Local Reference Filter for Life-Long Vision Aided Inertial Navigation

Korbinian Schmid
Department of Perception and Cognition
Robotics and Mechatronics Center
German Aerospace Center (DLR)
Email: Korbinian.Schmid@dlr.de

Felix Ruess
Department of Perception and Cognition
Robotics and Mechatronics Center
German Aerospace Center (DLR)
Email: Felix.Ruess@dlr.de

Darius Burschka
Institut for Computer Science VI
Technical University of Munich (TUM)
Email: burschka@in.tum.de

***Abstract*—Filter based system state estimation is widely used for hard-realtime applications. In long-term filter operation the estimation of unobservable system states can lead to numerical instability due to unbounded state uncertainties. We introduce a filter concept that estimates system states in respect to changing local references instead of one global reference. In this way unbounded state covariances can be reset in a consistent way. We show how local reference (LR) filtering can be integrated into filter prediction to be used in square root filter implementations. The concept of LR-filtering is applied to the problem of vision aided inertial navigation (LR-INS). The results of a simulated 24 h quadrotor flight using the LR-INS demonstrate long-term filter stability. Real quadrotor flight experiments show the usability of the LR-INS for a highly dynamic system with limited computational resources.**

I. Introduction

System state estimation for control requires sensor data fusion in hard realtime. For this purpose, probabilistic filters are often used due their simplicity, low computational complexity and deterministic timing behavior. To guarantee long-term stable state estimation a numerically robust filter implementation as well as full system state observability are fundamental. While numerically stable algorithms as for example square root filters are well established, a state estimation formulation with full observability can not always be guaranteed. This situation is critical in two aspects: firstly, unbounded filter covariances can cause numerical instability. Secondly, linearization of non-linear systems often assumes small state errors. If the errors rise unbounded the filter can become inconsistent. An example of an unobservable set of states are yaw and position estimates in Vision Aided Inertial Navigation Systems (VO-INS) [4].

In the recent years, the development of VO-INS showed great progress. Mourikis et al. [6] demonstrated a hard-realtime capable mono vision/IMU fusion algorithm using an Extended Kalman Filter (EKF). In their aproach a certain window over past poses is kept within the filter state vector to process feature measurements taken from different locations along the traveled trajectory. Using limited data windows makes realtime implementation possible but turns the system into an odometry system as trajectory loop closures can not be integrated.

Kaess et al. [2] combined filtering and smoothing to overcome this limitation. A separator state on top of the smoothing problem representing Bayes tree is used as interface between a global smoother and a local filter. In a synchronization step, updates on the separator are exchanged between filter and smoother. The current separator is continuously marginalized out of the filter to transfer states from the filter to the smoother and use a new separator state. Even though, the increase of globally referenced state uncertainty can be drastically reduced in the case of trajectory loop closures, it is still globally unbounded.

An approach to tackle the problem of bad scalability and global inconsistency of EKF-SLAM is sub-mapping [10]. Local sub-maps, with arbitrary origins, are combined within a global EKF. The demonstrated reference transformation of features can be adapted to realize state transformation for vision aided inertial navigation and locally limit the uncertainty increase of unobservable states. In our paper we generalize this idea and introduce a technique to overcome the general problem of unbounded filter covariances for unobservable filter states. Further, we consider hard-realtime constraints.

In our local reference filter a transformation function between the unobservable states and a local reference is defined while the local reference is augmented to the filter state vector. The filter prediction step is used to switch the filter states and its covariance to the new local reference which is marginalized out at the same time. We demonstrate the concept on a vision-aided inertial navigation filter. We sporadically change the filter reference frame to a new frame with a lower relative uncertainty compared to the current system state. All filter states and covariances are transformed to the new local reference frame which can be a node of any (non-realtime) high-level navigation system either topological or metric (as for example from a SLAM backend). In this way we separate local realtime state estimation from global navigation and relax timing constraints on the latter one. The implementation is realized as square root UD filter to improve numerical stability on embedded computers with limited floating point precision. The main contributions of this paper are:

- a Local Reference Square Root filter concept (LR-filter) to realize long-term stable state estimation including unobservable states

- the application of the LR-Filter concept to inertial navigation (LR-INS)
- a mechanism to combine global (topological or metric) navigation with long term stable, local, metric state estimation with hard-realtime constraints

This paper is structured as follows: in Section II we explain the general concept of the LR-filter and its implementation in square root form. In Section III a Local Reference Inertial Navigation System (LR-INS) is developed. In Section IV we prove long-term filter stability in a simulated 24 h quadrotor flight experiment and present experimental results of real quadrotor flights employing the LR-INS. We discuss results and limitations of the introduced LR-Filter in Section V to conclude the paper in Section VI.

II. Local Reference Filtering

In the LR-filter, the global state reference is transformed to a local reference to limit the unbounded increase of filter state covariances for unobservable system states. In terms of filtering, this includes state augmentation, marginalization and transformation. These operations can be included into a modified filter prediction step which we will derive in the first part of this Section. We give a short overview of the Square Root UD Filter and apply the modified Local Reference Filter prediction to Square Root filters. By including all operations into the square root UD prediction step, the covariance matrix factorization and its numerically superior properties can be kept at any time.

A. State Augmentation, Marginalization and Reference switching within Prediction

To change the reference of filter states and its corresponding covariance matrix a new local reference has to be defined including at least states corresponding to the unobservable system states. This local reference can be a partial clone of the current system state optionally combined with sensor measurements. At time k, we augment the local reference to the current state vector $\boldsymbol{x}_k$ by:

$$\bar{\boldsymbol{x}}_k = \begin{pmatrix} \boldsymbol{x}_k \\ \boldsymbol{x}_{aug} \end{pmatrix} = \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{z}_k) \tag{1}$$

where $\boldsymbol{z}_k$ is a sensor measurement disturbed by Additive White Gaussian Noise (AWGN) with covariance $\boldsymbol{R}_k$. The filter covariance for the new state vector is calculated using the Jacobian of $\boldsymbol{g}$:

$$\begin{aligned} \bar{\boldsymbol{P}}_{\boldsymbol{k}} &= \frac{\partial \bar{\boldsymbol{x}}_k}{\partial \boldsymbol{x}_k} \boldsymbol{P}_k \frac{\partial \bar{\boldsymbol{x}}_k}{\partial \boldsymbol{x}_k}^T + \frac{\partial \bar{\boldsymbol{x}}_k}{\partial \boldsymbol{z}_k} \boldsymbol{R}_k \frac{\partial \bar{\boldsymbol{x}}_k}{\partial \boldsymbol{z}_k}^T \\ &= \boldsymbol{A}_k \boldsymbol{P} \boldsymbol{A}_k^T + \boldsymbol{T}_k \boldsymbol{R}_k \boldsymbol{T}_k^T \end{aligned} \tag{2}$$

where $\boldsymbol{A}_k$ is the augmentation matrix, $\boldsymbol{T}_k$ is a noise transformation matrix. Using stochastic cloning [7], the augmentation matrix has exactly one $\mathbf{1}$ per row and the noise transformation matrix is the zero matrix.

