# FILTER DESIGN FOR SMALL INTEGRATED NAVIGATOR FOR PLANETARY EXPLORATION

David Heise, Stephen Steffes, Stephan Theil
DLR German Aerospace Center, Institute of Space Systems
Robert-Hooke-Str. 7, 28359 Bremen, Germany

**Summary**

The mass of a vehicle is limited by today's space propulsion technology. By reducing the mass of the components or of the complete subsystem it would be possible to shorten travel time or to increase the payload. One of the subsystems is the navigation system. Today's navigation systems for exploration missions have a total mass of over 10 kg. The aim is to reduce this total mass to less than 3 kg without loss of accuracy of the navigation system. To reach this goal the navigation system uses a combination of different sensors. One possible combination of sensors is introduced here. For this combination, a simulation was used to check how accurate this navigation solution is and how much influence the different sensors have on the navigation solution.

## 1. INTRODUCTION

With today's space propulsion technology one of the most limiting factors is the mass of the vehicle. By reducing the mass of its components or of complete subsystems it would be possible to shorten travel time by using more delta-v, reduce the launch cost by using smaller space transportation systems, increase the payload or allow new types of missions by adding new capabilities to the spacecraft.

The SINPLEX project's main objective is to minimize the navigation subsystem for exploration missions. The aim is to develop a new navigation subsystem with reduced weight and size without losing accuracy. To achieve this goal, the navigation subsystem works with a combination of different sensors in which data is fused by using the Extended Kalman Filter (EKF).

## 2. DESIGN

### 2.1. The Scenario

The navigation system analyzed in this work was designed to work in different mission phases like approach, rendezvous and landing missions. For this work a lunar landing scenario was chosen. This scenario begins in a 100 km x 10 km altitude descent orbit. The power descent phase begins near perigee and ends at 2 km altitude, where the landing phase starts. The simulation includes all the orbit parts of the scenario.

### 2.2. The Sensors

The main goal was to design a simulation for a planetary exploration navigation system. This simulation made it possible to see how much influence the different types of sensors have on the navigation results and to choose an optimal combination accordingly. The types of sensors which are used in this design are:

- Inertial Measurement Unit (IMU)
- Star Tracker (STR)
- Laser-Altimeter (LA)
- Navigation Camera (NC)

#### 2.2.1. The Inertial Measurement Unit

The IMU measures acceleration and rotation. The acceleration is measured through three orthogonally arranged accelerometers. The rotation is measured through three orthogonally arranged gyroscopes. By using the orthogonal arrangement it is possible to calculate the rotation and acceleration in any direction. The IMU model was based on MEMS sensors, which are inaccurate, but have very low mass. In this work, the following parameters were evaluated and had the following baseline values:

- Sample rate: 100 Hz
- Gyro Bias: 750 deg/hr
- Gyro Scale Factor: 3600 ppm
- Gyro Random Walk: 2 deg/sqrt(hr)
- Gyro Bias Stab: 15 deg/hr
- Accel Bias: 150 mg
- Accel Scale Factor: 12600 ppm
- Accel Random Walk: 1 m/s/sqrt(hr)
- Accel Bias Stab: 1.5 mg

The sensor model for the simulation was taken from the High Performance Satellite Dynamics Simulator (HPS), which is a library of sensor and dynamics models.

#### 2.2.2. The Star Tracker

The STR is a combination of a camera and an image processing algorithm to measure inertial attitude. The camera creates the image of the stars. This image is processed by the algorithm which calculates the attitude. This calculation is based on the fixed-stars appearing in the image. For this work, the following parameters were evaluated and had the following baseline values:

- Sample rate: 1 Hz

- x-Field of View: 16 degree
- y-Field of View: 16 degree
- Pixel resolution: 1024x1024
- Accuracy: 36 arcsec

The sensor model for the STR was taken from the HPS Database too.

#### 2.2.3. The Laser Altimeter

The LA measured the distance to an object. To calculate this distance, the sensor uses the time interval from the laser, which is reflected by the object. This kind of measurement works as long as the relative velocity of the object is not close to the speed of light. The parameters from the LA are:

- Sample rate: 2 Hz
- Measurement noise: 0.12 m

#### 2.2.4. The Navigation Camera

The images from the NC are used in two different algorithms: feature tracking and crater detection. For this work, the NC had the following parameter and baseline values:

- Field of View: 60 degree
- Pixel resolution: 1024x1024

##### 2.2.4.1. The Crater Navigation Algorithm

The Crater Navigation Algorithm (CNA) calculates the position in relation to the image. The accuracy of this algorithm depends on the surface maps. The craters on the image are evaluated and the algorithm tries to figure out which craters are in the image. Through the position of the crater on the image and the map, it is possible to calculate the position of the camera in relation to the image [1]. In this simulation, the crater navigation algorithm has the following parameters:

- Measurement noise: 300 meters
- Measurement frequency: 1 Hz
- Minimum altitude: 10 km

The minimum altitude is necessary to see enough craters in the database to get a valid measurement.

