Smoother based 3-D Attitude Estimation for Mobile Robot Localization

Stergios I. Roumeliotis¹, Gaurav S. Sukhatme² and George A. Bekey²

¹Department of Electrical Engineering
²Department of Computer Science
Institute for Robotics and Intelligent Systems
University of Southern California
Los Angeles, CA 90089-0781

ABSTRACT

The mobile robot localization problem is decomposed and treated as a two-stage iterative process. The attitude is estimated first and is then used for position estimation. The innovation of our method presented in the sequel is the incorporation of a smoother, in the attitude estimation loop that outperforms in estimate accuracy any other Kalman filter based technique. The smoother is ideally suited for this application because it can exploit the special nature of the data fused. These data are either provided by inertial sensors and, in general, appear with high frequency, or result from absolute orientation measuring devices and arrive at lower rates. Two Kalman filters form the smoother. During each time interval one of them propagates the attitude estimate forward in time until it is updated (by an absolute orientation sensor). Then, the second filter propagates the recently renewed estimate backwards this time. Combining the outcome of the two filters, the smoother optimally exploits the limited (only at certain time instances) observability of the system. The backward filter propagation is particularly beneficial for those parts of the trajectory that exhibit lower levels of accuracy as these were determined during the forward filter propagation. This way the estimates across each time interval, are smoothed and thus a more uniform and better quality orientation estimate is achieved. Another prominent feature of this new approach is that the frequency of the absolute orientation sensors can be used as a design parameter. The desired level of accuracy for the attitude estimation and subsequently for the three-dimensional position estimation can be preset. At the system model level, gyro modeling is used which relies on the integration of the kinematic equations to propagate the attitude estimates. Thus, the Indirect (error state) form of the Kalman filter is developed for both the forward and the backward parts of the smoother. The main benefit of this choice is that the tedious and highly sensitive dynamic modeling of the mobile robot and its environment is avoided. The proposed implementation is independent of the structure of the vehicle or the morphology of the ground. It can easily be transferred to another mobile platform provided that it carries an equivalent set of sensors. Finally, instead of Euler angles mostly found in the relevant robotics literature, quaternions are selected for the three-dimensional attitude representation. The quaternion parameterization was chosen mainly for three practical reasons. First, the prediction equations are treated linearly, then the representation is free of singularities and finally the attitude matrix is algebraic in the quaternion components, thus eliminating the need for transcendental functions. The proposed innovative algorithm is tested in simulation and the overall improvement in position estimation is demonstrated.

1. INTRODUCTION

In December 1996 NASA launched the pathfinder mission to Mars. On July 4th 1997, the mission deployed an autonomous micro-rover on the surface of Mars shown in figure 1. The robot’s primary mission was to do spectroscopic analysis on Martian rocks of interest to scientists on Earth. Due to challenges such as round trip communication
delay time the robot was equipped to perform a high degree of autonomous goal seeking and hazard avoidance behavior.

Future missions to Mars will demand longer traverses of similar rovers to sites of scientific interest separated by kilometers of distance (Hayati, Volpe, Backes, Balaram & Welch 1996). In order to autonomously navigate and perform its scientific tasks, the rover needs to know its exact position and orientation; to localize itself.

Localization is a problem concerning every autonomous vehicle. Different techniques have been developed to tackle this problem; these can be sorted into two main categories:

**Relative (local) localization** which consists of evaluating the position and the orientation through integration of information provided by diverse (usually encoders or inertial) sensors. The integration is started from the initial position and orientation and is continuously updated in time.

**Absolute (global) localization** which is the technique that permits the vehicle to determine its position directly in the domain of evolution of the motion. These methods usually rely on navigation beacons, active or passive landmarks, map matching or satellite-based signals like Global Positioning System (GPS).

Local localization is also known as dead-reckoning (DR). In DR we have two basic approaches: odometric systems and inertial navigation systems (INS). In most mobile robots odometry is implemented by means of optical encoders that monitor the wheel revolutions and steering angles of the robot wheels. Using simple geometric equations (kinematic model of the vehicle), the encoder data are then used to compute the momentary position of the vehicle relative to a known starting position. INS are widely used in aviation and lately in outdoor robots. Most of them consist of gyroscopes and accelerometers that provide angular rate and velocity rate information. By integrating this rate information INS systems calculate the position and orientation of the vehicle.

Though its simplicity has made DR a widely used technique, we can not rely on it for long distances. The basic drawbacks associated with it are:

- The kinematic model of the vehicle is never accurate (for example, we do not know with infinite precision the distance between the wheel axes of the vehicle).
- The sensor models also suffer from inaccuracies and can become very complicated (for example use of exponential model for the gyroscope drift).
- The sensor readings are corrupted by noise.
- The motion of the vehicle involves external sources of error that are not observable by the sensors used (for example, slippage in the direction of motion or in the direction perpendicular to motion).

Due to the above reasons, there is error in the calculation of the vehicle’s position and orientation which generally grows unbounded with time. Substantial improvement is provided by applying Kalman filtering techniques. These techniques have been used successfully in position estimation problems such as missile tracking and ship navigation for the last four decades (Grewal & Andrews 1993) and their field of application has been extended to mobile robots. The next step to limit the error growth relies on absolute localization techniques. Involvement of global sensor measurements can drastically increase the accuracy of the estimate and keep the associated uncertainty within certain bounds. The basic intuition is that estimates based on relative localization, degrade rapidly in time and show error growth that is unbounded. The introduction of absolute techniques serves to prevent the degradation in the estimate.

In this report we address the problem of 3-D localization for mobile robots in the absence of absolute positioning information. We concentrate on bounding the attitude uncertainty through periodic use of absolute attitude measurements. As a consequence the position estimate degrades slowly compared to the case when no absolute orientation measurements are available. In our method the attitude estimation relies on the gyros when the vehicle is in motion while a tri-axial accelerometer is used as an absolute orientation measuring device (roll and pitch) in conjunction with a sun sensor (yaw) when the vehicle is at stop (figure (2)). At the end of each interval of motion a smoother is introduced which propagates backwards the new absolute orientation information. The post processing of the previously acquired gyro information allows for uniformly lower uncertainty of the attitude estimate throughout the interval of smoothing; that is when the vehicle was in motion. Smoothing which is being applied to mobile
robot localization for the first time, has been successfully used in the postflight construction of the attitude profile of a spacecraft in order to support the integration of data from a space-born sensor like a camera for example. Before we plunge into the details of the method presented here, it would be useful to examine first some of the research efforts addressing the general problem of robot localization.

2. PREVIOUS WORK

The following review, contains representative examples of different approaches to robot localization.

In order to deal with systematic errors which are particularly notable in indoor applications, J. Borenstein and L. Feng present in (Borenstein & Feng 1995) a calibration technique called the UMBmark test. The dominant systematic error sources are identified as the difference in wheel diameter and the uncertainty about the effective wheel base. By measuring the individual contribution of these vehicle specific, dominant error sources, they counteract their effect in software.

In another paper (Borenstein & Feng 1996) the same authors use measurements from odometric sensors (wheel encoders) and INS (gyroscope) during different time intervals. Their method, gyrodometry, uses odometry data most of the time, while substituting gyro data only during brief instances (e.g. when the vehicle goes over a bump) during which gyro and odometry data differ drastically. This way the system is kept largely free of the drift associated with the gyroscope and compensates for the non-systematic errors introduced by odometry.

In (Barshan & Durrant-Whyte 1995) the authors use a low cost INS system (3 gyroscopes, a triaxial accelerometer) and 2 tilt sensors. Their approach is to incorporate in the system a priori information about the error characteristics of the inertial sensors and to use this directly in an Extended Kalman Filter (EKF) to estimate position before supplementing the INS with absolute sensing mechanisms. The main drawbacks of this work is that the statistical assumptions for the accelerations are not justified anywhere and the orientation information is missing in the calculation of the position.

On the other hand, work related to absolute localization is presented by J. J. Leonard and H. F. Durrant-Whyte. In (Leonard & Durrant-Whyte 1991) they develop a system in which the basic localization algorithm is formalized as a vehicle-tracking problem, employing an EKF to match beacon observations (environment features) to a navigation map to maintain an estimate of the position of the mobile robot.

In (Bonnifait & Garcia 1996) the authors also use an EKF in order to fuse odometry and angular measurements of known landmarks (light sources detected using a CCD camera).

In (Baumgartner & Skaar 1994) E. T. Baumgartner and S. B. Skaar estimate a vehicle's position and orientation based on the observation of visual cues located at discrete positions within structured environment. An EKF is used to combine these visual observations with sensed wheel rotations in order to produce continuously optimal estimates.
Most of the previously mentioned efforts limit themselves to the case of planar motion. In addition, their accuracy depends heavily on the presence of some form of an absolute positioning system (vision or beacon based). In our approach, we consider the case of motion on uneven terrain (3-D localization). We propose an estimation algorithm that is capable of incorporating absolute position measurements, and is also able to provide reliable estimates even in the absence of externally provided positioning information. The focus of our method is on attitude estimation. The advantage of this approach is that by smoothing the gyro data every time an absolute orientation measurement arrives, it is able to reduce the value of the attitude uncertainty throughout the whole time interval that the vehicle was in motion. During this time the vehicle's orientation was not observable. The estimator is developed in the form of an Indirect Kalman filter that operates on the error state. As in most inertial navigation applications in aerospace, gyro modeling is used instead of dynamic modeling and thus complicated modeling of the interaction of the vehicle with the terrain is avoided. In the next section we examine the dependence of the position estimation on the attitude estimation. In section 4 we compare different forms of the Kalman filter to conclude in favor of the Indirect. The planar case is presented in section 5 mainly because it is linear and allows for study of the observability and issues concerning the role of the Kalman filter as a signal processing filter. Section 6 contains a detailed derivation of the error state equations for the 3-D case using unit quaternions. The linear time-variant equations of the system model and the non-linear equations of the observation model are then derived. An Indirect Kalman filter based on these models is finally developed. Section 7 demonstrates the effect of the improved attitude estimation on the position estimate, while section 8 summarizes the conclusions of this work. Section 9 reveals issues of future consideration.

3. LOCALIZATION AND ATTITUDE ESTIMATION

The sensor suite of Rocky 7 includes wheel encoders, gyros, accelerometers and a sun sensor. Since there is no device measuring the absolute position of the rover (GPS signals are not detectable on the surface of Mars), the position can only be estimated through the integration of the accelerometer signal. The quality of this estimate depends heavily on the noise profile of this inertial sensor. Bias and noise in the accelerometer signal can cause serious drift of the position estimate. The task at hand is even more difficult because the propagation of the position relies upon the attitude estimate. Even small errors in the orientation can fast become large errors in position. Formally speaking, the position is not observable and thus the uncertainty of its estimate will grow unbounded. The most promising course of action with the current set of sensors would be to focus on gaining a very precise attitude estimate. As a result the position uncertainty will grow at a slower rate and will allow for increased autonomy. The estimator will sustain accurate position estimates during longer traverses before other localization techniques (such as visual
tracking) are required.

