

## **Estimation of attitudes from a low-cost miniaturized inertial platform using Kalman Filter-based sensor fusion algorithm**

N SHANTHA KUMAR<sup>1</sup> and T JANN<sup>2</sup>

<sup>1</sup>Flight Mechanics and Control Division, National Aerospace Laboratories, Bangalore 560 017, India

<sup>2</sup>DLR, Institute for Flugsystemtechnik, Braunschweig, Germany  
e-mail: nskumar@css.nal.res.in

**Abstract.** Due to costs, size and mass, commercially available inertial navigation systems are not suitable for small, autonomous flying vehicles like ALEX and other UAVs. In contrast, by using modern MEMS (or of similar class) sensors, hardware costs, size and mass can be reduced substantially. However, low-cost sensors often suffer from inaccuracy and are influenced greatly by temperature variation. In this work, such inaccuracies and dependence on temperature variations have been studied, modelled and compensated in order to reach an adequate quality of measurements for the estimation of attitudes. This has been done applying a Kalman Filter-based sensor fusion algorithm that combines sensor models, error parameters and estimation scheme. Attitude estimation from low-cost sensors is first realized in a Matlab/Simulink platform and then implemented on hardware by programming the micro controller and validated. The accuracies of the estimated roll and pitch attitudes are well within the stipulated accuracy level of  $\pm 5^\circ$  for the ALEX. However, the estimation of heading, which is mainly derived from the magnetometer readings, seems to be influenced greatly by the variation in local magnetic field.

**Keywords.** Estimation of attitudes; sensor fusion algorithm; inertial navigation systems; Kalman Filters low-cost sensors; miniaturized inertial platform.

### **1. Introduction**

Inertial measurement units (IMUs) have the ability to compute attitude in dynamic environments. When combined with GPS, the IMU/GPS system can provide critical information needed for navigation. A traditional IMU system typically consists of a set of spinning-mass vertical gyros that provide analog outputs. They are usually heavy and expensive to maintain. Furthermore, due to its analog nature, the traditional IMU requires data conversion when interfacing to a sophisticated digital navigation system, thereby increasing the overall system cost and complexity. High performance inertial-grade IMUs used in avionics systems exhibit

---

A list of symbols and abbreviations is given at the end of the paper

superior performance specifications but, because of the high cost of such units, their use in many applications is prohibitive.

A combination of low-cost MEMS sensors and digital signal processing techniques can provide an inexpensive and compact-sized adaptable alternative to existing IMUs. Closely coupled integration of the sensors, data acquisition elements and a Kalman Filter-based fusion algorithm allow the MEMS IMU to provide an accurate representation of the attitude of flight vehicles with performance comparable to older technology IMUs. Furthermore, the digital architecture's flexible interface allows easy integration of MEMS IMUs into most applications.

This paper describes how attitudes are derived accurately from the low-cost sensor platform using Kalman Filter-based sensor fusion algorithm.