We can apply a regular prediction step to the augmented state which results for the covariance prediction in:

$$\bar{\boldsymbol{P}}_{k+1} = \boldsymbol{\Phi}_k \boldsymbol{A}_k \boldsymbol{P}_k \boldsymbol{A}_k^T \boldsymbol{\Phi}_k^T + (\boldsymbol{G}_k \boldsymbol{Q}_k \boldsymbol{G}_k^T + \boldsymbol{T}_k \boldsymbol{R}_k \boldsymbol{T}_k^T) \tag{3}$$

where the augmented system matrix $\boldsymbol{\Phi}_k$ is an identity matrix of corresponding size with the original system matrix in the upper left corner. The augmented noise propagation matrix $\boldsymbol{G}_k$ is a zero matrix of corresponding size with the original noise propagation matrix in the top rows.

Analogously to Equation 1, we can define a transformation function $\boldsymbol{f}$ that transforms the system states at time $k+1$ (including all augmentations) into the new reference frame defined by the augmented state and, at the same time, removes the augmented reference state from the filter:

$$\bar{\bar{\boldsymbol{x}}}_{k+1} = \boldsymbol{f}(\bar{\boldsymbol{x}}_{k+1}) \tag{4}$$

Using the Jacobian of $\boldsymbol{f}(\cdot)$ the transformed state covariance can be calculated as:

$$\begin{aligned} \bar{\bar{\boldsymbol{P}}}_{k+1} =& \frac{\partial \bar{\bar{\boldsymbol{x}}}_{k+1}}{\partial \bar{\boldsymbol{x}}_{k+1}} \bar{\boldsymbol{P}}_{k+1} \frac{\partial \bar{\bar{\boldsymbol{x}}}_{k+1}}{\partial \bar{\boldsymbol{x}}_{k+1}}^T = \\ =& \boldsymbol{S}_k \boldsymbol{\Phi}_k \boldsymbol{A}_k \boldsymbol{P}_k \boldsymbol{A}_k^T \boldsymbol{\Phi}_k^T \boldsymbol{S}_k^T \\ &+ \boldsymbol{S}_k (\boldsymbol{G}_k \boldsymbol{Q}_k \boldsymbol{G}_k^T + \boldsymbol{T}_k \boldsymbol{R}_k \boldsymbol{T}_k^T) \boldsymbol{S}_k^T \end{aligned} \tag{5}$$

Equation 5 can be rewritten as:

$$\bar{\bar{\boldsymbol{P}}}_{k+1} = \tilde{\boldsymbol{\Phi}} \boldsymbol{P}_k \tilde{\boldsymbol{\Phi}}^T + \tilde{\boldsymbol{G}} \tilde{\boldsymbol{Q}} \tilde{\boldsymbol{G}}^T \tag{6}$$

which is exactly the form of a filter prediction step except for the (potentially) non quadratic form of the system matrix $\tilde{\boldsymbol{\Phi}}$. We will exploit this similarity in Section II-C to integrate state augmentation, propagation, marginalization and reference switching into the square root UD prediction algorithm.

Even though all operations for local filtering can be carried out within one prediction step, they do not have to be executed at the same time. For example, a reference switch can be executed at any time after the augmentation of a potential reference. Nevertheless, reference switching can be realized without the risk of covariance rank deficiencies due to state cloning if the formulation from Equation 5 is used. Otherwise, state cloning may create a problem for square root filter propagation algorithms.

B. Square Root UD filter

Often only single precision FPUs are available on small embedded computers. As per Maybeck [5], the naive implementation of Kalman filters inherently involves unstable numerics. Square root filters have vastly superior numerical properties. By factorization of the covariance matrix, symmetry and positive definiteness are implicitly guaranteed. Implemented in single precision they are at least as precise as a naive implementation in double precision for a modest increase in computational load. State cloning, which brings the covariance matrix close to a singular state, can be critical especially in a naive implementation. A numerically stable

implementation should be preferred. Therefore, we shortly recap the Square Root UD filter algorithm. The square root UD-Filter (SRUD) developed by Bierman and Thornton [1] uses a matrix factorization of the filter covariance matrix in the form $\boldsymbol{P} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{U}^T$ where $\boldsymbol{U}$ is a strictly upper triangular matrix and $\boldsymbol{D}$ a diagonal matrix. As per Maybeck [5], in terms of numerical stability and complexity, the SRUD filter is comparable to square root filters using Cholesky factorization but without the need for calculating actual square roots. The introduced method can be easily applied to both formulations. To stay consistent with the original SRUD publication, we keep the formulation with a strictly upper triangular matrix $\boldsymbol{U}$ instead of a strictly lower triangular matrix $\boldsymbol{L}$ as in the equivalent and more common LDLT decomposition.

Under the assumption of a diagonal noise matrix Q, without loss of generality, the filter prediction step

$$\boldsymbol{U}_{k+1}\boldsymbol{D}_{k+1}\boldsymbol{U}_{k+1}^T = \boldsymbol{\Phi}_k\boldsymbol{U}_k\boldsymbol{D}_k\boldsymbol{U}_k^T\boldsymbol{\Phi}_k^T + \boldsymbol{G}_k\boldsymbol{Q}_k\boldsymbol{G}_k^T \tag{7}$$

can be reformulated as

$$\begin{aligned} &\boldsymbol{Y}\boldsymbol{D}\boldsymbol{Y}^T \\ &\boldsymbol{Y} = \begin{pmatrix}\boldsymbol{\Phi}_k\boldsymbol{U}_k & \boldsymbol{G}_k\end{pmatrix}, \quad \boldsymbol{D} = diag(\boldsymbol{D}_k, \boldsymbol{Q}_k) \end{aligned} \tag{8}$$

The prediction step in $\boldsymbol{U}\boldsymbol{D}\boldsymbol{U}^T$ form is realized by triangularization of $\boldsymbol{Y}$ by the Modified Weighted Gram-Schmidt (MWGS) algorthim or by weighted Givens rotations. The resulting quadratic, strictly upper diagonal part of $\boldsymbol{Y}$ corresponds to $\boldsymbol{U}_{k+1}$. The corresponding $\boldsymbol{D}_{k+1}$ matrix results from the triangularization process.