The attitude estimate is used twice during position estimation:

i) The accelerometer measures both the vehicle’s acceleration and the projection of the gravitational acceleration on the accelerometer local frame. The relation between these is described by the following equation:

\[ \text{\( \ddot{p}(t) = \ddot{f}(t)/m = \ddot{a}_{\text{accelerometer}}(t) - R(q(t))\ddot{g} \)} \]

where \( \ddot{p} \) is the vehicle’s (non-gravitational) acceleration, \( \ddot{a}_{\text{accelerometer}} \) is the measurement from the 3-axis accelerometer and \( \ddot{g} \) is the gravitational acceleration. Precise knowledge of the orientation matrix \( R(q) \) is mandatory to extract \( \ddot{p} \) accurately.

ii) The next step requires integration of \( \ddot{p} \) to derive the position. The information of \( \ddot{p} \) is local (i.e. expressed in the local coordinate frame) and in order to calculate the position in global coordinates the attitude information is once again required:

\[ \dot{p}(t) = \int_0^t dt' \int_0^{t'} R(q(t''))\ddot{p}(t'') dt'' \]

3.1. Attitude Measuring Devices

Amongst the sensors onboard the rover, the gyroscopes can be used to calculate the attitude of the vehicle by simply integrating their signal. On the other hand, the sun sensor measures directly the values of the two components of a two-dimensional vector. This vector is the projection of the unit vector towards the sun on the sun sensor plane. Another sensory input of the same nature is required in order to satisfy attitude observability requirements. It is evident from equations (1) and (2) that the accelerometer is mainly used to advance the position estimate. It can also be used in an alternative way. An accelerometer can measure the gravitational acceleration, a three-dimensional vector parallel to the local vertical. This provides another orientation fix independent from the sun and thus makes the vehicle’s attitude observable.

This method fails when the rover is in motion. The gravity vector is then “contaminated” by the non-gravitational acceleration of the vehicle seen in equation (1). The gravity vector could be extracted while the vehicle is moving if an independent measurement of its own acceleration was available. Research efforts have tried to address this problem. In (Vaganov, Aldon & Fournier 1993) the authors attempt to remove the non-gravitational acceleration by approximating it. Odometry information from the encoders is used in this approximation. Each encoder signal has to undergo two differentiations to provide acceleration estimates along the local \( x \) and \( y \) directions. The main claim reported is that in the long term the roll and pitch estimates from the accelerometers have good mean values while in the short term the gyroscopes are mostly responsible for the attitude estimation. It is also clear that instantaneous attitude estimates provided by the accelerometer can be completely wrong. As the authors mention, robot vibrations are wrongly converted into angles and numerical differentiations affect the quality of the result. In their work, a Kalman filter that estimates the drift of the gyroscopes based on the accelerometer information is developed. The algorithm was tested in the case of a mobile robot that moves on flat ground except when one of the wheels climbs over a 2 cm plank using a small inclined plane. After that the robot is moving again on flat ground. We believe that this approach is sufficient for indoor applications and can deal with cases of motion over small objects but it is not accurate enough for outdoor environments mainly because of the limited accuracy of the estimates of the non-gravitational acceleration. This is due to numerical differentiation, limited precision of encoders, longitudinal wheel slippage and lateral skidding during turning. The approach of Fuke and Krotkov (Fuke & Krotkov 1996) is similar. They concentrate on motion in a straight line over inclined surfaces (sandy craters). Their method suffers from the same drawbacks as the previously discussed method.

A more thorough consideration of the problem would require dynamic modeling of the vehicle. An estimator that incorporates a dynamic model of the vehicle could estimate its non-gravitational accelerations. An example of an effort along these lines comes from the Aerobots domain. In (Scheid, Bayard, Balaram & Gennery 1997) an estimator for a telerobotic balloon is described. While the balloon is floating in the air, the “swaying” of its gondola can be modeled as a pendulum. That allows for modeling of its dynamics. The result is an estimator that uses, in effect, the accelerometer as another source of attitude information.
3.2. Dynamic Model Replacement

In our approach we avoid dynamic modeling and we restrict ourselves to use the accelerometer only when the rover is at stop. The reasoning behind this is as follows:

i) The most elementary of all reasons is that every time there is a modification on the rover (i.e. a mass changes, or a part is relocated, or dimensions are altered) the dynamic modeling has to be redone. The produced estimator is tailored for a specific structure. A slightly different vehicle would require a new estimator.

ii) A more practical reason is that dynamic modeling would require a very large number of states (consider that the vehicle has 6 wheels, 4 steering joints, 3 bogey joints). An estimator has to be practical as far as its computational needs are concerned. The size of the estimated state can have large computational demands with very little gain in precision.

iii) Dynamic modeling and the added complexity caused does not always produces the expected results. One example is an attempt by Lefferts and Markley (Lefferts & Markley 1976) to model the attitude dynamics of the NIMBUS-6 spacecraft, which indicated that dynamics modeling with elaborate torque models could still not give acceptable attitude determination accuracy. For this reason most attitude estimation applications in the aerospace domain use gyros in a dynamic model replacement mode (Wertz 1978).

iv) Modeling a mobile robot moving on rough terrain is more complicated than that of a spacecraft essentially because of the interaction between the vehicle and the ground. The external forces on a spacecraft travelling on a ballistic trajectory in space, are precisely described by the law of gravitational forces. The interaction of a rover with the ground depends on many parameters and requires assumptions to be made (the point contact assumption is frequently used). The modeling of the vehicle-terrain dynamic effects as the wheel impact, wheel slippage, wheel sinkage, requires prior knowledge of the ground parameters (i.e. friction coefficients as a function of soil shear strength, elastic modulus etc). In addition, the lateral slippage is not observable and can not be accounted for in the dynamic model. Precise modeling of the motors is also required to obtain the real values of the torques acting on each of the wheels. At this point we also have to mention that the applicability of Kalman filtering techniques rests on the availability of an accurate dynamic model. An inaccurate model can cause the estimate to drift away from the real value.

v) Looking at the problem from another perspective, the vehicle has a suspension system (bogey joints) that allows it to climb over objects of height larger than its wheel’s diameter. One of the purposes of every suspension system is to decouple the vehicle’s motion from the terrain morphology. The “ideal” suspension system would support a motion of the vehicle that would iniminate to some extent, the motion of a hovercraft. The rover then could move in a very smooth fashion. A motion like this is prone to be effectively estimated following methods that rely on the use of inertial navigation systems as a dynamic model replacement, and are independent of the particular terrain.

As previously mentioned, we use the accelerometer as an attitude sensor, only when the vehicle is at stop. The stopping is justifiable for a variety of reasons. To begin with, the rover is expected to stop to deploy instruments against rocks, as a spectrometer or a drill. In other instances it is expected to stop to save energy, or to just wait for commands from the operator on earth. In every case speed is not the main requirement and thus the slow motion can include some small intervals of stoppage. The gain in accuracy and simplicity are two more reasons to exploit the stop to obtain a better attitude estimate.

When the vehicle is at stop the accelerometer measures only the gravitational acceleration:

\[ \ddot{a}_{\text{accelerometer}} = R(q)\ddot{g} \]  \hspace{1cm} (3)

from this equation the roll and pitch of the vehicle can be precisely calculated. The sun sensor then provides for the yaw measurement and thus the matrix \( R(q) \) is known precisely when at stop.

4. FORMS OF THE KALMAN FILTER

As mentioned before, Kalman filtering has been widely used for localization purposes. The forms that usually appear in mobile robot applications are the (basic) linear Kalman filter and the Extended Kalman filter (EKF). In this work we choose to use the Indirect-feedback form of the Kalman filter. In the following sections we will derive the equations needed for such formulation. Before that we should examine the reasons to select this form over others commonly found in the robotics literature.
4.1. Indirect versus Direct Kalman filter

A very important aspect of the implementation of a Kalman filter in conjunction with inertial navigation systems (INS) is the use of the indirect instead of the direct form, also referred to as the error state and the total state formulation respectively (Maybeck 1979). As the name indicates, in the total state (direct) formulation, total states such as orientation are among the variables in the filter, and the measurements are INS outputs, such as from a gyro, and external source signals. In the error state (indirect) formulation, the errors in orientation are among the estimated variables, and each measurement presented to the filter is the difference between the INS and the external source data.

There are some serious drawbacks in the direct filter implementation. Being in the INS loop and using the total state representation, the filter would have to maintain explicit, accurate awareness of the vehicle's angular motion as well as attempt to suppress noisy and erroneous data. Sampled data require a sampling rate of at least twice the frequency of the highest frequency signal of interest (in practice a factor of 5-10 is used) for adequate reconstruction of the continuous time system behavior. The filter would have to perform all the required computations within a short sampling period. Moreover, in most cases, the estimation algorithm and thus the Kalman filter is allocated only a small portion of the capabilities of a central processor, and it runs in the background at a lower priority than more critical algorithms, such as real-time vision, obstacle avoidance, fault detection etc.

In addition, the dynamics involved in the total state description of the filter include a high frequency component and are well described only by a non-linear model. The development of a Kalman filter is predicated upon an adequate linear system model, and such a total state model does not exist.

Another disadvantage of the direct filter design is that if the filter fails (as by a temporary computer failure) the entire navigation algorithm will fail. The INS is useless without the filter. From the reliability point of view it would be desirable to provide an emergency degraded performance mode in such a case of failure*. A direct filter would be ideal for fault detection and identification purposes (Roumeliotis, Sukhatme & Bekey 1998a), (Roumeliotis, Sukhatme & Bekey 1998b).

The Indirect (error state) Kalman filter estimates the errors in the navigation and attitude information using the difference between the INS and external sources of data. The INS itself is able to follow the high frequency motions of the vehicle very accurately, and there is no need to model these dynamics explicitly in the filter. Instead, the dynamics upon which the Indirect filter is based is a set of inertial system error propagation equations which are low frequency and very adequately represented as linear. Because the filter is out of the INS loop and it is based on low frequency dynamics, its sample rate can be much lower than that of the Direct filter. In fact, an effective Indirect filter can be developed with a sample period (of the external source) of the order of minutes (Maybeck 1979). This is very practical with respect to the amount of computer time required. For these reasons, the error state formulation is used in essentially all terrestrial aided inertial navigation systems (Siouris 1993).

4.2. Feedforward versus feedback Indirect Kalman filter

The basic difference between the feedforward and feedback Indirect Kalman filter is mainly in the way they handle the updated error estimate. In the first case the updated error estimate is being fed forward to correct the current orientation estimate without updating the INS. In the feedback formulation the correction is actually fed back to the INS to correct its “new” starting point, i.e. the state that the integration for the new time step will start from. In a sense the difference between the feedforward and feedback forms is equivalent to the difference between the Linearized Kalman filter and the Extended Kalman filter. In the second one the state propagation starts from the corrected (updated) state right after a measurement while in the Linearized filter the propagation continues at the state that the propagation has reached when the measurement appeared, thus ignoring the correction just computed. The Linearized Kalman filter and the feedforward Indirect Kalman filter are free to drift with unbounded errors. The basic assumption of Kalman filtering that the estimation errors stay small becomes invalid and thus the performance of the estimator will degrade.

5. PLANAR MOTION. ONE GYRO CASE - BIAS COMPENSATION

The greatest difficulty in all attitude estimation approaches that use gyro, is the low frequency noise component, also referred to as bias or drift that violates the white noise assumption required for standard Kalman filtering. This

*As we will see in the sequel the Indirect filter in case of a failure can continue to provide estimates by acting as an integrator on the INS data.
problem has attracted the interest of many researchers since the early days of the space program. The first papers describing statistical models of gyro drifts were by Newton (Newton 1960) and Hammon (Hammon 1962). Newton considered additive white noise in the gyro drift, while Hammon assumed that the gyro drift rates are exponentially correlated random processes. Dushman (Dushman 1962) considered a drift-rate model obtained by adding a random walk component to Hammon’s autocorrelation function. Inclusion of the gyro noise model in a Kalman filter by suitably augmenting the state vector has the potential to provide estimates of the sensor bias when the observability requirement is satisfied. Early implementations of gyro noise models in Kalman filters can be found in: (Potter & Velde 1968), (Pauling, Jackson & Brown 1969), (Toda, Heiss & Schlee 1969).