## 2. Objective and approach

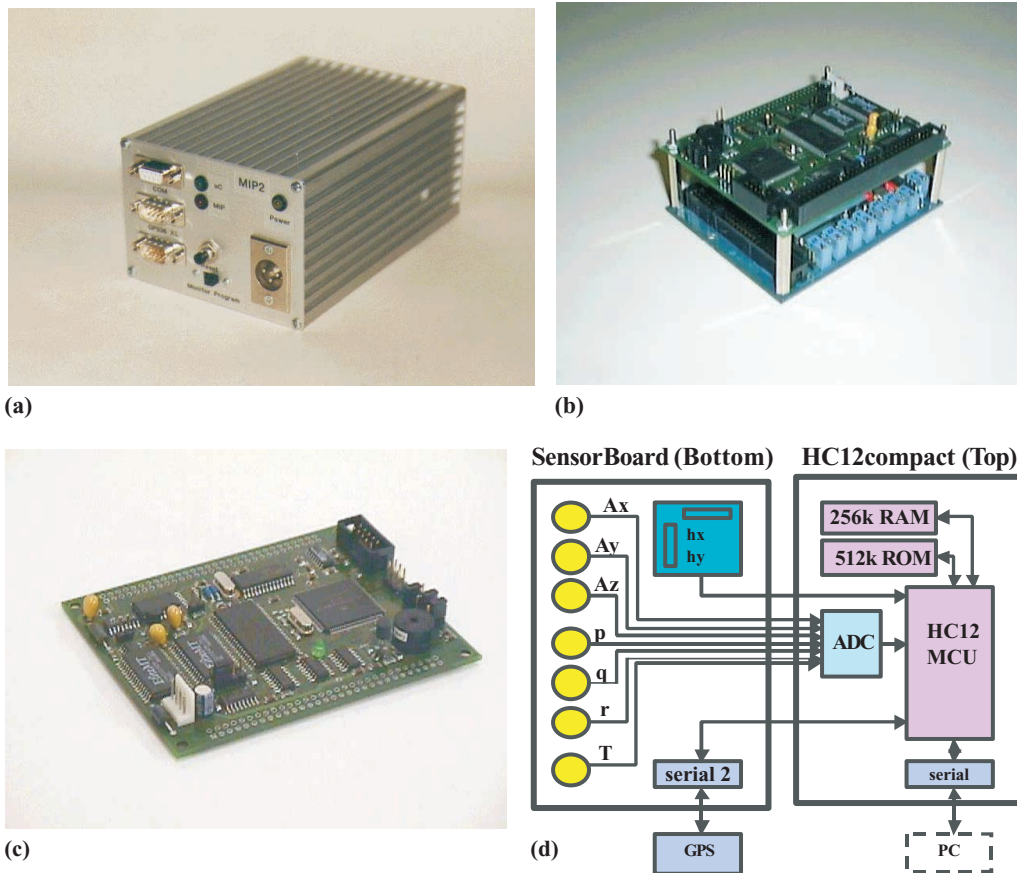
At the Institute of Flight Systems of the German Aerospace Center (DLR), two small and extensively instrumented vehicles ALEX-I and ALEX-II (shown in figure 1) were developed and flight tested in order to identify the dynamic behaviour of a parafoil-load system and to investigate guidance, navigation and control (GNC) concepts for autonomous landing (Doherr & Jann 1997; Jann *et al* 1999; Jann 2001).

As a spin-off of this project, a small sensor platform called MIP (miniaturized inertial platform) shown in figure 2 and consisting of small, low-cost accelerometers, rate gyros, magnetometers and a temperature transducer was developed. The main objectives of this small, low-cost MIP is to reduce the overall weight of the ALEX and to prove the low-cost sensor technology for GNC application for small aerial vehicles like ALEX and other UAVs.

For ALEX, as data from the GPS receiver are available, deriving the position information from this small sensor box is not that critical. In contrast, attitudes are required to be computed from the output of the three angular rate sensors in the platform using the strapdown algorithm. Due to integration, measurement errors are accumulated, if no corrections are applied and the Kalman Filter provides this correction in a very optimal way.



**Figure 1.** ALEX in flight.



**Figure 2.** (a) MIP. (b) Inside view of MIP. (c) HC12 Compact board. (d) Connection diagram of MIP.

As independent measurements, the accelerometers provide roll and pitch attitude reference using gravity. The magnetometer called V2X (vector 2X) is a two-axis magnetometer that measures the magnetic field in a single plane created by its two sensors, which are perpendicular to each other on the board. That is, it measures the earth’s magnetic field along the body  $X$ – $Y$  plane. If the earth’s magnetic field is known at the particular place in the geodetic plane, the same can be related to the measured magnetic field along the body axis through Euler angles that can be estimated. Therefore, fusing together the accelerometers’ and magnetometer’s measurements provides sufficient information to estimate all the three attitudes. An extended Kalman Filter is used to fuse measurements of the accelerometers and the magnetometer and estimate the attitudes. In this scheme the accelerometers and magnetometer provide an attitude reference and the filter provides corrections to the attitude trajectory calculated from integration of the rate gyros output. To compensate for the inaccuracies present in the low-cost sensors, they have been modelled according to the calibration tables and used in the filter.

The Kalman Filter-based fusion algorithm for the estimation of attitudes from low-cost MIP is first realized and studied in a Matlab/Simulink environment and then the algorithm is implemented on the hardware by programming micro-controller (Motorola, HC12 compact) enclosed inside the MIP box and tested by subjecting the MIP to pure angular motion.

### 3. Miniaturized inertial platform (MIP)

MIP is a miniaturized multi sensor inertial platform, designed with the main intention of reducing the weight of the ALEX and also to prove the low-cost sensor technology for autonomous systems like ALEX.

#### 3.1 Architecture

Figure 2 shows the MIP and its architecture. The hardware of the MIP consists of two motherboards stacked one above the other. The bottom board, called the sensor board, consists of all the sensors, and the top board, called the micro-controller board, consists of a micro-controller HC12 compact, which is used to acquire and process the sensors output data.