Filter updates are realized in two steps: first measurements are decorrelated by diagonalizing the measurement noise matrix $\boldsymbol{R}$. Then, several scalar updates are carried out via modified Cholesky rank one downdates.

C. Local Reference Square Root UD Filter

State augmentation and reference switching can be implicitly realized in UD-form without de- and refactoring of the covariance matrix. In Section II-A, we showed that filter state augmentation and reference switching can be seen as covariance prediction in a naive filter implementation. We use this formulation as basis for implicit state augmentation and reference switching for a UD factorized covariance matrix. Substituting the covariance matrix $\boldsymbol{P}$ in Equation 6 by its $\boldsymbol{U}\boldsymbol{D}\boldsymbol{U}^T$ factorization leads to the prediction step

$$\boldsymbol{U}_{k+1}\boldsymbol{D}_{k+1}\boldsymbol{U}_{k+1}^T = \tilde{\boldsymbol{\Phi}}\boldsymbol{U}_k\boldsymbol{D}_k\boldsymbol{U}_k^T\tilde{\boldsymbol{\Phi}}^T + \tilde{\boldsymbol{G}}\boldsymbol{Q}_k\tilde{\boldsymbol{G}} \tag{9}$$

which is equivalent to the regular SRUD prediction Equation 7 except for the non quadratic form of the system matrix $\tilde{\boldsymbol{\Phi}}$. At this point the diaginal matrix $\boldsymbol{D}_k$ still has the size of the old state vector before augmentation or marginalization. Applying triangularization on $\boldsymbol{Y}$ of Equation 8 implicitly adapts the size of $\boldsymbol{D}_{k+1}$ corresponding to the new state vector size. Nevertheless, triangularization might fail if the resulting covariance matrix is singular, depending on the used algorithm. Stochastic cloning always introduces a covariance matrix rank deficiency directly after cloning. By combining augmentation and prediction the matrix rank can be filled up by the system noise within one prediction step. Therefore, special care has to be taken to choose a suitable discrete approximation of the continuous prediction noise term $\boldsymbol{G}(t)\boldsymbol{Q}(t)\boldsymbol{G}(t)^T$.

III. Local Reference Inertial Navigation System

In this Section we apply the concept of LR-Filtering to vision aided inertial navigation. We fuse odometry measurements of a stereo camera system with acceleration and angular rate measurements of an IMU. Furthermore, we assume the sporadical availability of 6D-landmark measurements. The pose of the landmark is included into the state vector to be used as new local reference the filter states can be switched into.

A. Vision Based Keyframe Inertial Navigation

Our inertial navigation system, previously introduced in [9], is based on an error state space Extended Kalman Filter in feedback configuration. The current kinematic and dynamic states are calculated by the computationally cheap strap down algorithm (SDA) at frequency f_{SDA}. The frequency is chosen corresponding to the expected system dynamics. The filter is running at frequency $f_{EKF} \leq f_{SDA}$ calculating the error covariances of the system states. The result of the filter update step is used for state correction whenever a measurement is available. We define the direct system main state and the corresponding indirect (filter) main state as:

$$\boldsymbol{x} = \left(\boldsymbol{p}_{N_xB}^{N_x,T}\ \boldsymbol{v}_{N_xB}^{N_x,T}\ \boldsymbol{q}_B^{N_x,T}\ \boldsymbol{b}_a^T\ \boldsymbol{b}_\omega^T\right)^T \in \mathbb{R}^{16} \tag{10}$$

$$\boldsymbol{\delta} = \left(\delta\boldsymbol{p}^T\ \delta\boldsymbol{v}^T\ \delta\boldsymbol{\sigma}^T\ \delta\boldsymbol{b}_a^T\ \delta\boldsymbol{b}_\omega^T\right)^T \in \mathbb{R}^{15} \tag{11}$$

where the direct state includes the body position, velocity and orientation quaternion relative to the navigation frame N_x and IMU accelerometer and gyroscope biases. The indirect main state includes the corresponding errors of the direct state (with angle errors expressed in Euler representation).

For the inertial navigation system error propagation we employ the following linearized, continuous-time error transition model as for example derived by Wendel [11]:

$$\begin{aligned} \dot{\boldsymbol{\delta}} &= \mathbf{F}\,\boldsymbol{\delta} + \mathbf{G}\,\boldsymbol{n} \\ &= \begin{pmatrix} \mathbf{O}_{3x3} & \mathbf{I}_3 & \mathbf{O}_{3x3} & \mathbf{O}_{3x3} & \mathbf{O}_{3x3} \\ \mathbf{O}_{3x3} & \mathbf{O}_{3x3} & -\left\lfloor \boldsymbol{a}_{N_xB}^{N_x} \right\rfloor & -\mathbf{C}_B^{N_x} & \mathbf{O}_{3x3} \\ \mathbf{O}_{3x3} & \mathbf{O}_{3x3} & \mathbf{O}_{3x3} & \mathbf{O}_{3x3} & -\mathbf{C}_B^{N_x} \\ \mathbf{O}_{3x3} & \mathbf{O}_{3x3} & \mathbf{O}_{3x3} & \mathbf{O}_{3x3} & \mathbf{O}_{3x3} \\ \mathbf{O}_{3x3} & \mathbf{O}_{3x3} & \mathbf{O}_{3x3} & \mathbf{O}_{3x3} & \mathbf{O}_{3x3} \end{pmatrix} \boldsymbol{\delta} \\ &+ \begin{pmatrix} \mathbf{O}_{3x3} & \mathbf{O}_{3x3} & \mathbf{O}_{3x3} & \mathbf{O}_{3x3} \\ \mathbf{C}_B^{N_x} & \mathbf{O}_{3x3} & \mathbf{O}_{3x3} & \mathbf{O}_{3x3} \\ \mathbf{O}_{3x3} & \mathbf{C}_B^{N_x} & \mathbf{O}_{3x3} & \mathbf{O}_{3x3} \\ \mathbf{O}_{3x3} & \mathbf{O}_{3x3} & \mathbf{I}_3 & \mathbf{O}_{3x3} \\ \mathbf{O}_{3x3} & \mathbf{O}_{3x3} & \mathbf{O}_{3x3} & \mathbf{I}_3 \end{pmatrix} \boldsymbol{n} \end{aligned} \tag{12}$$

where $\boldsymbol{a}_{N_xB}^{N_x}$ is the specific force (body acceleration relative to the current navigation frame N_x expressed in N_x), $\lfloor \ldots \rfloor$ the skew symmetric matrix operator of a vector and $\mathbf{C}_B^{N_x}$ the rotation matrix transforming a vector expressed in the body