5.1. Systron Donner Quartz Gyro

In the case of Rocky 7, the gyro device that has been used is the Systron Donner Quartz Gyro (QRS11-100-420). In the information provided in (Allen & Chang 1993), it is obvious that this gyro does not have a stable bias. From page 1-4: "Low Rate Application - These gyro showed reasonable performance for rate scale factor stability but would not be useful for applications where bias stability was of high importance to meet mission requirements. The bias changed significantly as the input rate was changing making predictable bias compensation very difficult”.

Long term bias stability data were gathered in a sequel to the initial testing experiments with the intent to create a stochastic model useful for attitude estimator performance prediction. The noise model used is the one in (King 1984) and (Kochakian 1980). This model does not capture the fact that in reality the bias uncertainty will not grow to infinity after infinite time. It is a useful and simple construct for attitude estimator design and simulation, under the assumption that the bias will be calibrated at intervals which are short compared to when the noise model predicts grossly large bias variance. The model assumes that the gyro noise is composed of 3 elements, namely: rate noise \( n_r(t) \) (additive white noise), rate flicker noise \( n_f(t) \) (generated when white noise passes through a filter with transfer function \( 1/\sqrt{s} \)) and a rate random walk \( n_w(t) \) (generated when white noise passes through a filter with transfer function \( 1/s \)). The Power Spectral Density (PSD) of the noise is measured experimentally and the logarithmic plots of the PSD with respect to frequency are used to fit the described model. The intensities calculated from that (ignoring the flicker noise) were: \( \sigma_r = \sqrt{N_r} = 0.009 \, (deg/sec)/\sqrt{Hz} \) and \( \sigma_w = \sqrt{N_w} = 0.0005012 \, (deg/sec)/\sqrt{Hz} \).

5.2. Gyro Noise Model

In our approach we use the simple and realistic model (Farrenkopf 1978). In this model the angular velocity \( \dot{\theta} = \omega \) is related to the gyro output \( \omega_m \) according to the equation:

\[
\dot{\theta} = \omega_m + b + n_r
\]  

where \( b \) is the drift-rate bias and \( n_r \) is the drift-rate noise. \( n_r \) is assumed to be a Gaussian white-noise process:

\[
E[n_r(t)] = 0
\]  

\[
E[n_r(t) n_r(t')] = N_r \delta(t - t')
\]  

The drift-rate bias is not a static quantity but it is driven by a second Gaussian white-noise process, the gyro drift-rate ramp noise:

\[
\dot{b} = n_w
\]  

with

\[
E[n_w(t)] = 0
\]
\[
E[n_w(t)n'_w(t')] = N_w \delta(t - t')
\]  
(9)

The two noise processes are assumed to be uncorrelated

\[
E[n_w(t)n'_w(t')] = 0
\]  
(10)

We study the simple case of attitude estimation when the vehicle moves in a plane. It is assumed that only one gyro is used and that attitude information (the yaw) is provided directly by another sensor\(^1\) at frequent intervals. The purpose of this section is to reveal how the gyro bias can be estimated using an absolute measurement of the orientation. The two-dimensional case has the advantage that the system equations are linear; this allows us to study observability analytically. In the sequel an Indirect Kalman filter to estimate the orientation is developed.

### 5.3. Equations of the feedback Indirect Kalman filter

As previously mentioned in equation (4) the real angular velocity can be expressed as:

\[
\dot{\theta}_{\text{true}} = \omega_{\text{measured}} + b_{\text{true}} + n_r
\]  
(11)

where

\[
\dot{b}_{\text{true}} = n_w
\]  
(12)

These equation can be rearranged in a matrix form as:

\[
\frac{d}{dt} \begin{bmatrix} \theta_{\text{true}} \\ b_{\text{true}} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \theta_{\text{true}} \\ b_{\text{true}} \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \omega_{\text{measured}} + \begin{bmatrix} n_r \\ n_w \end{bmatrix}
\]  
(13)

The absolute orientation measurement is:

\[
z = \theta_{\text{measured}} = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \theta_{\text{true}} \\ b_{\text{true}} \end{bmatrix} + n_\theta
\]  
(14)

where \(n_\theta\) is the noise for the sensor measuring the absolute orientation. The assumption is that \(n_\theta\) is a Gaussian white-noise process with:

\[
E[n_\theta(t)] = 0
\]  
(15)

\[
E[n_\theta(t)n'_\theta(t')] = N_\theta \delta(t - t')
\]  
(16)

The orientation estimate obtained by integrating the gyro signal is given by:

\[
\dot{\theta}_i = \omega_{\text{measured}} + b_i
\]  
(17)

while the rate of the estimated bias is given by:

\[
\dot{b}_i = 0
\]  
(18)

The state equations for the integrator can be written in a matrix form as:

---

\(^1\)This external sensor can be a magnetic compass or a sun sensor in this two dimensional example. From now on we will assume that this sensor measures the absolute orientation of the vehicle.
\[ \frac{d}{dt} \begin{bmatrix} \theta_i \\ b_i \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \theta_i \\ b_i \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \omega_{measured} \]  

(19)

Subtracting equations (11) and (17) the error in orientation can be written as:

\[ \Delta \dot{\theta} = \Delta b + n_r \]  

(20)

where \( \Delta \theta \) is the error in orientation and \( \Delta b \) is the bias error. Subtracting equations (12) and (18) the bias error can be written as:

\[ \Delta \dot{b} = n_w \]  

(21)

The error propagation equations for the Indirect (error state) Kalman filter can be rearranged in a matrix form as:

\[ \frac{d}{dt} \begin{bmatrix} \Delta \theta \\ \Delta b \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta \theta \\ \Delta b \end{bmatrix} + \begin{bmatrix} n_r \\ n_w \end{bmatrix} \]  

(22)

or in a more compact form as:

\[ \frac{d}{dt} \Delta x = F \Delta x + n \]  

(23)

We will assume that the measurement provided to the Indirect Kalman filter is:

\[ \Delta z = \theta_{\text{measured}} - \theta_i = \theta_{\text{true}} + n_\theta - \theta_i = \Delta \theta + n_\theta \]  

(24)

where \( \theta_i \) is available through the gyro signal integration and \( \theta_{\text{measured}} \) is the absolute orientation measurement. This equation in matrix form becomes:

\[ \Delta z = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \Delta \theta \\ \Delta b \end{bmatrix} + n_\theta \]  

(25)

or in a more compact form:

\[ \Delta z = H \Delta x + n_\theta \]  

(26)

The continuous Kalman filter equation for the covariance is:

\[ \dot{P} = FP + PF^T + Q - PH^T R^{-1} H P \]  

(27)

where \( P \) is the covariance matrix, \( F \) is the system matrix, \( H \) is the measurement matrix, \( Q \) is the system noise covariance matrix and \( R \) is the measurement noise covariance matrix. Consider this equation in the steady state case where \( \lim_{t \to \infty} (\dot{P}) = 0 \) and we have

\[
0 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix} + \begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} + \begin{bmatrix} N_r & 0 \\ 0 & N_w \end{bmatrix} - \\
\begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \left[ N_\theta \right]^{-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix} 
\]  

(28)
Solving this equation for the elements of matrix $P$ we find:

$$p_{11} = \sqrt{N_\theta \sqrt{N_r + 2\sqrt{N_w N_\theta}}}$$  \hspace{1cm} (29)

$$p_{12} = \sqrt{N_w N_\theta}$$  \hspace{1cm} (30)

$$p_{22} = \sqrt{N_w \sqrt{N_r + 2\sqrt{N_w N_\theta}}}$$  \hspace{1cm} (31)

Now the steady state Kalman gain can be calculated from:

$$K = PH^T R^{-1} = \begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} N_\theta \\ \sqrt{N_w N_\theta} \end{bmatrix}^{-1} = \begin{bmatrix} k_1 \\ k_2 \end{bmatrix}$$  \hspace{1cm} (32)

The Indirect Kalman filter estimates the error states. The estimate propagation equation with the added correction is:

$$\frac{d}{dt} \begin{bmatrix} \Delta \theta \\ \Delta b \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta \theta \\ \Delta b \end{bmatrix} + \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} \left( \Delta z - \Delta \theta \right)$$  \hspace{1cm} (33)

Substituting the error state estimates according to

$$\Delta \theta = \hat{\theta} - \theta_i$$  \hspace{1cm} (34)

$$\Delta b = \hat{b} - b_i$$  \hspace{1cm} (35)

in (33) we have

$$\frac{d}{dt} \begin{bmatrix} \hat{\theta} - \theta_i \\ \hat{b} - b_i \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{\theta} - \theta_i \\ \hat{b} - b_i \end{bmatrix} + \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} \left( \Delta z - \Delta \theta \right)$$  \hspace{1cm} (36)

separating the estimated from the integrated quantities we have:

$$\frac{d}{dt} \begin{bmatrix} \hat{\theta} \\ \hat{b} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{\theta} \\ \hat{b} \end{bmatrix} + \left( \frac{d}{dt} \begin{bmatrix} \theta_i \\ b_i \end{bmatrix} - \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \theta_i \\ b_i \end{bmatrix} \right) + \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} \left( \Delta z - \Delta \theta \right)$$  \hspace{1cm} (37)

Notice that from equations (24) and (35):

$$\Delta z - \Delta \theta = (\theta_{measured} - \theta_i) - (\hat{\theta} - \theta_i) = \theta_{measured} - \hat{\theta}$$  \hspace{1cm} (38)

Now substituting in (37) from (19) and (38) we have

$$\frac{d}{dt} \begin{bmatrix} \hat{\theta} \\ \hat{b} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{\theta} \\ \hat{b} \end{bmatrix} + \begin{bmatrix} 0 & 1 \end{bmatrix} \omega_{measured} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} \left( \theta_{measured} - \hat{\theta} \right)$$  \hspace{1cm} (39)

Deriving the Laplace transform of this equation we have

$$s \begin{bmatrix} \Theta(s) \\ B(s) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Theta(s) \\ B(s) \end{bmatrix} + \begin{bmatrix} 1 & 0 \end{bmatrix} \Omega_{measured}(s) + \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} \left( \Theta_{measured}(s) - \hat{\Theta}(s) \right)$$  \hspace{1cm} (40)
which becomes

\[
\begin{bmatrix}
  s & 0 \\
 0 & s
\end{bmatrix} - \begin{bmatrix}
  0 & 1 \\
 0 & 0
\end{bmatrix} \begin{bmatrix}
  k_1 & 0 \\
 k_2 & 0
\end{bmatrix} \begin{bmatrix}
  \Theta(s) \\
 B(s)
\end{bmatrix} = \begin{bmatrix}
  1 & 0 \\
 k_1 & k_2
\end{bmatrix} \Omega_{\text{measured}}(s) + \begin{bmatrix}
  k_1 \\
 k_2
\end{bmatrix} \Theta_{\text{measured}}(s)
\]

(41)

solving for \( \Theta(s) \) and \( B(s) \)

\[
\begin{bmatrix}
  \Theta(s) \\
 B(s)
\end{bmatrix} = \begin{bmatrix}
 s + k_1 & -1 \\
 k_2 & s
\end{bmatrix}^{-1} \left( \begin{bmatrix}
  1 \\
 0
\end{bmatrix} \Omega_{\text{measured}}(s) + \begin{bmatrix}
  k_1 \\
 k_2
\end{bmatrix} \Theta_{\text{measured}}(s) \right)
\]

(42)

Finally the quantity of interest in the frequency domain is expressed as:

\[
\tilde{\Theta}(s) = \frac{s}{s^2 + k_1 s + k_2} \Omega_{\text{measured}}(s) + \frac{k_1 s + k_2}{s^2 + k_1 s + k_2} \Theta_{\text{measured}}(s)
\]

(43)

or

\[
\tilde{\Theta}(s) = \frac{s^2}{s^2 + k_1 s + k_2} \Omega_{\text{measured}}(s) + \frac{k_1 s + k_2}{s^2 + k_1 s + k_2} \Theta_{\text{measured}}(s)
\]

(44)

or

\[
\Theta(s) = F(s) \frac{\Omega_{\text{measured}}(s)}{s} + (1 - F(s)) \Theta_{\text{measured}}(s)
\]

(45)

which reveals that the Indirect Kalman filter weighs the two different sources of information (i.e. the integrated gyro signal and the absolute orientation measurement) in a complimentary fashion according to their noise characteristics. To acquire more intuition on how the Indirect Kalman filter deals with the noise, we should examine the case where the sensor inputs are only noise, (i.e. the sensor signals do not contain any useful information at any frequency).