#### 3.2 Sensors

The sensor board of the MIP consists of three orthogonally mounted rate gyros, three orthogonally mounted linear accelerometers, a temperature transducer and a two-axis magnetometer with two sensors along the body  $X$ - $Y$  axis. A temperature transducer is provided in the MIP to account for the temperature variation effect on the other sensors. All these sensors are low-cost, and of small size (of MEMS class) brought from aerospace consumer bulk market. A GPS receiver connected to the MIP via one of the serial communication interfaces (SCI) provides the position information.

#### 3.3 Micro-controller

The data-processing unit of MIP is the HC12 compact with 16 MHz clock speed. It is a universal micro-controller module on the basis of a motorola MC68HC812A4 micro-controller unit (MCU). Figure 2 shows the HC12 compact.

The MC68HC812A4 MCU (Valvano 2000) is a 16-bit device composed of standard on-chip peripheral modules like a 16-bit CPU12, a Lite integration module (LIM), two asynchronous SCIs (SCI0 and SCI1), an SPI, a timer, an 8-bit ADC, a 1-kilobyte RAM, a 4-kilobyte EEPROM and a memory expansion logic with chip selects, key wakeup ports and a PLL.

In addition to the above on-chip features of the MCU, the HC12 compact has the following peripheral units:

- 512 KB flash memory and 256 KB RAM
- 12-bit, 11 channels ADC
- 12-bit, 2 channels DAC
- CAN controller
- RS232 interface driver
- Beeper
- Indicator LED

The outputs of the rate gyros, accelerometers and temperature transducer are analog and are connected to the ADC of the micro-controller board whereas the output of the magnetometer is digital and is connected directly to the micro-controller as shown in the connection diagram of figure 2.

#### 4. Sensor modelling

Since sensors of low-cost category suffer from inaccuracy and are influenced greatly by the temperature variation, such inaccuracies and dependency on temperature variations have been studied, modelled and compensated in order to reach an adequate quality of the measurements. Based on the calibration data, the sensors misalignment, temperature drift and CG offsets are modelled as given below. Thus, these models in reality represent the true sensor model and are included as a SIMULINK block for generating the sensor output and the inverse model is used in the filter for estimating the attitudes.

##### 4.1 Misalignment error

$$\begin{bmatrix} A_x \\ A_y \\ A_z \end{bmatrix}_{\text{Meas}} = \begin{bmatrix} 1.0021 & 0.0188 & -0.0146 \\ -0.0093 & 1.0008 & 0.0160 \\ 0.0401 & -0.0137 & 0.9998 \end{bmatrix} \begin{bmatrix} A_x \\ A_y \\ A_z \end{bmatrix}_{\text{true}}$$

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix}_{\text{Meas}} = \begin{bmatrix} 1.0294 & -0.0024 & 0.0157 \\ 0.0139 & 1.0137 & 0.0100 \\ -0.0091 & -0.0254 & 0.9838 \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix}_{\text{true}} \quad (1)$$

##### 4.2 Temperature drift

$$\begin{bmatrix} A_x \\ A_y \\ A_z \end{bmatrix}_{\text{Meas}} = \begin{bmatrix} A_x \\ A_y \\ A_z \end{bmatrix}_{\text{True}} + \begin{bmatrix} -0.0049T & +0.2955 \\ -0.0179T & +0.4348 \\ -0.0005T^2 & +0.0425T & -1.0268 \end{bmatrix}$$

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix}_{\text{Meas}} = \begin{bmatrix} p \\ q \\ r \end{bmatrix}_{\text{True}} + \begin{bmatrix} 4.1420 \times 10^{-5}T^3 & -0.0039T^2 + 0.0876T - 0.3803 \\ & 0.0018T^2 - 0.0236T - 0.5370 \\ 1.7881 \times 10^{-4}T^3 & -0.0150T^2 + 0.3564T - 2.2526 \end{bmatrix} \quad (2)$$

where  $T$  is the temperature in  $^{\circ}\text{C}$  measured from temperature transducer.

##### 4.3 CG offset

$$\begin{bmatrix} A_x \\ A_y \\ A_z \end{bmatrix}_{\text{Meas}} \cong \begin{bmatrix} A_x \\ A_y \\ A_z \end{bmatrix}_{\text{True}} + \begin{bmatrix} -(q^2 + r^2) X_{A_x} \\ -(p^2 + r^2) Y_{A_y} \\ -(p^2 + q^2) Z_{A_z} \end{bmatrix} \quad (3)$$

#### 5. Kalman Filter and multi sensor data fusion

As mentioned earlier, attitudes are computed from the output of the three rate sensors using a strapdown algorithm and, due to integration, measurement errors accumulate, if no correction occurs. Kalman Filter provides this correction in a very optimal way and is the ideal tool for fusing the multiple sensors output data in arriving at the accurate estimation of states (attitudes).