frame to the frame N_x. The uncertainties in the error propagation for translation and rotation are modeled as additive zero-mean, white Gaussian noise (AWGN). The accelerometer and gyroscope biases are modeled as random walk processes driven by AWGN. The noise vector $\boldsymbol{n_s}$ has the spectral density $\boldsymbol{Q}$, such that

$$\mathbf{Q} = \mathrm{diag}(\mathbf{Q}_{\boldsymbol{a}}, \mathbf{Q}_{\boldsymbol{\omega}}, \mathbf{Q}_{\boldsymbol{b_a}}, \mathbf{Q}_{\boldsymbol{b_\omega}}) \tag{13}$$

$$\mathbf{Q}_{\boldsymbol{s}} = \boldsymbol{E}\left[\boldsymbol{n_s} \boldsymbol{n_s}^T\right] \mid \boldsymbol{s} \in \{\boldsymbol{a}, \boldsymbol{\omega}, \boldsymbol{b_a}, \boldsymbol{b_\omega}\} \ . \tag{14}$$

The system matrix $\boldsymbol{F}$ is discretized at time step k for the filter time interval $T = 1/f_{EKF}$ as $\boldsymbol{\Phi}_k = e^{\boldsymbol{\Phi} T}$.

Our visual odometry algorithm provides a transformation measurement from a keyframe to the last captured image with the according measurement noise. The algorithm choses the keyframe with the smallest accumulated measurement noise as reference. In this way, drift can be avoided while moving in a small area or while standing, compared to frame to frame odometry. To process these delta pose measurements and compensate for measurement delays introduced by the vision pipeline, we clone the robot pose at the time of image capturing. Whenever a new image is captured by the stereo cameras, we register the hardware camera synchronization trigger to instantaneously initiate pose state cloning. Analogously to Equation 1 we define the augmented state vector as:

$$\begin{aligned} \bar{\boldsymbol{x}}_k &= \begin{pmatrix} \boldsymbol{x}_k \\ \boldsymbol{x}_{\boldsymbol{p},\boldsymbol{\sigma}} \end{pmatrix} = \boldsymbol{g}_x(\boldsymbol{x}_k, \tilde{\boldsymbol{z}}_k) \\ \bar{\boldsymbol{\delta}}_k &= \begin{pmatrix} \boldsymbol{\delta}_k \\ \boldsymbol{\delta}_{p,\sigma} \end{pmatrix} = \boldsymbol{g}_\delta(\boldsymbol{\delta}_k, \tilde{\boldsymbol{z}}_k) \end{aligned} \tag{15}$$

where $\boldsymbol{x}_{\boldsymbol{p},\boldsymbol{\sigma}} = (\boldsymbol{p}_{N_x B}^{N_x,T} \ \ \boldsymbol{q}_B^{N_x,T})^T$ and $\boldsymbol{\delta}_{p,\sigma}^T = (\boldsymbol{\delta}_p^T \ \ \boldsymbol{\delta}_\sigma^T)^T$ are pose and pose error, respectively, at time k. As no measurements are involved the noise transformation matrix $\boldsymbol{T}_k$ of Equation 2 is the zero matrix while $\boldsymbol{A}_k = \frac{\partial \boldsymbol{g}_\delta}{\partial \boldsymbol{\delta}_k}$. Further details on the keyframe based VO-INS, including corresponding measurement Equations, can be found in [3].

B. Local Reference Augmentation

Similarly to visual odometry measurements, we employ state cloning for the processing of 6D landmark measurements of a camera. We clone the current robot pose as frame $\boldsymbol{B}_z$ at the exact time of the camera hardware trigger measuring a potential reference frame N_{x+1} (PRF). At the time of arrival of the measurement the PRF state is augmented in the filter using its measurement noise $\boldsymbol{R}$ for initialization. This augmentation is similar to classical EKF-SLAM feature augmentation except for two differences: first, we use an error state space filter formulation. Second, the robot measurement pose and PRF augmentation can occur at different times. These measurement time delays are implicitly compensated by state cloning at the exact hardware trigger time of image capturing and referencing the corresponding augmented robot pose in the filter update step. The PRF pose augmentation measurement, corresponding to Equation 1, is expressed in the current navigation frame N_x as

$$\begin{aligned} \boldsymbol{g}_{x,N_{x+1}}(\boldsymbol{x}, \tilde{\boldsymbol{z}}_{N_{x+1}}) = &\begin{pmatrix} \boldsymbol{\delta}_{N_x N_{x+1}}^{N_x} \\ \boldsymbol{\delta}_{N_{x+1}}^{N_x} \end{pmatrix} \\ &\begin{pmatrix} \boldsymbol{p}_{N_x B_z}^{N_x} + \boldsymbol{C}_{B_z}^{N_x}(\boldsymbol{C}_C^{B_z} \tilde{\boldsymbol{p}}_{CN_{x+1}}^C + \boldsymbol{p}_{BC}^B) \\ \boldsymbol{q}_{B_z}^{N_x} \boldsymbol{q}_C^B \tilde{\boldsymbol{q}}_{N_{x+1}}^C \end{pmatrix} \end{aligned} \tag{16}$$

where $\boldsymbol{p}_{N_x B_z}^{N_x}$ is the augmented estimated robot pose at time of image capturing, $\tilde{\boldsymbol{p}}_{CN_{x+1}}^C$ the PRF position in the camera frame, $\boldsymbol{p}_{BC}^B$ the known translation vector between IMU frame and camera, $\boldsymbol{C}_B^{N_x}$ the estimated rotation matrix between navigation and IMU frame, $\boldsymbol{q}_C^B$ and $\boldsymbol{C}_C^B$ the known camera orientation between IMU and camera frame as quaternion and rotation matrix respectively and $\tilde{\boldsymbol{q}}_{N_{x+1}}^C$ the PRF orientation measurement quaternion. From Equation 16 we can derive $\boldsymbol{g}_{\delta,N_{x+1}}$ and find the relevant parts of the noise augmentation and noise transformation matrices referencing the partial error state $\boldsymbol{\delta}_{p,\sigma}^T$ augmented at the time of image capturing as:

$$\begin{aligned} \boldsymbol{A}_{\boldsymbol{\delta}_{p,\sigma}} &= \begin{pmatrix} \boldsymbol{I}_{3x3} & -\left\lfloor \boldsymbol{C}_{B_z}^{N_x}(\boldsymbol{C}_C^{B_z} \tilde{\boldsymbol{p}}_{N_{x+1}} + \boldsymbol{p}_C^{B_z}) \right\rfloor \\ \boldsymbol{0}_{3x3} & \boldsymbol{I}_{3x3} \end{pmatrix} \\ \boldsymbol{T}_{\boldsymbol{\delta}_{p,\sigma},k} &= \begin{pmatrix} \boldsymbol{C}_c^{N_x} & \boldsymbol{0}_{3x3} \\ \boldsymbol{0}_{3x3} & \boldsymbol{C}_c^{N_x} \end{pmatrix} \end{aligned} \tag{17}$$