In figure 3, on the first plot we can see the effect of the integration process on the gyro noise. The already strong noise components in the lower frequencies are amplified because of the integration process and thus any useful low frequency information would have been contaminated by this noise. The obvious conclusion is that we can not rely on the gyroscope integration to estimate low frequency motion. On the other hand in higher frequencies the effect of the integrator is to suppress the gyro noise and therefore the gyro becomes reliable for high frequency motion. The ideal situation would be to fuse the integrated gyro information with a sensor (an absolute orientation sensor as a sun sensor for example) that has a complementary noise profile (i.e. the noise is small at lower frequencies and becomes larger at higher frequencies) or at least has a constant noise profile, the same for all frequencies. The second case has been assumed and the noise profiles of both the integrated gyro and the absolute orientation sensor are depicted in the second plot of figure 3.

At this point we should clarify how the Indirect Kalman filter optimally combines the information from the two different sources by examining it as a signal processing entity. The explanation comes from the third plot of figure 3. Function \( F(s) \) which filters the integrated gyro signal works as a high pass filter. It supresses the low frequency noise content and allows the higher frequencies to pass undistorted. To the contrary, function \( 1 - F(s) \) that filters the absolute orientation sensor acts like a low pass filter. It allows the critical low frequency information from the absolute orientation sensor to pass undistorted while reduces its strength in the higher frequencies. This is desired since the estimator performs better if it relies on the gyro in that range.
Figure 3. In the first plot the power spectral density of the gyro noise is displayed before \((\Omega(s))\) and after \((\Omega(s)/s)\) integration \((1/s)\). In the second plot the power spectral densities of the integrated gyro noise and of the absolute orientation sensor noise are presented. The third plot displays the power spectral densities of function \(F(s)\) that filters the integrated gyro noise and of function \(1 - F(s)\) that filters the absolute orientation sensor noise.

5.4. Observability

5.4.1. Measuring Orientation Continuously

The observability matrix of the system is:

\[
\begin{bmatrix}
H^T & F^T \\
F^T & H^T \\
\end{bmatrix} = \begin{bmatrix}
1 & 0 \\
0 & 1 \\
\end{bmatrix}
\]

(46)

Clearly it has rank 2. Thus the system is observable and the uncertainty for both states (orientation and bias) is bounded. This can be seen in figure 4. Both covariances reach a steady state. It is worth noticing that though the bias is not directly measured it can be estimated. This is shown in figure 5.

5.4.2. Measuring Orientation at certain instances

In this case the observability matrix

\[
\begin{bmatrix}
H^T & F^T \\
F^T & H^T \\
\end{bmatrix} = \begin{bmatrix}
0 & 0 \\
0 & 0 \\
\end{bmatrix}
\]

(47)

looses its rank for those time intervals for which we do not have an absolute measurement of the orientation. This can be seen graphically in figure 6. The covariance (which captures the uncertainty) of both the orientation and the bias grow unbounded until a new measurement from the absolute orientation sensor is received. This causes a decrease in the uncertainty which then begins to grow as before. Following this strategy (i.e. taking an absolute orientation measurement only at certain instances) produces a “saw tooth” pattern in the covariance plot.
The formal explanation is provided by the discrete Kalman filter formulation for the covariance:

\[ P_{k,k-1} = \Phi(k, k-1) P_{k-1,k-1} \Phi(k, k-1)^T + Q_{k-1} \]  

(48)

\[ P_{k,k} = P_{k,k-1} - P_{k,k-1} H_k^T [H_k P_{k,k-1} H_k^T + R_k]^{-1} H_k P_{k,k-1} \]  

(49)

where \( P_{k-1,k-1} \) is the covariance matrix from the previous step (time \( t_{k-1} \), measurement k-1), \( P_{k,k-1} \) is the propagated covariance (time \( t_k \), measurement k-1), \( P_{k,k} \) is the updated covariance after the new absolute orientation measurement (time \( t_k \), measurement k). \( \Phi(k, k-1) \) is the state transition matrix from time step \( t_{k-1} \) to time step \( t_k \). \( H_k \) is the measurement matrix at time \( t_k \), \( Q_k \) is the system noise covariance matrix at time \( t_k \) and \( R_k \) is the measurement noise covariance matrix at time \( t_k \).

Equation (48) represents the increase in the system uncertainty at every time step due to the propagated previous system uncertainty and the newly added system noise. This equation is valid at every time step and causes the covariance to grow unbounded. Equation (49) is applied only when a new measurement arrives. It reduces the value of the propagated covariance matrix \( P_{k,k-1} \) since it subtracts from it a positive definite matrix (the \( P_{k,k-1} H_k^T [H_k P_{k,k-1} H_k^T + R_k]^{-1} H_k P_{k,k-1} \)). The smaller the matrix \( R_k \) (i.e. the better the quality of the absolute orientation measurement), the larger the decrease.

At this point we focus on how the Indirect Kalman filter tracks the bias in the case that the system is observable only at certain time instances. In figure (7) we see that in the absence of any other information the filter assumes that the bias is constant. When a new absolute orientation measurement is introduced, the filter updates the bias estimate according to the cumulative effect of the bias on the error in orientation during the preceding time interval. After that the bias is assumed again to be constant, until the next measurement.
Figure 5. The solid line represents the true value of the gyro bias. The dotted line is the estimate of the bias from the Indirect Kalman filter. Though the estimate follows the true value it is obvious that there is a lag. This is because the absolute orientation sensor does not measure the bias directly. It only measures its effect on the orientation error.

It is obvious that in the long run, uncertainty in orientation is bounded. It will not exceed a certain limit depending on the frequency of the absolute orientation updates and on the accuracy of this sensor along with the noisy profile of the gyro and its bias. It seems that the frequency of the absolute measurements is a design parameter that the system engineer can define based on mission specific uncertainty bounds.

5.5. Backward filter
At this point we should revisit the initial problem at hand. We built an Indirect Kalman filter that estimates the vehicle orientation mainly because we need the attitude information to calculate the position update of the rover. The orientation information is required at every time step (equations (1) and (2) ) and its accuracy is crucial for precise position estimation. As we can see from the orientation covariance plot in figure (6) the covariance of the estimate reaches its maximum right before a new absolute orientation measurement. At the time instances before that, the orientation information becomes untrustworthy and the validity of the position updates that depend on it, decays.

In the proposed novel approach we take advantage of the fact that the uncertainty decreases considerably right after the absolute orientation measurement at some time $t_k$. We involve a backward propagation of the Indirect Kalman filter every time we establish a good estimate of the orientation. This is right after each time a new absolute orientation measurement is introduced. In the backward propagation the initial condition for the state is the updated (corrected) state estimate at time $t_k$. Then a backward integration using the same previously acquired (stored) gyro values is performed to calculate the orientation at each previous time instant up to the last absolute orientation measurement at time $t_{k-M}$. The effect of it is to obtain very accurate estimates for the interval right before and close to time $t_k$. These estimates will have the same behavior as they move away (in the backward time direction)
Figure 6. In the first plot the orientation covariance is presented. The sudden decreases take place at the instances when an absolute orientation measurement is provided. The second plot portrays the bias covariance for the same case.

from $t_k$ as the forward filter estimates have when they move away from time $t_{k-M}$ (previous absolute orientation measurement). As the backward filter estimates get closer to $t_{k-M}$ their covariance increases and the quality of the estimate decreases. This is clearly shown in figure (8). To better understand the functionality of the backward filter we examine the equations that govern it as presented in (Gelb 1994).

If the system equation is

$$\dot{x} = Fx + n$$

(50)

then by substituting

$$\tau = T - t$$

(51)

where $\tau$ is the backward time variable and

$$T = t_k - t_{k-M}$$

(52)

is the time interval of smoothing, the backward system equation can be derived from:

$$\frac{dx}{d\tau} = \frac{dx}{dt} \frac{dt}{d\tau} = -\dot{x}$$

(53)

and it is:

$$\frac{dx_b}{d\tau} = -Fx_b - n$$

(54)

The continuous time covariance propagation equation for the forward filter is:
Figure 7. The flat parts depict the constant bias assumption in the integrator, while at the sharp steps changes take place when absolute orientation measurements are provided.

\[
\dot{P} = F P + P F^T + Q
\]

and correspondingly the covariance backwards propagation equation would be:

\[
\dot{P}_b = -F P_b - P_b F^T + Q
\]

In the discrete time domain the previous equation becomes:

\[
P_{b(k-1,k)} = \Phi(k-1,k) P_{b(k,k)} \Phi(k-1,k)^T + Q_k
\]

which is of the same form as (48) and thus the covariance of the backward filter also increases during propagation in the same manner as the forward filter’s covariance does. The main difference is that \(P_b\) starts from a small value at time \(t_k\) and it grows as it moves towards \(t_{k-M}\). These results are depicted in figure (8) for the orientation covariance. In figure (9) we can see the performance of the backward filter for the variable of most interest, the orientation. The orientation estimate for the backward filter is close to its true value for \(t = 400\,sec\) and deviates from it as it reaches close to \(t = 200\,sec\). The next step is to optimally combine these two estimates in the smoother.

5.6. Smoother

The smoother combines the estimates of the forward and the backward Kalman filter to provide estimates of the state at every time as if all the measurements during the entire smoothing interval were available. The total (smoothed) result outperforms both filters. This is evident in figure (8) where the total covariance (or smoothed covariance) is at all times smaller than the smallest of the two filters. This is explained if we look at the equation for the smoothed
Figure 8. The solid line represents the orientation covariance of the forward filter, the dashed the orientation covariance of the backward filter as they propagate during the intervals between three absolute orientation measurements at 0, 100 and 400 secs respectively. The dotted line shows the orientation covariance resulting from the combination of the forward and the backward filter.

covariance:

\[ P_{total}^{-1} = P_f^{-1} + P_b^{-1} \]  

(58)

The equation for the total (smoothed) state is\(^1\)

\[ P_{total}^{-1} \hat{x}_{total} = P_f^{-1} \hat{x}_f + P_b^{-1} \hat{x}_b \]  

(59)

Equation (59) demonstrates how the states are optimally combined to produce the smoothed (total) estimate. Each of the covariance matrices represents the uncertainty of the estimate. The inverse of it represents the certainty or the information contained in this estimate. The higher the uncertainty, the larger the covariance matrix, or the smaller the information matrix, or the smaller the contribution of this filter to the total estimate. In other words the previous equation weighs each of the available estimates (from the forward and the backward filter) according to their certainty. The result is the optimal estimate we can have if all the measurements of the time interval of smoothing were available at once. The achieved improvement is obvious in figures (8), (9), and (10).