##### 2.2.4.2. The Feature Tracker Algorithm

The Feature Tracking Algorithm (FTA) evaluates the images from the NC and tracks any found features in the images. The result of this algorithm is the feature ID and the pixel coordinate on the image. This information is first converted into an optical flow measurement before being used in the filter. The simulation model was simplified so that the moon was assumed to be a sphere. The features which are tracked were only at the surface of the sphere and have no relation to the real surface design of the moon. It is only an abstract simulation to test the behavior of the FTA and whether it is applicable. The following parameters were used for the FTA:

- Measurement noise: 0.5 pixel
- Measurement frequency: 4 Hz
- Number features used: 10

### 2.3. The navigation filter

The navigation filter is a combination of the strapdown algorithm and an EKF. The implementation of such a navigation filter with a different sensor combination was done by Steffes [2]. This implementation was modified to fit the new sensor combination. The new concept of the filter is shown in Figure 1.

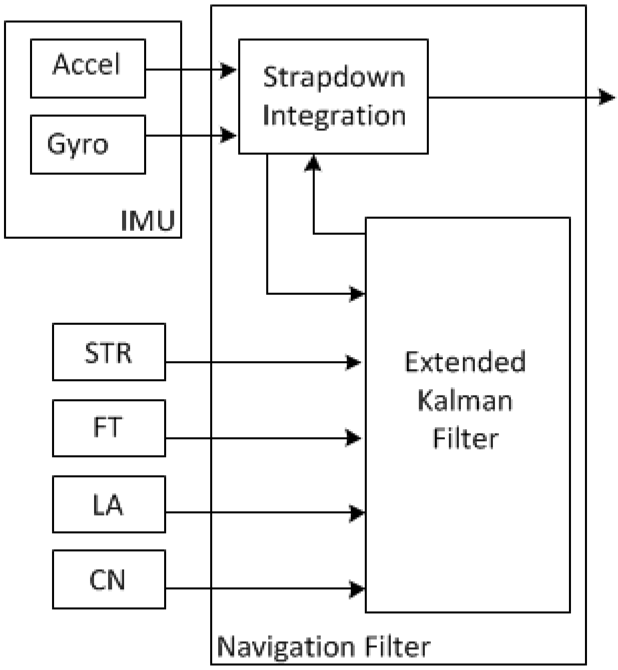


**Figure 1: Navigation filter block diagram.**

The strapdown algorithm integrates the estimated position, velocity and attitude of the vehicle using the IMU measurements. Because of the IMU measurement noise the accuracy of this calculation is very poor. To get a more accurate solution the EKF corrects the position, velocity and attitude using the measurements of the other sensors. See Steffes [2] for more details.

## 3. THE FEATURE TRACKER

This chapter explains the realization of the feature tracker model and how the measurements of this model are used in the EKF.

### 3.1. The Feature Tracker Model

The feature tracker model returns a list of pixel coordinates in the image and a unique ID for each feature. To create a feature, an abstract method was used. We used a spherical moon and through the position of the navigation camera the moon is always in the field of view.

The first step of the FTA is to calculate which part of the surface of the moon is in the field of view. After this step, the feature tracker checked if old features are still in the field of view and calculated the new pixel coordinates. The limit of traceable features was set to 100. If there are less features in the image, new features were created on the surface and the associated pixel coordinates were determined.

After this implementation errors in the measurements were considered. These errors consist of pixel noise and the loss of a few random features.

### 3.2. The Feature Tracker Update

The measurements of the feature tracker were not directly used in the EKF. They were first converted into an optical flow measurement. Through this conversion, the measurement model is given as follows:

$$h(x) = \left( \frac{v^{CAM} \times \varepsilon^{CAM}}{D} - \omega^{CAM} \right) \times \varepsilon^{CAM}$$

Where:

$h(x)$ = the real optical flow

$v^{CAM}$ = the velocity in the camera frame

$\varepsilon^{CAM}$ = the feature direction vector in the camera frame

$\omega^{CAM}$ = the rotation rate of the camera in the camera frame

$D$ = distance of the camera to the feature

The update function of the EKF needs the following 3 matrices:

- $z$ = the measurement error in the prediction
- $R$ = the covariance matrix of the measurement noise
- $H$ = the observation matrix