After augmentation of the PRF further PRF measurements can be processed in the filter. Therefore, we repeat state cloning at the exact camera measurement trigger time to save the new body frame B_z and process the (delayed) arriving measurement by the linearized PRF error measurement equation expressed in the camera frame C_z with $\boldsymbol{R}_{N_x}^{C_z} = \boldsymbol{R}_B^C \boldsymbol{R}_{N_x}^{B_z}$

$$\begin{aligned} \boldsymbol{z}_\delta &= \boldsymbol{\Xi} \begin{pmatrix} \boldsymbol{\delta}_{\boldsymbol{p},\boldsymbol{\sigma}} \\ \boldsymbol{\delta}_{N_{x+1}} \end{pmatrix} + \boldsymbol{n}_{N_{x+1}} \\ \boldsymbol{\Xi} &= \begin{pmatrix} -\boldsymbol{R}_{N_x}^{C_z} & \boldsymbol{R}_{N_x}^{C_z} \left\lfloor \boldsymbol{p}_{N_x B}^{N_x} - \boldsymbol{p}_{N_x N_{x+1}}^{N_x} \right\rfloor & \boldsymbol{R}_{N_x}^{C_z} & \boldsymbol{0}_{3x3} \\ \boldsymbol{0}_{3x3} & -\boldsymbol{R}_{N_x}^{C_z} & \boldsymbol{0}_{3x3} & \boldsymbol{R}_{N_x}^{C_z} \end{pmatrix} \end{aligned} \tag{18}$$

C. Navigation Frame Switching

Position and yaw relative to the initial reference frame are unobservable. Therefore, we switch the filter reference to an augmented PRF. All states relative to the navigation frame, which are poses and velocities, have to be transformed into the new reference frame N_{x+1}. We know the PRF estimate relative to the current reference frame N_x by the translation vector $\boldsymbol{p}_{N_x N_{x+1}}^{N_x}$ and the rotation as matrix $\boldsymbol{C}_{N_x}^{N_{x+1}}$ and corresponding quaternion $\boldsymbol{q}_{N_x}^{N_{x+1}}$.

In analogy to Equation 4, we can formulate reference switching as function on the current state vectors. Direct state positions, velocities and orientations are transformed as

$$\begin{aligned} \boldsymbol{p}^{N_x} &= \boldsymbol{C}_{N_x}^{N_{x+1}}(\boldsymbol{p}^{N_x} - \boldsymbol{p}_{N_x N_{x+1}}^{N_x}) = \boldsymbol{C}_{N_x}^{N_{x+1}} \Delta \boldsymbol{p}^{N_x} \\ \boldsymbol{v}_{N_x B}^{N_{x+1}} &= \boldsymbol{C}_{N_x}^{N_{x+1}} \boldsymbol{v}_{N_x B}^{N_x} \\ \boldsymbol{q}^{N_{x+1}} &= \boldsymbol{q}_{N_x}^{N_{x+1}} \boldsymbol{q}^{N_x} \end{aligned} \tag{19}$$

The corresponding linearized transformations in tangent space can be derived as:

$$\begin{aligned}\boldsymbol{\delta}_{\boldsymbol{p}}^{N_{x+1}} &= \boldsymbol{C}_{N_x}^{N_{x+1}}(\boldsymbol{\delta}_{p}^{N_x} - \boldsymbol{\delta}_{N_x N_{x+1}}^{N_x} + \left\lfloor \Delta \boldsymbol{p}^{N_x} \right\rfloor \boldsymbol{\delta}_{\boldsymbol{\sigma},N_{x+1}}^{N_x}) \\ \boldsymbol{\delta}_{\boldsymbol{v}}^{N_{x+1}} &= \boldsymbol{C}_{N_x}^{N_{x+1}}(\boldsymbol{\delta}_{\boldsymbol{v},N_x B}^{N_x} + \left\lfloor \hat{\boldsymbol{v}}_{N_x B}^{N_x} \right\rfloor \boldsymbol{\delta}_{\boldsymbol{\sigma},N_{x+1}}^{N_x}) \\ \boldsymbol{\delta}_{\boldsymbol{\sigma}}^{N_{x+1}} &= \boldsymbol{C}_{N_x}^{N_1}(\boldsymbol{\delta}_{\boldsymbol{\sigma}}^{N_x} - \boldsymbol{\delta}_{N_{x+1}}^{N_x})\end{aligned} \tag{20}$$

In the same transformation, we marginalize out the PRF state. After filter switching the PRF measurement of Equation 18 becomes an absolute pose measurement.

In some situations a full reference switch is not practical. For UAV navigation, for example, the navigation frame x- and y-axes should be orthogonal to the gravity vector. This can be realized by manipulating the rotation matrix $\boldsymbol{C}_{N_x}^{N_{x+1}}$ in a way that only a rotation about the gravity vector is applied. Even the identity matrix can be used as transformation matrix defining the current orientation as the new reference orientation. Nevertheless, using the latter alternative yaw drift is not bound whereas using the former the yaw reference is readapted to a local but static reference frame. We employ option one in the following experiments.

IV. Experiments

We demonstrate the concept of our LR-INS in simulations and in quadrotor flight experiments. In both situations we switch off the keyframe feature of our visual odometry system to accelerate the increase of covariances for unobservable states. The resulting frame to frame odometry is used on many vision based UAVs and can be compared to a configuration with velocity measurements from optical flow sensors.