At this point we should make clear that the smoothed estimate is not available in real-time. It requires the forward Kalman filter to propagate the estimate from \( t_{k-M} \) to \( t_k \) before the backward filter starts. Thus the smoothed estimation is performed off-line and so is the position estimation using the smoothed orientation estimates.

\(^1\)It should be clarified that the quantities with subscript \( f \) forward are the same as those previously used in the Indirect Kalman filter without any subscript, i.e. \( P_f = P \) and \( \hat{x}_f = \hat{x} \). The new notation is used to distinguish these quantities from the corresponding ones of the backward filter, where the subscript \( b \) is used to denote backward, and the ones of the smoother where the subscript total is used for the smoothed (total) estimate.
Figure 9. The solid line shows the true orientation of the vehicle, the dashed the forward filter’s estimate and the dashed-dotted the backward filter’s estimate. The dotted line is the resulting orientation estimate from the smoother. The smoothed (total) estimate stays close to the backward filter estimate for the second half of the smoothing while for the first part it relies on both filters and slowly abandoms the backward filter estimates which drift away from the true values.

6. 3-D ATTITUDE ESTIMATION

In the previous section we demonstrated the proposed method for the case of planar motion. The linear form of the system and measurement equations allowed us to study observability analytically and examine the role of the Kalman filter as a signal processing unit weighting the sensory information according to their noise power spectral density profile. We also demonstrated the improvement in the overall quality of the attitude estimate when the smoother was involved. The lack of observability for the largest part of the trajectory forces the covariance of the estimate, as calculated in the Kalman filter, to vary between small values right after each (absolute) measurement to large ones right before the next one. The increase of the uncertainty is due to the gyro noise and the bias-drift associated with this sensor. During the instances of observability, the accumulated effect of the gyro noise is cancelled and precise attitude information is provided to the filter. The estimate is trustworthy for limited time though. After the vehicle starts to move again the quality of it decreases. Any algorithm that computes the position of the robot using these attitude estimates will drift especially a little before the next absolute orientation measurement is available. To circumvent this anomaly that appears in situations where the system is observable only during certain time instances, and that manifests itself as a saw-tooth pattern in the time profile of the covariance of the estimate, a smoother has been used. The smoother computes the best estimate of the attitude using all the measurements during a particular time interval. That includes the first measurement which is an absolute orientation one, all the intermediate (inertial) ones coming from the gyro and finally the last one that is an absolute orientation and marks the end of this time interval and the begining of the next one. The improvement is larger for these parts of the trajectory that the estimate is of the lowest quality, i.e. for the time instances right before an absolute measurement. Thus, we are able to sustain uniformly lower levels of uncertainty throughout the duration of the motion.

In this section we apply these ideas in the case of a rover on the surface of Mars. The sensors onboard are three
Figure 10. The solid line shows the true gyro bias, the dashed the forward filter’s estimate and the dashed-dotted the backward filter’s estimate. The dotted line is the resulting bias estimate from the smoother. The smoothed (total) estimate stays close to the backward filter estimate for the second half of the smoothing interval while for the first part it depends on both the forward filter’s estimate and the backward filter’s estimate. This is due to the fact that the initial bias value of the backward filter is more trustworthy for this time interval than the initial value for the forward filter. The explanation is that the initial bias value for the backward filter is the one obtained at the end of the smoothing time interval; it is provided from the forward filter and it is based on the bias values during this particular interval. The forward filter’s bias initial value is based on the bias values during the previous time interval and thus it is definitely less accurate.
Figure 11. Algorithm Flow Chart: The robot is idle until it is commanded to move (by a task planner or externally). While it is in motion the forward Kalman filter processes the information provided by the gyros and produces (real-time) a first approximation of the attitude estimate. The covariance of this estimate is calculated within the filter in every cycle of it and when it exceeds a preset threshold the robot is forced to stop. Then an absolute orientation measurement is acquired using the sun sensor and the three-axial accelerometer. A backward estimation is performed (off-line). Its results are combined with the ones from the forward filter within the smoother (off-line). Finally the position is being estimated (off-line) using the attitude estimate for each instance of the trajectory.
gyroscopes, a sun sensor and an accelerometer. As mentioned before, when the vehicle moves on the cratered, rocky slopes of the planet the gyros are the only source of attitude information. When it stops the accelerometer, that measures only the gravitational force now, determines the unit vector of the local vertical (roll and pitch information) and the sun sensor specifies the heading of the vehicle (yaw information). This is the absolute orientation of the vehicle and it is available only at stop.

The proposed method is summarized in the following algorithm: The vehicle is commanded to move for some time. A (forward) Indirect Kalman filter processes the incoming gyro values (angular velocities) and provides an initial real-time attitude estimate. The gyro values are stored for later processing. The covariance of the estimate is monitored and when that becomes unreliable the vehicle stops. Then, another absolute orientation measurement is collected and the smoother is activated. The first part of it which is the backward filter, initiates from the last measurement and propagates the orientation estimate backwards using the stored values for the angular velocities. Finally, the resulting estimates are combined with the previous ones (from the forward filter) and the smoothed attitude estimate is passed to the position estimating module.

This new method is not limited to robots used for extra-terrestrial exploration. It can be applied to any other autonomous vehicle equipped an equivalent set of sensors. The mixing of high frequency inertial sensors with usually lower frequency absolute (position or orientation) sensors is becoming very common in mobile robotics. Robots on earth equipped with GPS or landmark tracking devices, carry most of the time additional sensors that can be used for localization when the GPS signal becomes obsolete or the landmarks intractable. These added encoders or inertial sensors increase the overall redundancy of the system. The presented framework can be used to optimally combine them with the GPS for example.

### 6.1. Quaternions in attitude representation

The three-parameter Euler angle representation has been used in most applications of the Kalman filter in robot localization (Fuke & Krotkov 1996), (Bonnifait & Garcia 1996). However the kinematic equations for Euler angles involve non-linear and computationally expensive trigonometric functions. The computational cost using quaternions is less than using Euler angles.\(^5\) It is more compact because only four parameters, rather than nine, are needed. Furthermore in the Euler angle representation the angles become undefined for some rotations (the gimbal lock situation) which causes problems in Kalman filtering applications. Among all the representations for finite rotations, only those of four parameters behave well for arbitrary rotations. The reason is that a non-singular mapping between parameters and their corresponding rotational transformation matrix requires a set of four parameters at least. This issue was addressed by Stuelpnagel in 1964 (Stuelpnagel 1964).

Quaternions are an elegant and convenient parameterization of the attitude. A unit quaternion is a global non-singular four parameter representation also known as Euler symmetric parameters. Physical quantities pertaining to the motion of rotation such as angular displacement, velocity, acceleration and momentum are derived in terms of Euler parameters in a simple manner. Manipulating equations is much easier when using quaternion algebra (Chou 1992). The generalized commutative properties of quaternion multiplication are very useful when combining kinematic equations derived from successive rotations in space. Quaternions were invented by Hamilton in 1843 (Hamilton 1866). Their use in simulation of attitude dynamics was promoted by Robinson (Robinson 1958), and Mitchell and Rogers (Mitchell & Rogers 1965). The attitude matrix computed from a quaternion (as a quadratic function) is orthogonal if the sum of squares of the quaternion components is unity. If propagation errors result in a violation of this constraint the quaternion can be renormalized by dividing its components by (scalar) square root of the sum of their squares.\(^6\)

Some of the early applications of quaternions to strapdown guidance can be found in Wilcox (Wilcox 1967) and Mortenson (Mortenson 1974).

\(^5\)For quaternions there is a relative simple form for combining the parameters for two individual rotations to give the parameters of the product of the two rotations; 16 multiplications are required. In the case of Euler angles, combination of two individual rotations is done through attitude matrix multiplication; 27 multiplications are required.

\(^6\)Giardina et al. (Giardina, Bronson & Wallen 1975) showed that the attitude matrix computed from the renormalized quaternion is identical to the one given by their optimal orthogonalization.
6.2. Attitude kinematics

The physical counterparts of quaternions are the rotational axis \( \mathbf{n} \) and the rotational angle \( \theta \) that are used in the Euler theorem regarding finite rotations. That is taking the vector part of a quaternion and normalizing it, we can find the rotational axis right away, and from the last parameter we can obtain the angle of rotation (Craig 1989). Following the notation in (Lefferts, Markley & Shuster 1982), a unit quaternion is defined as

\[
q = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix}
\]  

(60)

with the constraint

\[
q^T q = 1
\]  

(61)

where

\[
q_1 = n_x \sin(\theta/2), q_2 = n_y \sin(\theta/2), q_3 = n_z \sin(\theta/2), q_4 = \cos(\theta/2)
\]  

(62)

and

\[
\mathbf{n} = \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix}
\]  

(63)

is the unit vector of the axis of rotation and \( \theta \) is the angle of rotation. The first three elements of the quaternion can be written in a compact form as:

\[
\mathbf{\tilde{q}} = \mathbf{n} \sin(\theta/2)
\]  

(64)

The attitude matrix is obtained from the quaternion according to the relation:

\[
A(q) = (|q|^2 - \mathbf{q}^T \mathbf{q}) I_{3 \times 3} + 2 \mathbf{q} \mathbf{q}^T + 2q_4 \begin{bmatrix} 0 & q_3 & -q_2 \\ -q_3 & 0 & q_1 \\ q_2 & -q_1 & 0 \end{bmatrix}
\]  

(65)

where

\[
\begin{bmatrix} 0 & q_3 & -q_2 \\ -q_3 & 0 & q_1 \\ q_2 & -q_1 & 0 \end{bmatrix}
\]  

(66)

is a \( 3 \times 3 \) skew symmetric matrix generated by the \( 3 \times 1 \) vector \( \mathbf{q} \). The matrix \( A(q) \) transforms representations of vectors in the reference coordinate system to representations in the body fixed coordinate system. The composition (product) of two quaternions is defined from the relation

\[
A(q')A(q) = A(q' \otimes q)
\]  

(67)

and it is

\[
q' \otimes q = \begin{bmatrix} q' \end{bmatrix} q
\]  

(68)

with
\[
\begin{bmatrix}
q' \\
\end{bmatrix} = 
\begin{bmatrix}
q_4' & q_3' & -q_2' & q_1' \\
-q_3' & q_4' & q_1' & q_2' \\
q_2' & -q_1' & q_3' & q_4' \\
-q_1' & -q_2' & -q_3' & q_4'
\end{bmatrix}
\]

(69)

The rate of change of the attitude matrix with time is given by:

\[
\frac{d}{dt} A(t) = \left[ \left[ \omega(t) \right] \right] A(t)
\]

(70)

where the corresponding rate for the quaternion is:

\[
\frac{d}{dt} q(t) = \frac{1}{2} \Omega(\omega(t)) q(t)
\]

(71)

with

\[
\Omega(\omega) = 
\begin{bmatrix}
0 & \omega_3 & -\omega_2 & \omega_1 \\
-\omega_3 & 0 & \omega_1 & \omega_2 \\
\omega_2 & -\omega_1 & 0 & \omega_3 \\
-\omega_1 & -\omega_2 & -\omega_3 & 0
\end{bmatrix}
\]

(72)

At this point we present an approximate body-referenced representation of the error state vector and covariance matrix. The error state includes the bias error and the quaternion error. The bias error is defined as the difference between the true and estimated bias.