$z$ is defined through:

$$z = \hat{\Omega} - \tilde{\Omega}$$

Where the predicted optical flow is:

$$\hat{\Omega} = h(\hat{x})$$

The calculation of the measured optical flow is different and defined as:

$$\tilde{\Omega} = \begin{bmatrix} \varphi \\ \theta \end{bmatrix}$$

Where $\varphi$ and $\theta$ are the result of the transformation from $\tilde{V}^{CAM}$ into the spherical coordinate system. The calculation of $\tilde{V}^{CAM}$ is shown below:

$$\tilde{V}^{CAM} = \tilde{\omega}_t \times \tilde{\varepsilon}_t$$

where

$$\tilde{\omega}_t = \tilde{\omega}_t * \tilde{\omega}_t^*$$

$$\tilde{\omega}_t = \frac{a\cos(\tilde{\varepsilon}_{t-1} \times \tilde{\varepsilon}_{t+1})}{\Delta(t_{t-1} - t_{t+1})}$$

$$\tilde{\omega}_t^* = \tilde{\varepsilon}_{t-1} \times \tilde{\varepsilon}_{t+1}.$$

Both optical flow calculations are done with 10 features, those which are closest to the center of the image at the time t. This criterion is important, because only the distance to the center of the image is known through the LA measurement. Another criterion is that only features could be used which exist in the images at time t-1, t and t+1. All three images and features are needed for the calculation. In the real world the navigation system must wait until all three images are available before doing this calculation.

The covariance matrix of the measurement noise includes not only the noise factor, but also the unknown surface detail error. A feature which is further away from the center of the image becomes less accurate.

$H$ was calculated numerically instead of finding a closed form solution because the function is very complex. The observation matrix is calculated using:

$$H = \frac{h(x+a) - h(x-a)}{2 * a}$$

Where $a$ is a small change in $x$.

## 4. THE TRADEOFF

In the tradeoff two different things were analyzed. At first, the influence of the field of view of the navigation camera was analyzed. For this reason, all parameters remain constant, except the field of view of the navigation camera. It was changed from 40 to 80 degrees. This test shows that the field of view has very little influence on the navigation result. The only thing we noticed was a slightly more accurate attitude in case of smaller field of view and a slightly better position with the larger field of view. However, the difference was so small that the influence is hardly notable.

The second part of the tradeoff has the purpose to analyze the influence of the frequencies of the sensors on the navigation results. Table 1shows how the frequencies where changed during the Monte-Carlo simulation.

The baseline values are the same as those listed in section 2. The results of the baseline Monte-Carlo Simulation are shown in Figure 2 and Figure 3.

**Table 1: Tradeoff parameters.**

| Sensor | Lowest Frequency | Highest Frequency |
|---|---|---|
| STR | 0.1 Hz | 1 Hz |
| LA | 0.1 Hz | 2 Hz |
| FT | 4 Hz | 10 Hz |
| CN | 0.1 Hz | 1 Hz |