A. Simulated UAV flight

We show the long-term stability of the LR-INS in a simulated 24 h quadrotor flight. In our simulation environment, we defined four distinct landmarks. The quadrotor is randomly commanded to the landmark locations. With the quadrotor closer than two meters a noisy PRF measurement expressed in the virtual camera frame is simulated. Furthermore, IMU measurements and delta pose measurements (also in virtual camera frame) are simulated during the entire flight time. All sensor measurements are disturbed by zero mean AWGN. Table I lists the corresponding simulation parameters. The same parameters are used within the filter. Considering the simulated visual odometry sensor we choose rather conservative frequency and noise parameters compared to our real implementation to accelerate the increase of state uncertainty and emphasize the effect of reference switching. Figure 1 depicts the LR-INS filter results for position, velocity and attitude. All estimates stay well inside their 3σ uncertainty bounds except for some sporadic outliers on the z-axis. Furthermore, all uncertainties are bound due to regular local reference switching. During the 24 h flight the reference frame was switched 6386 times. We use the Normalized Estimation Error Squared (NEES) divided by the number of included states to rate the consistency of the filter. For an ideal filter the weighted NEES should have a mean of 1. We get a NEES mean of 0.83, 0.44 and 0.09 for position, velocity and attitude, respectively. This means that the filter covariance estimates are conservative for all three states.

TABLE I
24 h Simulation parameters

Parameter	Value	Unit
IMU frequency	180	Hz
Acceleration std. dev.	1e-2	m/s^2
Gyroscope std. dev.	1e-3	deg/s
Visual odometry frequency	5	Hz
Visual odometry position std. dev.	1e-3	m
Visual odometry orientation std. dev.	5e-1	deg
PRF frequency	5	Hz
PRF std. dev.	2e-2	m
PRF std. dev.	1e-1	deg

Figure 2 depicts a 2 minute zoom view for unobservable x-position and yaw angle. The estimates for the other unobservable states, which are y-and z-position, show a similar behavior. The moment of PRF augmentation is marked by black circles, the filter switch into the PRF by magenta circles. With the integration of PRF measurements the state uncertainties are stabilized. As expected, they drop as soon as the PRF is used as new filter reference. At the time of switching the state errors do not completely vanish as we switch into the estimate of the new local reference while the ground truth gives the real PRF poses.

B. Relative UAV navigation

We verified the usability of the LR-INS approach in flight experiments using the quadrotor platform depicted in Figure 3. The system is equipped with a sensor unit [8] including a stereo camera system an ADIS16405 IMU, a core2duo computer board for visual odometry calculation, an FPGA board for Semi Global Matching (SGM) stereo image processing of 0.5 MPixel@14.6Hz and an Omap3530 based real-time processor board for sensor data fusion and control.

Disparity images are the basis of our key-frame based visual odometry algorithm. Using hardware acceleration increases the measurement frequency but we still have a latency of about 250 ms for the entire vision pipeline. The measurement delay is implicitly compensated by the hardware triggered pose augmentation of the LR-INS. An analysis of the influence of visual odometry measurement delay and frequency can be found in [9].

The left stereo camera is used for PRF detection. Four PRFs in form of APRIL tags are spread within the experimental area. This area of 4x5x2m (LxWxH) is defined by our motion capture system which tracks the pose of a marker mounted on the quadrotor serving as our ground truth. The relative positions between the PRFs are used as controller waypoints. With the quadrotor reaching the waypoint a switch into the PRF is conducted and the next waypoint is set (at the last waypoint starting at the first again). In Figure 4 we depict the estimation errors compared to ground truth.

Similarly to our simulation, the covariances for position drop at the time we switch into a PRF. As we set the ground truth relative to the measured landmark as well, the measured errors become zero. In contrast, the yaw angle covariance

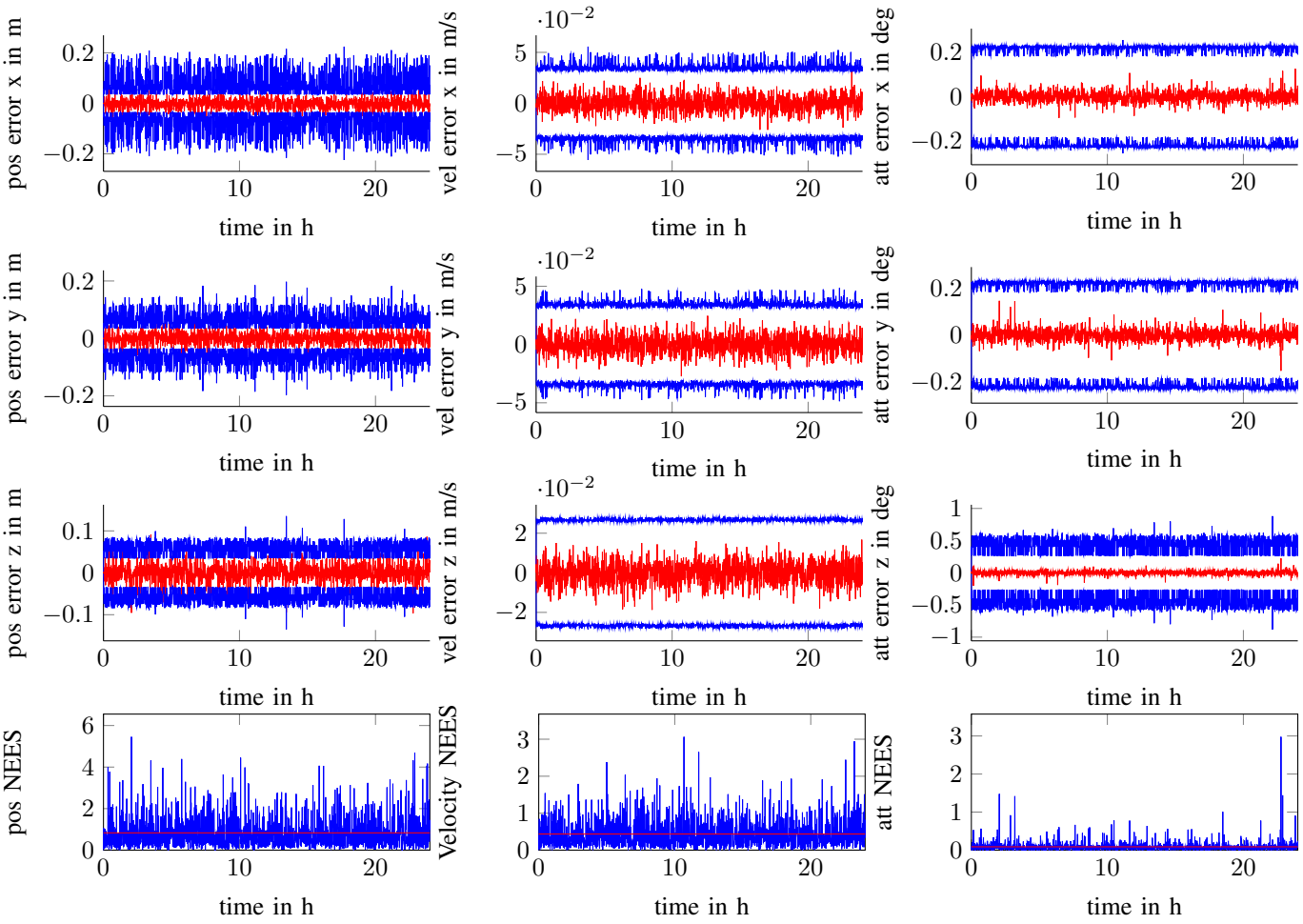

Fig. 1. LR-INS filter results for a 24 h quadrotor flight: estimation errors in red and 3σ uncertainty bounds in blue. The Normalized Estimation Error Squared in blue (NEES) is weighed by the number of included states, its mean is marked by the red line.