\[
\Delta \tilde{b} = \tilde{b}_{true} - \tilde{b}_i
\]

(73)

The quaternion error is not the arithmetic difference between the true and estimated but it is expressed as the quaternion which must be composed with the estimated quaternion in order to obtain the true quaternion. That is:

\[
\delta q = q_{true} \otimes q_i^{-1}
\]

(74)

or

\[
q_{true} = \delta q \otimes q_i
\]

(75)

The advantage of this representation is that since the incremental quaternion corresponds very closely to a small rotation, the fourth component will be close to unity and thus the attitude information of interest is contained in the three vector component \(\delta \tilde{q}\) where

\[
\delta q \approx \begin{bmatrix}
\delta \tilde{q} \\
1
\end{bmatrix}
\]

(76)

Starting from equations:

\[
\frac{d}{dt} q_{true} = \frac{1}{2} \Omega(\tilde{\theta}_{true}) q_{true}
\]

(77)

and

\[
\frac{d}{dt} q_i = \frac{1}{2} \Omega(\tilde{\theta}_i) q_i
\]

(78)
\[
\frac{d}{dt} \delta \vec{q} = \left[ \left[ \vec{\omega}_m \right] \right] \delta \vec{q} - \frac{1}{2} (\Delta \vec{b} + \vec{n}_r) 
\]  
(79)

and

\[
\frac{d}{dt} \delta q_z = 0 
\]  
(80)

Using the infinitesimal angle assumption in equation (64), \( \delta \vec{q} \) can be written as

\[
\delta \vec{q} = \frac{1}{2} \delta \vec{\theta} 
\]  
(81)

and thus equation (79) can be rewritten as

\[
\frac{d}{dt} \delta \vec{\theta} = \left[ \left[ \vec{\omega}_m \right] \right] \delta \vec{\theta} - (\Delta \vec{b} + \vec{n}_r) 
\]  
(82)

Differentiating equation (73) and making the same assumptions for the true and estimated bias as in the previous section, equations (12) and (18), the bias error dynamic equation can be expressed as

\[
\frac{d}{dt} \Delta \vec{b} = \vec{n}_w 
\]  
(83)

Combining equations (82) and (83) we can describe the error state equation as

\[
\frac{d}{dt} \begin{bmatrix} \delta \vec{\theta} \\ \Delta \vec{b} \end{bmatrix} = \begin{bmatrix} \left[ \left[ \vec{\omega}_m \right] \right] & -I_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} \end{bmatrix} \begin{bmatrix} \delta \vec{\theta} \\ \Delta \vec{b} \end{bmatrix} + \begin{bmatrix} -I_{3 \times 3} \\ 0_{3 \times 3} \end{bmatrix} \begin{bmatrix} 0 \\ I_{3 \times 3} \end{bmatrix} \begin{bmatrix} \vec{n}_r \\ \vec{n}_w \end{bmatrix} 
\]  
(84)

or in a more compact form

\[
\frac{d}{dt} \Delta \vec{x} = \begin{bmatrix} F \end{bmatrix} \Delta \vec{x} + \begin{bmatrix} G \end{bmatrix} \vec{n} 
\]  
(85)

6.3. Discrete system: Indirect forward Kalman filter equations

6.3.1. Propagation

At this point we define as \( q_{k|k} (\vec{b}_{k|k}) \) the quaternion (bias) estimate at time \( t_k \) based on data up to and including \( z(t_k) \), \( q_{k|k-1} (\vec{b}_{k|k-1}) \) the quaternion (bias) estimate at time \( t_{k-1} \) propagated to \( t_k \), right before the measurement update at \( t_k \). The estimated angular velocity is defined as:

\[
\vec{\omega}_{k|k-1} = \vec{\omega}_m(t_k) - \vec{b}_{k|k-1} 
\]  
(86)

before the update, and as

\[
\vec{\omega}_{k|k} = \vec{\omega}_m(t_k) - \vec{b}_{k|k} 
\]  
(87)

right after the update.

The full estimated quaternion is propagated over the interval \( \Delta t_k = t_k - t_{k-1} \) according to the following equation (Wertz 1978):

\[
q_{k|k-1} = \left\{ \exp\left[ \frac{1}{2} \Omega(\vec{\omega}_{avg}) \Delta t_k \right] + \left[ \Omega(\vec{\omega}_{k|k-1}) \Omega(\vec{\omega}_{k|k-1} \Delta t_k) - \Omega(\vec{\omega}_{k|k-1}) \Omega(\vec{\omega}_{k|k-1}) \right] \Delta t_k^2 \right\} q_{k-1|k-1} 
\]  
(88)
where the average angular velocity for this interval is approximately
\[
\bar{\omega}_{\text{avg}} = \frac{\bar{\omega}_{k/k-1} + \bar{\omega}_{k-1/k-1}}{2}
\]  
(89)

The bias estimate is constant over the propagation interval
\[
\tilde{b}_{k/k-1} = \tilde{b}_{k-1/k-1}
\]  
(90)

The propagation equation for the error state covariance is
\[
P_{k/k-1} = \Phi(k, k - 1) P_{k-1/k-1} \Phi^T(k, k - 1) + Q_k
\]  
(91)

If the average angular velocity \(\bar{\omega}_{\text{avg}}\) is constant over the interval \(\Delta t_k\), with magnitude \(\omega_{\text{avg}}\) then the discrete system transition matrix \(\Phi(k, k - 1)\) can be calculated from equation (84) as:
\[
\Phi(k, k - 1) = \begin{bmatrix}
\exp \left( \left[ \begin{array}{c}
\bar{\omega}_{\text{avg}} \\
0_{3 \times 3}
\end{array} \right] \Delta t_k \right) & -\Psi(\Delta t_k) \end{bmatrix}
\]  
(92)

where the matrix \(\Psi\) is
\[
\Psi(\Delta t_k) = I_{3 \times 3} \Delta t_k + \left[ \begin{array}{c}
\bar{\omega}_{\text{avg}} \\
0_{3 \times 3}
\end{array} \right] (1 - \cos(\omega_{\text{avg}} \Delta t_k)) / \omega_{\text{avg}}^2 + \left[ \begin{array}{c}
\bar{\omega}_{\text{avg}} \\
0_{3 \times 3}
\end{array} \right]^2 (\omega_{\text{avg}} \Delta t_k - \sin(\omega_{\text{avg}} \Delta t_k)) / \omega_{\text{avg}}^3
\]  
(93)

Finally the system noise covariance matrix \(Q_k\) can be calculated from
\[
Q_k = \int_0^{
\Delta t_k}
\begin{bmatrix}
\sigma_n^2 I_{3 \times 3} + \sigma_n^2 \Phi(\tau) \Psi^T(\tau) & -\sigma_n^2 \Psi(\tau) \\
-\sigma_n^2 \Psi^T(\tau) & \sigma_n^2 I_{3 \times 3}
\end{bmatrix}
d\tau
\]  
(94)

### 6.3.2. Update

Whenever the rover stops an absolute orientation measurement is available from the sun sensor and the accelerometer. This is used to update the estimated error state and the covariance (Mendel 1987). The Kalman gain matrix is given by:
\[
K_k = P_{k/k-1} H_k^T \left( H_k P_{k/k-1} H_k^T + R_k \right)^{-1}
\]  
(95)

The updated covariance is:
\[
P_{k/k} = P_{k/k-1} - K_k H_k P_{k/k-1}
\]  
(96)

and the updated error state
\[
\Delta x_{k/k} = \Delta x_{k/k-1} + K_k \Delta z(t_k)
\]  
(97)

or
\[
\begin{bmatrix}
\delta \tilde{b}_{k/k} \\
\Delta \tilde{b}_{k/k}
\end{bmatrix} = K_k \Delta z(t_k)
\]  
(98)

where \(\Delta z(t_k)\) is the measurement residual. The propagated error \(\Delta x_{k/k-1}\) is zero because we have implemented the feedback formulation of the Indirect Kalman filter. Every time we have a measurement the update is included in the (full) state and therefore the next estimate of the error state is assumed to be zero. This update is:
\[ q_{k/k} = \delta q_{k/k} \otimes q_{k/k-1} = \left[ \frac{\delta \vec{q}_{k/k}}{1} \right] \otimes q_{k/k-1} \]  

where

\[ \delta \vec{q}_{k/k} = \frac{1}{2} \delta \vec{\theta}_{k/k} \]  

and

\[ \vec{b}_{k/k} = \vec{b}_{k/k-1} + \Delta \vec{b}_{k/k} \]

### 6.3.3. Observation model

The attitude sensors considered here are the sun sensor and the (three-axial) accelerometer both used when the rover is at stop. The first one measures the unit vector towards the sun and the second one provides the unit vector in the vertical direction. These measurements depend explicitly on the attitude but not on the gyroscopes’ biases.

Let the observed vector in the sensor frame be

\[ \vec{p}_S(t_k) = T_{S\leftarrow B} A(q(t_k)) \vec{p}_I + \vec{n}_p \]  

where \( \vec{p}_S \) is the measured unit vector in the sensor frame, \( \vec{p}_I \) is the reference vector in the inertial frame (pointing towards the direction of the sun for example - we assume that it does not change significantly for small time intervals), \( T_{S\leftarrow B} \) is the transformation matrix from the body frame to the sensor frame, \( A(q(t_k)) \) is the true orientation matrix of the robot at time \( t_k \), \( q(t_k) \) is the true attitude quaternion (\( q_{\text{true}} \)) at time \( t_k \) and \( \vec{n}_p \) is the sensor noise. If \( p_{S,k/k-1} \) is the measured unit vector estimate at time \( t_k \) and \( \Pi \) is the matrix that projects the measurements on a plane perpendicular to the sensor boresight, then the residual \( \Delta z(t_k) \) can be written as:

\[ \Delta z(t_k) = \Pi (p_S(t_k) - p_{S,k/k-1}) = \Pi (T_{S\leftarrow B} A(q(t_k)) \vec{p}_I + \vec{n}_p) - T_{S\leftarrow B} A(q_{k/k-1}) \vec{p}_I \]  

The true orientation matrix is

\[ A(q(t_k)) = A(\delta \vec{q}(t_k)) \otimes q_{k/k-1} = A(\delta \vec{q}(t_k)) A(q_{k/k-1}) \]  

From equation (65) we have:

\[ A(\delta \vec{q}(t_k)) = ([|\delta q_z(t_k)|^2 - \delta \vec{q}^T(t_k) \delta \vec{q}(t_k)] I_{3\times3} + 2 \delta \vec{q}(t_k) \delta \vec{q}^T(t_k) + 2 \delta q_z(t_k) \begin{bmatrix} \delta \vec{q}(t_k) \end{bmatrix}] \]  

Using the small rotation angle approximations

\[ \delta q_z(t_k) \simeq 1 \]  

\[ \delta \vec{q}^T(t_k) \delta \vec{q}(t_k) \simeq 0 \]  

\[ \delta \vec{q}(t_k) \delta \vec{q}^T(t_k) \simeq 0_{3\times3} \]  

we can write

\[ A(\delta \vec{q}(t_k)) \simeq I_{3\times3} + 2 \begin{bmatrix} \delta \vec{q}(t_k) \end{bmatrix} \]  