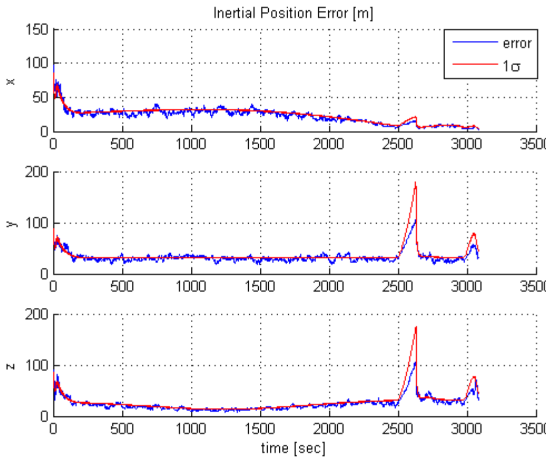


**Figure 2: Baseline position error plot**

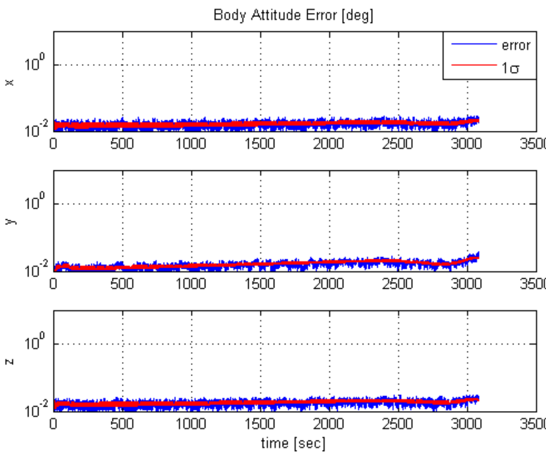


**Figure 3: Baseline attitude error plot**

Figure 2 shows the position error over the time. The whole plot shows an error with less than 100 meters except for the time interval between 2550 and 2650 seconds. The reason for this is that the crater navigation algorithm is turned off below an altitude from 10 km. This altitude is reached at the time at 2500 seconds. The second reason is the rotation maneuver at this time. Through this maneuver, the feature tracker measurements are very inaccurate due to the strong movement of the features on the image and the linearization error in the optical flow calculation. This is the reason, why the frequency of the feature tracker cannot decrease even more. If the frequency of the feature tracker were decreased, this peak would grow. To reduce the peak, the frequency of the feature tracker must be increased considerably. Even at twice the frequency, the peak hardly changes size.

The Figure 3 shows the attitude error over the time. Due to the star tracker, which has a very accurate attitude measurement, the error is very low.

Figure 4 and Figure 5 show the results of the Monte-Carlo Simulation with only the lowest frequencies.

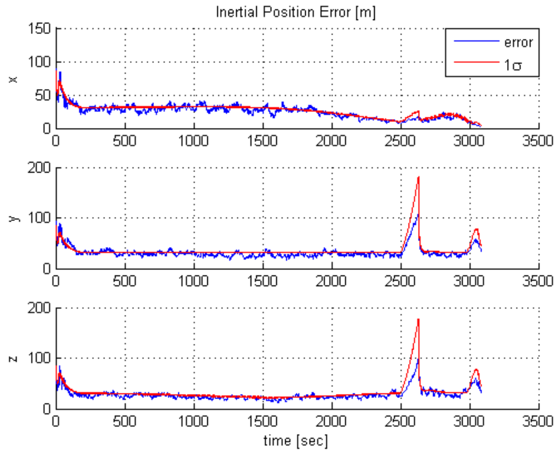


**Figure 4: Worst case position error plot**

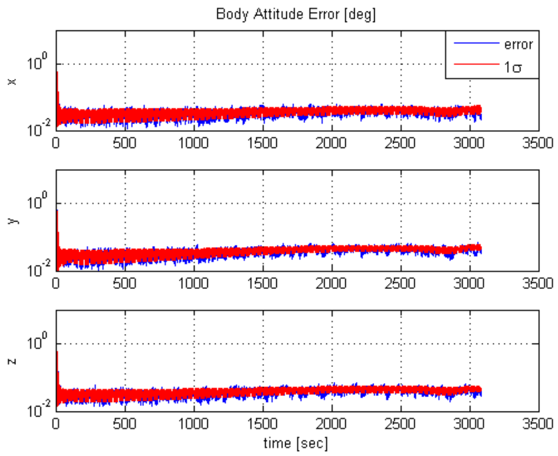


**Figure 5: Worst case attitude error plot**

By comparing these plots with the baseline results, the only difference which is really notable is the jump in the attitude error plot. This "jump" is created by the low frequency of the star tracker, which corrects the attitude every 10 seconds and in between, the attitude error increases.

## 5. CONCLUSION

The presented navigation filter and the tradeoff results show us an accurate navigation solution, even at lower frequencies. This shows that the frequencies or the quality of a single sensor can be decreased to get an even smaller navigation system without losing much in terms of accuracy. Another important result is that the feature tracker needs a certain frequency to work correctly because of the linearization error in the EKF update calculation. If possible, the frequency of the feature tracker can be turned to the current vehicle dynamics such that this linearization error is reduced.

## 6. REFERENCES


The sensor models were taken from the High Performance Satellite Dynamics Simulator (HPS), which is a project from Zentrum für angewandte Raumfahrttechnologie und Mikrogravitation (ZARM) and Deutsches Zentrum für Luft- und Raumfahrt (DLR).

[1]: B. Maass, H.Krüger and S.Theil, "An Edge-Free, Scale-, Pose- and Illumination-Invariant Approach to Crater Detection for Spacecraft Navigation", 7th International Symposium on Image and Signal Processing and Analysis, pp. 603-608, 2011

[2]: S. Steffes "Real-Time Navigation Algorithm for the SHEFEX2 Hybrid Navigation System Experiment", AIAA Guidance, Navigation and Control Conference 2012

[3] D. Heise, "Entwicklung eines optischen Navigationssystems für Explorationsmissionen", Master Thesis, Universität Bremen, 2012

[4] I. Pelivan, A. Heidecker and S. Theil, „High performance satellite dynamics and control simulation for multipurpose application", Journal of Aerospace Engineering, Sciences and Applications, pp. 119-130, 2012