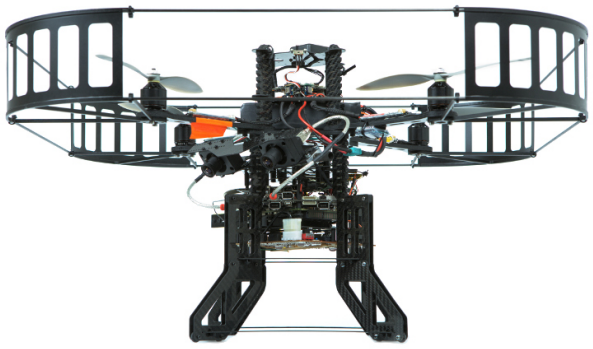

Fig. 3. Quadrotor platform used for LR-INS flight experiments.

becomes higher at switching time. This is caused by the high measurement uncertainty of the PRF compared to the very small yaw drift between filter switches. The increase of yaw covariance induces spikes in the covariances for roll and pitch which are caused by the transformation by the uncertain yaw angle.

Considering error values, the quality of the estimate for position is worse than what we usually achieved using visual odometry only (as for example published in [3]). Furthermore, small jumps in the position error can be observed. These are caused by bad measurements from the APRIL tag detector. As we do not have the real covariance of the PRF measurement we use constant values. Considering the angle errors, there are small oscillations. It can be seen that in these areas the ground truth also sometimes drops out completely. This supports the assumption that the oscillations are coming from bad ground truth measurements.

The NEES shows regular patterns and is considerably higher than in our simulations. We assume that both effects are caused by the static measurement covariances of the APRIL detector which does not reflect the real quality of the measurement. Therefore, we are working on a quality measurement of the detector.

Nevertheless, the actual goal of reference switching, the

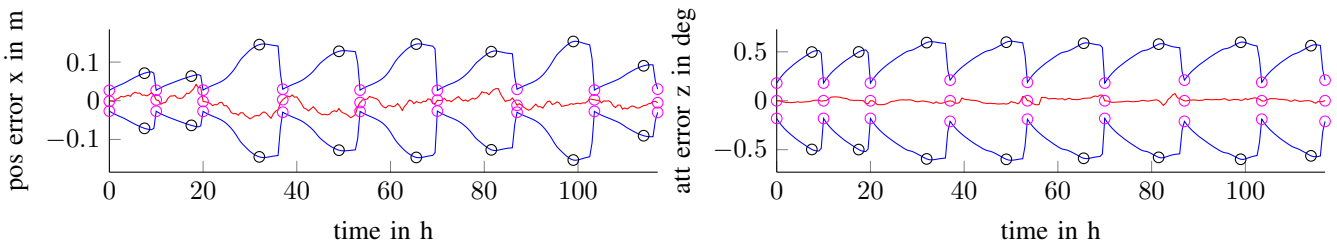

Fig. 2. LR-INS filter results for a 24 h quadrotor flight, zoom-view for position and yaw angle at t=12h. Estimation errors in red and 3σ uncertainty bounds in blue. Black circles indicate augmentation of PRF, magenta circles mark filter switch into PRF.

limitation of covariance increase of unobservable position and yaw can be clearly seen. The quadrotor conducted a flight locally precise to the given references. The whole filter runs with hard-realtime constraints as it is used for control of the highly dynamic quadrotor.

V. Discussion

The introduced LR-filter is a simple method to realize long-term stable filter based state estimation including unobservable system states. By the integration of all needed operations into a general filter prediction step the actual filter implementation method can be freely chosen. State augmentation and reference switching can be easily integrated into square root filters which are numerically superior compared to naive implementations. The computational overhead of square root implementations can be lowered by choosing a clever state ordering. Re-triangularization within the prediction step proceeds from the first row not in triangular form to the matrix triangle base. Therefore, states constant during prediction should be held at the triangle peak whereas changing states should be kept close to the bottom of the triangle.

Filter methods are a good choice in applications with hard-realtime constraints and limited computational resources. Nevertheless, linearization effects degrade the theoretical optimality of the filter. Depending on the non-linearity of the state switching function further linearization errors are induced. Nonlinear, iterative fixed lag smoothers would not suffer from these effects but at the cost of higher computational requirements. Unbounded covariances for unobservable states would not result in numerical problems for smoothers as the anchor state prior can be easily changed.

Our experiments demonstrated that the application of the LR-filter to vision aided inertial navigation results in a long-term stable navigation solution, the LR-INS. In our experiments, we used exteroceptive PRFs. Nevertheless, it is also possible to use directly an augmented robot pose as reference frame for switching. Furthermore, a filter switch could be realized only internally. By re-transforming the estimated system states back to the original global frame, state transformation can be made transparent to external modules as for example a controller while the increase of state covariances is still limited. It has to be considered that the covariances refer to the local reference and not to the global.

The LR-INS can be easily combined with other high level navigation solutions as for example topological navigation or SLAM systems that can also be based on different sensor domains. Human like navigation from one distinct landmark to the next can be easily realized at low computational cost. Considering multi-robot scenarios, navigation relative to a common (changing) reference can be easily realized with a simple navigation filter.

VI. Conclusion and Future Work

We introduced a Local Reference filter approach for stable long-term state estimation including unobservable states. All required operations, which are state augmentation, marginalization and transformation are included into a modified filter prediction step. With this formulation local reference filtering can be integrated into numerically stable square root filters. We applied the LR-filter to vision aided inertial navigation and developed the Local Reference Inertial Navigation System (LR-INS).

We conducted a 24 h simulation of the LR-INS with 6386 reference switches. We showed that the covariances are bound for all states including position and yaw which are unobservable. The evaluation of the Normalized Estimation Error Squared (NEES) showed that the filter is conservative but consistent.

We demonstrated the usability of the LR-INS for control of a highly dynamic, inherently unstable quadrotor with limited on-board processing resources. Switching continuously its reference frame, the quadrotor can navigate robustly using time delayed relative pose updates from an on-board stereo-odometry system. The LR-INS was demonstrated to be suitable for navigation and control of systems with hard-realtime constraints.