Substituting back in (104) we get

\[ A(q(t_k)) \simeq (I_{3\times3} + 2 \begin{bmatrix} \delta \vec{q}(t_k) \end{bmatrix}) A(q_{k/k-1}) = (I_{3\times3} + \begin{bmatrix} \delta \vec{\theta}(t_k) \end{bmatrix}) A(q_{k/k-1}) \]
Using this result in (103) we have similar results to those in (Sedlak & Chu 1993):

\[
\Delta z(t_k) = \Pi^T (T^* - B^* \begin{bmatrix} \delta \theta(t_k) \end{bmatrix}) A(q_k|k-1) + \tilde{n}_p = \begin{bmatrix} \frac{\dot{s}_x \times A(q_k|k-1) \mathbf{P}_t}{s_y \times A(q_k|k-1) \mathbf{P}_t} \end{bmatrix} \delta \theta(t_k) = h_{k/k-1} \delta \theta(t_k) + \Pi \tilde{n}_p \quad (111)
\]

where \( \dot{s}_x \) and \( \dot{s}_y \) are the unit vectors along the sensor axes. From the previous equation it is obvious that the observation model is non-linear and to calculate the sensitivity (measurement) matrix we have to derive the partial derivatives of \( \Delta z \) with respect to the estimated error states. The measurements are independent from the gyro biases and thus:

\[
H_k = \frac{\partial(\Delta z)}{\partial(\delta \theta)} = \begin{bmatrix} h_{k/k-1} & 0_{2 \times 3} \end{bmatrix} \quad (112)
\]

6.4. Backward filter

In the flow chart shown in figure 11 we see that the robot has to stop every time the uncertainty grows over a preset threshold. Then the backward filter is engaged and the last attitude estimate is propagated backwards in time. This last estimate is very precise because it is heavily based on the absolute orientation measurements acquired when the robot stopped. During the time instances preceding stopping the uncertainty has grown considerably and the information provided by the forward filter was of low quality. For this part of the trajectory the backward filter is crucially needed. It uses the stored gyro values and recalculates the orientation angles starting from a very precise initial estimate. While the backward filter is close to its starting point it is able to provide estimates of higher confidence than those of the forward filter.

In order to derive the equations for the 3-D backward Indirect Kalman filter we start again from the equations of the system for the forward case:

\[
\dot{x} = F \ x + G \ w \quad (113)
\]

\[
z = H \ x + v \quad (114)
\]

By defining

\[
\tau = T - t \quad (115)
\]

where \( \tau \) is the backward time variable and \( T = t_k - t_{k-M} \) is the time interval of smoothing, the backward system equation can be derived from:

\[
\frac{dx}{d\tau} = \frac{dx}{dt} \frac{dt}{d\tau} = -\dot{x} \quad (116)
\]

or

\[
\frac{dx_b}{d\tau} = -F \ x_b - G \ w \quad (117)
\]

Thus, equation (84) for the backward system becomes:

\[
\frac{d}{dt} \begin{bmatrix} \delta \theta_b \ 
\Delta \dot{\theta}_b \n\end{bmatrix} = \begin{bmatrix} - \begin{bmatrix} \omega_m \\
I_{3 \times 3} \\
0_{3 \times 3} 
\end{bmatrix} & I_{3 \times 3} & 0_{3 \times 3} \n0_{3 \times 3} & -I_{3 \times 3} & \tilde{n}_p \n0_{3 \times 3} & \tilde{n}_w \n\end{bmatrix} \Delta \theta_b + \begin{bmatrix} I_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \n0_{3 \times 3} & -I_{3 \times 3} & \tilde{n}_p \n0_{3 \times 3} & \tilde{n}_w \n\end{bmatrix} \Delta \theta_b \quad (118)
\]

and from this the discrete backward system transition matrix is calculated to be:

\[
\Phi(b, k - 1, k) = \exp\left(\begin{bmatrix} -\omega_{av} \\
0_{3 \times 3} \n\end{bmatrix} \Delta t_k \right) \begin{bmatrix} \Psi_b(\Delta t_k) & 0_{3 \times 3} \n0_{3 \times 3} & I_{3 \times 3} \n\end{bmatrix} \quad (119)
\]
with

\[ \Psi_b(\Delta t_k) = I_{3 \times 3} \Delta t_k - \left( [\bar{\omega}_{avg}] (1 - \cos(\bar{\omega}_{avg} \Delta t_k)) / \bar{\omega}_{avg}^2 + [\bar{\omega}_{avg}]^2 \right) (\bar{\omega}_{avg} \Delta t_k - \sin(\bar{\omega}_{avg} \Delta t_k)) / \bar{\omega}_{avg}^3 \] (120)

Instead of equation (88) the quaternion estimate is propagated according to:

\[ q_{b,k-1/k-1} = \exp [\frac{1}{2} \Omega(\bar{\omega}_{avg}) \Delta t_k] + [\Omega(\bar{\omega}_{k-1/k-1}) \Omega(\bar{\omega}_{k-1/k-1}) - \Omega(\bar{\omega}_{k-1/k-1}) \Omega(\bar{\omega}_{k-1/k-1})] \Delta t_k^2 / 48 \] (121)

The bias propagation remains the same as before (90). The direction of propagation does not affect an assumed constant variable which is:

\[ \tilde{b}_{b,k-1/k-1} = \tilde{b}_{b,k-1} \] (122)

The backward propagation equation for the covariance is now:

\[ P_{b,k-1/k-1} = \Phi_b(k-1,k) P_{b,k-1} \Phi_b^T(k-1,k) + Q_{b,k} \] (123)

where

\[ Q_{b,k} = \int_0^{\Delta t_k} \left[ \begin{array}{ccc} \sigma_f^2 I_{3 \times 3} + \sigma_w^2 \Psi_b(\tau) & \sigma_w^2 \Psi_b(\tau) \\ \sigma_w^2 \Psi_b^T(\tau) & \sigma_w^2 I_{3 \times 3} \end{array} \right] d\tau \] (124)

No more new absolute measurements are collected during the backward propagation of the filter and thus, the update equations and the observability model for the backward filter are neglected.

6.5. Smoother

The smoother problem consists of the construction of the best estimate of the state of a system over a time period using all the measurements in that time interval. The areas of application consist usually of non real-time cases as opposed to real-time applications that are concerned with filters (Reynolds 1990). A typical application of smoothers is in the post-flight construction of a trajectory or attitude profile of a spacecraft in order to support the integration of data from a space-borne sensor.

In our case, the time that the robot stops to get an absolute orientation measurement allows for post-processing and therefore application of the smoother. The 3-D case though presents some additional difficulties because of the particular form of the error quaternion used. In order to calculate the total (smoothed) estimate we have to use equation (59) that combines the forward and the backward one. Each of the error states has been defined as the state that if we add to the estimate it gives us the real state. This is not true for the quaternion error state which was defined in equation (74) as the state that has to be combined with the estimate to give the real attitude. To circumvent this problem we have to calculate \( \Delta q_{b,k} \) instead of \( \delta q_{b,k} \) at each step. Additionally the calculated covariance is:

\[ P_{5 \times 6} = E\left[ \begin{array}{c} \delta \tilde{q} \delta \tilde{q}^T \\ \delta \tilde{b} \delta \tilde{b}^T \end{array} \right] \] (125)

while what is required is:

\[ P_{7 \times 7} = E\left[ \begin{array}{ccc} \Delta q \Delta q^T & \Delta q \Delta \tilde{b}^T \\ \Delta \tilde{b} \Delta q^T & \Delta \tilde{b} \Delta \tilde{b}^T \end{array} \right] \] (126)

The solution to that (as derived in Appendix II) is to produce the desired quantities using the following equations:
Figure 12. In this figure are presented four different sequences of values for the first quaternion. The solid line is the true value, the dashed is the estimated from the forward filter, the dotted-dashed is the estimated from the backward filter and the dotted is the one produced by the smoother. The initial estimate for the backward filter (i.e. final absolute measurement at $t = 600$ secs) was very precise and thus for most of the time estimates produced by the backward filter are very close to the true values of $q_1$. The smoothed estimates also remain close to the true ones and they only diverge near $t = 300$ secs when the smoother weighs the forward filter estimates more. The smoothed estimate at $t = 400$ secs is almost solely depend on the (initial) absolute measurement at that time instant.
Figure 13. This is very usual outcome due to the bias estimation. The forward filter estimate drifts to the right because it has underestimated the gyro bias. The backward filter overestimates thus and thus drifts to the left (in the opposite direction). The smoothed estimate outperforms both filters minimizing the average estimation error.

\[
\begin{bmatrix}
\Delta q \\
\Delta \tilde{b}
\end{bmatrix} =
\begin{bmatrix}
\Xi(q) & 0_{3 \times 3} \\
0_{4 \times 3} & I_{3 \times 3}
\end{bmatrix}
\begin{bmatrix}
\delta \tilde{q} \\
\Delta \tilde{b}
\end{bmatrix}
\] (127)

or

\[
\Delta x = S(q) \delta x
\] (128)

and

\[
P_{7 \times 7} = S(q) P_{6 \times 6} S^T(q)
\] (129)

**7. FROM ATTITUDE ESTIMATES TO POSITION ESTIMATES**

The accuracy of the position estimate depends heavily on the accuracy of the attitude estimate. Though the position can be calculated in real-time using the output of the forward Kalman filter we choose not to do that. Instead in our algorithm the position estimation takes place off-line as described in (11). After the vehicle stops to collect an absolute orientation measurement the off-line smoothing of the attitude estimation is performed. The resulting
Figure 14. The solid line shows the true bias for the second gyroscope, the dashed the forward filter's estimate and the dashed-dotted the backward filter's estimate. The dotted line is the resulting bias estimate from the smoother. The smoothed (total) estimate stays close to the backward filter estimate for the second half of each smoothing interval while for the first part it depends on both the forward filter's estimate and the backward filter's estimate. This is due to the fact that the initial bias value for the backward filter is more trustworthy for this time interval than the initial value of the forward filter. The explanation is that the initial bias value for the backward filter is the one obtained at the end of the smoothing time interval, it is provided from the forward filter and it is based on the bias values during this particular interval. The forward filter's bias initial value is based on the bias values during the previous time interval and thus it is definitely less accurate.
Figure 15. The solid line represents the covariance related to $q_2$ as calculated from the forward filter. The dashed line is the same quantity from the backward filter, and finally the dashed dotted line is the $q_2$'s overall covariance due to the smoother. It is obvious that at all times the total covariance is lower than any of the corresponding ones calculated from the two filters. Its value remains bounded and varies slightly during the smoothing interval.

Figure 16. Position Estimation Block Diagram
estimate is accurate and is used to compute the current position. As we mentioned before the attitude estimate is an input to equations (1) and (2). Under the assumption that the integration step is very small, we can simplify this calculation according to figure 16. First the increase in position is calculated due to the sensed vehicle acceleration and the current velocity:

\[ (130) \]

this incremental is expressed in local coordinates and it has to be transformed to global coordinates

\[ (131) \]

before it can be used to compute the next position

\[ (132) \]

In order to make available at every step the corresponding velocity we also have to calculate its incremental during every measurement cycle. Thus,

\[ (133) \]

and

\[ (134) \]

The new velocity is

\[ (135) \]

This result has to be transformed to local coordinates before it is fed back for the next position update:

\[ (136) \]

8. CONCLUSIONS

In this report we decomposed the localization algorithm in attitude estimation and, subsequently, position estimation. A novel approach that incorporates a smoother was presented. An Indirect (error state) Kalman filter that incorporates inertial navigation and absolute measurements was developed for this purpose. The dynamic model was replaced by gyro modeling which relies on the integration of the kinematic equations. The two dimensional case was studied first. Observability issues and the effect of the filter on the sensory noise were examined. The error state equations for the three dimensional case were derived and used to formulate the filter’s time-variant system model and non-linear observation model. Quaternions were selected for the three dimensional attitude representation. This report contains detailed derivations of the quaternion algebra equations needed in a 3-D estimator. Finally, the improvement due to the proposed method was demonstrated in simulation. Uniformly smaller values of the covariance of the estimate were sustained throughout each of the trials.

