emmanuel Delande, Enrico Zucchelli, and Moriba Jah. Multi-fidelity orbit uncertainty propagation with systematic errors. *amos*, page 14, 2019.

[6] Brandon A. Jones. Multi-fidelity methods for orbit determination. In *Proceedings of the 2018 Advanced Maui Optical and Space Surveillance Technologies Conference*, Wailea, Maui, Hawaii, September 11-14 2018.

[7] Brandon A. Jones and Ryan Weisman. Multi-fidelity orbit uncertainty propagation. *Acta Astronautica*, 155:406–417, February 2019.

[8] Simon J. Julier and Jeffrey K. Uhlmann. A new extension of the Kalman filter to nonlinear systems. *Proceedings of SPIE*, 3068:182–193, 1997.

[9] X Rong Li and Vesselin P Jilkov. Survey of maneuvering target tracking: decision-based methods. In *Signal and Data Processing of Small Targets 2002*, volume 4728, pages 511–534. International Society for Optics and Photonics, 2002.

[10] Xiao-Rong Li and Yaakov Bar-Shalom. Multiple-model estimation with variable structure. *IEEE Transactions on Automatic Control*, 41(4):478–493, April 1996.

[11] Daniel P. Lubey and Daniel J. Scheeres. An optimal control based estimator for maneuver and natural dynamics reconstruction. In *Proceedings of the 2013 Advanced Maui Optical and Space Surveillance Technologies Conference*, Wailea, Maui, Hawaii, September 2013.

[12] Efim Mazor, Amir Averbuch, Yakov Bar-Shalom, and Joshua Dayan. Interacting multiple model methods in target tracking: a survey. *IEEE Transactions on aerospace and electronic systems*, 34(1):103–123, 1998.

[13] Akil Narayan, Claude Gittelson, and Dongbin Xiu. A stochastic collocation algorithm with multifidelity models. *SIAM Journal on Scientific Computing*, 36(2):A495–A521, 2014.

[14] Gérard Petit and Brian Luzum. IERS conventions (2010). IERS Technical Note 36, International Earth Rotation and Reference Systems Service (IERS), Frankfurt am Main, Germany, 2010.

[15] X. Rong Li and Vesselin P. Jilkov. Survey of maneuvering target tracking. part v. multiple-model methods. *IEEE Transactions on Aerospace and Electronic Systems*, 41(4):1255–1321, Oct 2005.

[16] J Siminski, T Flohrer, and Thomas Schildknecht. Assessment of post-maneuver observation correlation using short-arc tracklets. *Journal of the British Interplanetary Society*, 70:63–68, 2017.

[17] Byron D. Tapley, Bob E. Schutz, and George H. Born. *Statistical Orbit Determination*. Elsevier Academic Press, Burlington, MA, first edition, 2004.

[18] Xueyu Zhu, Akil Narayan, and Dongbin Xiu. Computational aspects of stochastic collocation with multifidelity models. *SIAM/ASA Journal on Uncertainty Quantification*, 2(1):444–463, 2014.