As a next step, we will combine our Local Reference Inertial Navigation System with a high-level, scalable, topological mapping approach. In this way long-term locally metric navigation can be realized on resource limited mobile robotic platforms.

Acknowledgments

The authors would like to thank Teodor Tomic (DLR) for the adaptation of his quadrotor controller for the LR-INS. Furthermore, we would like to thank Michael Suppa (DLR) for his great support of our work and the fruitful discussions.

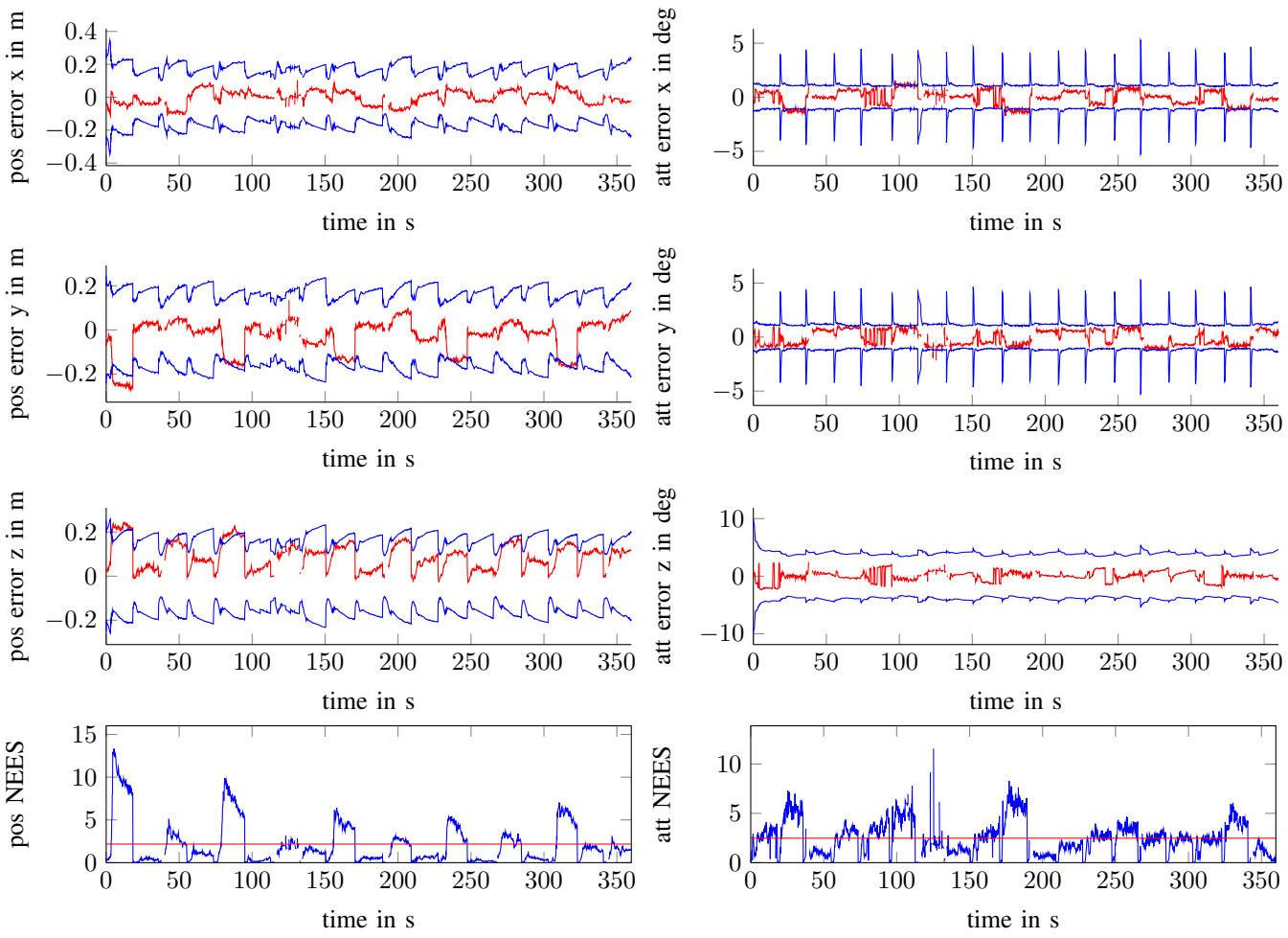

Fig. 4. LR-INS filter results for a real quadrotor flight:

References

[1] GJ Bierman and CL Thornton. Numerical comparison of Kalman filter algorithms: Orbit determination case study. *Automatica*, 13:23–35, 1977.

[2] Michael Kaess and Stephen Williams. Concurrent filtering and smoothing. *. . . (FUSION), 2012 15th . . .*, 2012.

[3] Heiko Hirschmüller Korbinian Schmid, Teodor Tomic, Elmar Mair, Philipp Lutz. Autonomous Vision Based MAV for Indoor and Outdoor Navigation. *Accepted for Journal of Field Robotics, Special Issue on Low-Altitude Flight of UAVs*, 2014.

[4] A Martinelli. Vision and IMU data fusion: Closed-form solutions for attitude, speed, absolute scale, and bias determination. *Robotics, IEEE Transactions on*, 28(1): 44–60, 2012.

[5] PS Maybeck. *Stochastic models, estimation, and control*. 1982. ISBN 0124807011.

[6] A.I. Mourikis and S.I. Roumeliotis. A Multi-State Constraint Kalman Filter for Vision-aided Inertial Navigation. *Proceedings 2007 IEEE International Conference on Robotics and Automation*, 2007. ISSN 1050-4729.

[7] S.I. Roumeliotis and J.W. Burdick. Stochastic cloning: a generalized framework for processing relative state measurements. *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, 2(1):1788–1795.

[8] K. Schmid and H. Hirschmüller. Stereo Vision and IMU based Real-Time Ego-Motion and Depth Image Computation on a Handheld Device. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, 2013.

[9] K. Schmid, F. Ruess, M. Suppa, and D. Burschka. State estimation for highly dynamic flying systems using key frame odometry with varying time delays. *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2997–3004, October 2012.

[10] JD Tardós and J Neira. Robust mapping and localization in indoor environments using sonar data. *. . . International Journal of . . .*, 21(4):311–330, 2002.

[11] Jan Wendel. *Integrierte Navigationssysteme: Sensordatenfusion, GPS und Inertiale Navigation*. Oldenbourg Verlag, 2007.