9. FUTURE WORK

Sensor characterization: — Sun Sensor specifications.
Accelerometer specifications.
Application of the presented method in cases that the INS sensors are fused with other absolute sensors - measuring position (e.g. vision cues, star sensors, beacons etc)
APPENDIX I

In this section we define:

\[ \bar{\theta}_{true} = \bar{\omega}_m - \bar{b}_{true} - \bar{n}_e \]  

(137)

\[ \bar{\theta}_i = \bar{\omega}_m - \bar{b}_i \]  

(138)

We also repeat equations (74), (70) and (71):

\[ \delta q = q_{true} \otimes q_i^{-1} \]  

(139)

\[ \frac{d}{dt} A(t) = \left[ \left[ \bar{\omega}(t) \right] \right] A(t) \]  

(140)

\[ \frac{d}{dt} q(t) = \frac{1}{2} \Omega(\bar{\omega}(t))q(t) \]  

(141)

which we use to prove (79):

\[ \frac{d}{dt} \delta \bar{q} = \left[ \left[ \bar{\omega}_m \right] \right] \delta \bar{q} - \frac{1}{2}(\Delta \bar{b} + \bar{n}_e) \]  

(142)

First we rewrite equation (139) in terms of the corresponding attitude matrices as:

\[ A(\delta q) = A(q_{true} \otimes q_i^{-1}) = A(q_{true})A(q_i^{-1}) \]  

(143)

differentiating with respect to time we have:

\[ \frac{d}{dt} A(\delta q) = \frac{d}{dt} (A(q_{true}))A(q_i^{-1}) + A(q_{true}) \frac{d}{dt} (A(q_i^{-1})) \]  

(144)

From

\[ 1 = q_i \otimes q_i^{-1} \]  

(145)

we have

\[ I_{3 \times 3} = A(q_i \otimes q_i^{-1}) = A(q_i)A(q_i^{-1}) \]  

(146)

which with differentiation with respect to time gives:

\[ \frac{d}{dt} (A(q_i))A(q_i^{-1}) + A(q_i) \frac{d}{dt} (A(q_i^{-1})) = 0 \]  

(147)

or

\[ \frac{d}{dt} (A(q_i^{-1})) = -A^{-1}(q_i) \frac{d}{dt} (A(q_i))A(q_i^{-1}) \]  

(148)

or using \( A^{-1}(q_i) = A(q_i^{-1}) \)

\[ \frac{d}{dt} (A(q_i^{-1})) = -A(q_i^{-1}) \frac{d}{dt} (A(q_i))A(q_i^{-1}) \]  

(149)
Substituting in equation (144) equation (149) we have:

$$\frac{d}{dt}A(\delta q) = \frac{d}{dt}(A(q_{\text{true}}))A(q_i^{-1}) - A(q_{\text{true}})A(q_i^{-1}) \frac{d}{dt}(A(q_i))A(q_i^{-1})$$

(150)

or by using (140) with the appropriate rotational velocity vectors:

$$\frac{d}{dt}A(\delta q) = \left[\begin{array}{c} \dot{\theta}_{\text{true}} \\ 0 \end{array}\right] A(q_{\text{true}})A(q_i^{-1}) - A(q_{\text{true}})A(q_i^{-1}) \left[\begin{array}{c} \dot{\theta}_i \\ 0 \end{array}\right] A(q_i)A(q_i^{-1})$$

(151)

Combining the matrix products we have:

$$\frac{d}{dt}A(\delta q) = \left[\begin{array}{c} \dot{\theta}_{\text{true}} \\ 0 \end{array}\right] A(q_{\text{true}} \otimes q_i^{-1}) - A(q_{\text{true}} \otimes q_i^{-1}) \left[\begin{array}{c} \dot{\theta}_i \\ 0 \end{array}\right] A(q_i \otimes q_i^{-1})$$

(152)

or

$$\frac{d}{dt}A(\delta q) = \left[\begin{array}{c} \dot{\theta}_{\text{true}} \\ 0 \end{array}\right] A(\delta q) - A(\delta q) \left[\begin{array}{c} \dot{\theta}_i \\ 0 \end{array}\right]$$

(153)

The previous equation written in quaternion form is:

$$\frac{d}{dt} \delta q = \frac{1}{2} \left[\begin{array}{c} \dot{\omega}_m - \tau_{\text{true}} - \tilde{n}_r \\ 0 \end{array}\right] \otimes \delta q - \delta q \otimes \left[\begin{array}{c} \dot{\omega}_m - \tilde{\omega}_i \\ 0 \end{array}\right]$$

(154)

which using equations (137) and (138) becomes:

$$\frac{d}{dt} \delta q = \frac{1}{2} \left[\begin{array}{c} \dot{\omega}_m - \tau_{\text{true}} - \tilde{n}_r \\ 0 \end{array}\right] \otimes \delta q - \delta q \otimes \left[\begin{array}{c} \dot{\omega}_m - \tilde{\omega}_i \\ 0 \end{array}\right]$$

(155)

or

$$\frac{d}{dt} \delta q = \frac{1}{2} \left[\begin{array}{c} \dot{\omega}_m \\ 0 \end{array}\right] \otimes \delta q - \delta q \otimes \left[\begin{array}{c} \dot{\omega}_m \\ 0 \end{array}\right] - \frac{1}{2} \left[\begin{array}{c} \tau_{\text{true}} + \tilde{n}_r \\ 0 \end{array}\right] \otimes \delta q - \delta q \otimes \left[\begin{array}{c} \tilde{\omega}_i \\ 0 \end{array}\right]$$

(156)

At this point we need to use a more compact form of combining two quaternion vectors. If

$$a = \left[\begin{array}{c} \tilde{a} \\ a_0 \end{array}\right], b = \left[\begin{array}{c} \tilde{b} \\ b_0 \end{array}\right]$$

(157)

then

$$a \otimes b = \left[\begin{array}{c} a_0 \tilde{b} + b_0 \tilde{a} - \tilde{a} \times \tilde{b} \\ a_0 b_0 - \tilde{a}^T \tilde{b} \end{array}\right]$$

(158)

Applying these in the first term of (156) we have:

$$\begin{align*}
\left[\begin{array}{c} \tilde{\omega}_m \\ 0 \end{array}\right] & \otimes \left[\begin{array}{c} \delta \tilde{q} \\ \delta q_s \end{array}\right] - \left[\begin{array}{c} \delta \tilde{q} \\ \delta q_s \end{array}\right] \otimes \left[\begin{array}{c} \tilde{\omega}_m \\ 0 \end{array}\right] = \\
\left[\begin{array}{c} \tilde{\omega}_m \delta q + \delta \tilde{q} \times \tilde{\omega}_m \\ 0 \end{array}\right] & - \left[\begin{array}{c} \dot{\omega}_m \delta q + \delta \tilde{q} \times \tilde{\omega}_m \\ 0 \end{array}\right] = \\
\left[\begin{array}{c} \tilde{\omega}_m \delta q + \delta \tilde{q} \times \tilde{\omega}_m \\ 0 \end{array}\right] & - \left[\begin{array}{c} \dot{\omega}_m \delta q + \delta \tilde{q} \times \tilde{\omega}_m \\ 0 \end{array}\right]
\end{align*}$$

36
\[ -2 \left[ \begin{array}{c} \ddot{\omega}_m \\ \delta \vec{q} \end{array} \right] \]  \hfill (159)

Similarly the second part of (156) can be written as:

\[ \left[ \begin{array}{c} \ddot{b}_{true} + \vec{n}_r \\ 0 \end{array} \right] \otimes \left[ \begin{array}{c} \delta \dot{q} \\ \delta q_4 \end{array} \right] - \left[ \begin{array}{c} \delta \dot{q} \\ \delta q_4 \end{array} \right] \otimes \left[ \begin{array}{c} \ddot{b}_i \\ 0 \end{array} \right] = \]

\[ \left[ \begin{array}{c} 0 \delta \dot{q} + \delta q_4 (\ddot{b}_{true} + \vec{n}_r) - (\ddot{b}_{true} + \vec{n}_r) \times \delta \dot{q} \\ 0 \delta q_4 - (\ddot{b}_{true} + \vec{n}_r)^T \delta \dot{q} \end{array} \right] - \left[ \begin{array}{c} \delta q_4 \ddot{b}_i + 0 \delta \dot{q} - \delta \dot{q} \times \ddot{b}_i \\ \delta q_4 0 - \delta \dot{q}^T \ddot{b}_i \end{array} \right] = \]

\[ \left[ \begin{array}{c} \delta \omega \\ 0 \end{array} \right] \delta q_4 - \left[ \begin{array}{c} \delta \omega \times \delta \dot{q} \\ -\delta \omega^T \delta \dot{q} \end{array} \right] \]

\[ \hfill (160) \]

where

\[ \delta \omega = \ddot{b}_{true} - \ddot{b}_i + \vec{n}_r = \Delta \ddot{b} + \vec{n}_r \]  \hfill (161)

As we have mentioned before for very small rotations we can assume that \( \delta q_4 \approx 1 \) and thus equation (160) can be simplified as:

\[ \left[ \begin{array}{c} \ddot{b}_{true} + \vec{n}_r \\ 0 \end{array} \right] \otimes \left[ \begin{array}{c} \delta \dot{q} \\ \delta q_4 \end{array} \right] - \left[ \begin{array}{c} \delta \dot{q} \\ \delta q_4 \end{array} \right] \otimes \left[ \begin{array}{c} \ddot{b}_i \\ 0 \end{array} \right] \approx \left[ \begin{array}{c} \delta \omega \\ 0 \end{array} \right] - \phi (|\ddot{\omega}| |\delta \dot{q}|) \approx \left[ \begin{array}{c} \delta \omega \\ 0 \end{array} \right] \]  \hfill (162)

Now substituting equations (159) and (162) in (156) we have:

\[ \frac{d}{dt} \left[ \begin{array}{c} \delta \dot{q} \\ \delta q_4 \end{array} \right] = \frac{1}{2} (-2) \left[ \begin{array}{c} \ddot{\omega}_m \times \delta \dot{q} \\ 0 \end{array} \right] + \frac{1}{2} \left[ \begin{array}{c} \delta \omega \times \delta \dot{q} \\ 0 \end{array} \right] = \left[ \begin{array}{c} \ddot{\omega}_m \times \delta \dot{q} \\ 0 \end{array} \right] + \frac{1}{2} \left[ \begin{array}{c} \delta \omega \times \delta \dot{q} \\ 0 \end{array} \right] \]  \hfill (163)

Separating the vector from the scalar terms the previous equation is described by:

\[ \frac{d}{dt} \delta \dot{q} = \left[ \begin{array}{c} \ddot{\omega}_m \end{array} \right] \delta \dot{q} - \frac{1}{2} (\Delta \ddot{b} + \vec{n}_r) \]  \hfill (164)

and

\[ \frac{d}{dt} \delta q_4 = 0 \]  \hfill (165)

**APPENDIX II**
ACKNOWLEDGMENTS

The authors would like to thank B. Balaram, S. Hayati, G. Rodriguez, P. Shanker for insightful discussions over the past years. The authors also feel indebted to D. Bayard for gratefully devoting time and thought to this research effort.

REFERENCES