Simultaneous estimation of the states and unknown parameters like sensor bias is possible by augmenting the state vector with the unknown parameters and applying filtering algorithm to the augmented nonlinear model. This extended Kalman Filter (EKF) is a recursive algorithm that estimates system states  $X$  (attitudes and sensor bias) and state error covariance  $P$  by using the measured system output  $Z_m$  (accelerometers and magnetometers) and the known input  $U_m$  (rates) in the specified nonlinear system and output models with assumed process and measurement noise statistics  $Q$  and  $R$  respectively.

In EKF, the nonlinear system is represented by the nonlinear state equation as,

$$\dot{X}(t) = f[X(t), U_m(t)] + G_A w(t); X(0) = X_0, \quad (4)$$

and the measurement equation as,

$$Z_m(t) = h[X(t), U_m(t)] + v(t). \quad (5)$$

Equations (4) and (5) are first linearized about the prior best estimates of the states at each instant of time by finite difference method as,

$$A_{(k)} = \left[ \frac{\delta f}{\delta X} \right]_{X_{(k)}, U_{m(k)}}; B_{(k)} = \left[ \frac{\delta f}{\delta U_m} \right]_{X_{(k)}, U_{m(k)}}; H_{(k)} = \left[ \frac{\delta h}{\delta X} \right]_{X_{(k)}, U_{m(k)}}. \quad (6)$$

The linearized system is then discretised in time by computing the system state transition matrix  $\Phi_{(k+1,k)}$  from  $A_{(k)}$  as (Collinson 1998),

$$\Phi_{(k+1,k)} = e^{A_{(k)}\Delta t} \cong [I + A_{(k)}\Delta t], \quad (7)$$

where  $\Delta t$  is the sampling time.

The EKF consists of the time update (prediction step) and measurement update (correction step). In time update, the time propagation of states and the state error covariance matrix are obtained by the following equations,

$$\tilde{X}_{(k+1)} = \hat{X}_{(k)} + \int f[X(t), U_m(t)] dt, \quad (8)$$

$$\tilde{P}_{(k+1)} = \Phi_{(k+1,k)} \hat{P}_{(k)} \Phi_{(k+1,k)}^T + G_{A(k)} Q G_{A(k)}^T. \quad (9)$$

In the correction step, the measurement update of states and the state error covariance matrix are obtained by using measurements as and when they are available as given below,

$$\hat{X}_{(k+1)} = \tilde{X}_{(k+1)} + K_{(k+1)} [Z_{m(k+1)} - h\{\tilde{X}_{(k+1)}\}], \quad (10)$$

$$\hat{P}_{(k+1)} = [I - K_{(k+1)} H_{(k+1)}] \tilde{P}_{(k+1)}, \quad (11)$$

where  $K_{(k+1)}$  is the Kalman gain at instant  $k + 1$  which is given by,

$$K_{(k+1)} = \tilde{P}_{(k+1)} H_{(k+1)}^T [H_{(k+1)} \tilde{P}_{(k+1)} H_{(k+1)}^T + R]^{-1}. \quad (12)$$

The state integration, (8), is done by fourth order Runge–Kutta method.

In measurement update, the measurements are fused and the states are updated.

### 5.1 Kalman Filter tuning and performance check

In order to get the optimal estimate of the states, the Kalman Filter is tuned manually with  $Q$  relatively lower than  $R$  (Schultz & Melsa 1967). Once the filter is tuned, its performance is checked by verifying innovation sequence for zero mean and whiteness (Anderson & Moore 1979; Candy 1987). As per these conditions, the 95% of the residual should lie within the  $\pm 2R_{e(k)}^{1/2}$  bounds, where

$$R_{e(k)} = H_{(k)} \tilde{P}_{(k)} H_{(k)}^T + R, \quad (13)$$

and the autocorrelation of the residual should lie within the  $\pm 1.96/N^{1/2}$  where  $N$  is the number of samples.

Along with this the estimated states with the standard deviation given by  $\pm \hat{P}_{(k)}^{1/2}$  and the filter model output with the output measurements are plotted and checked to ensure the performance of the filter.

## 6. Matlab/Simulink implementation

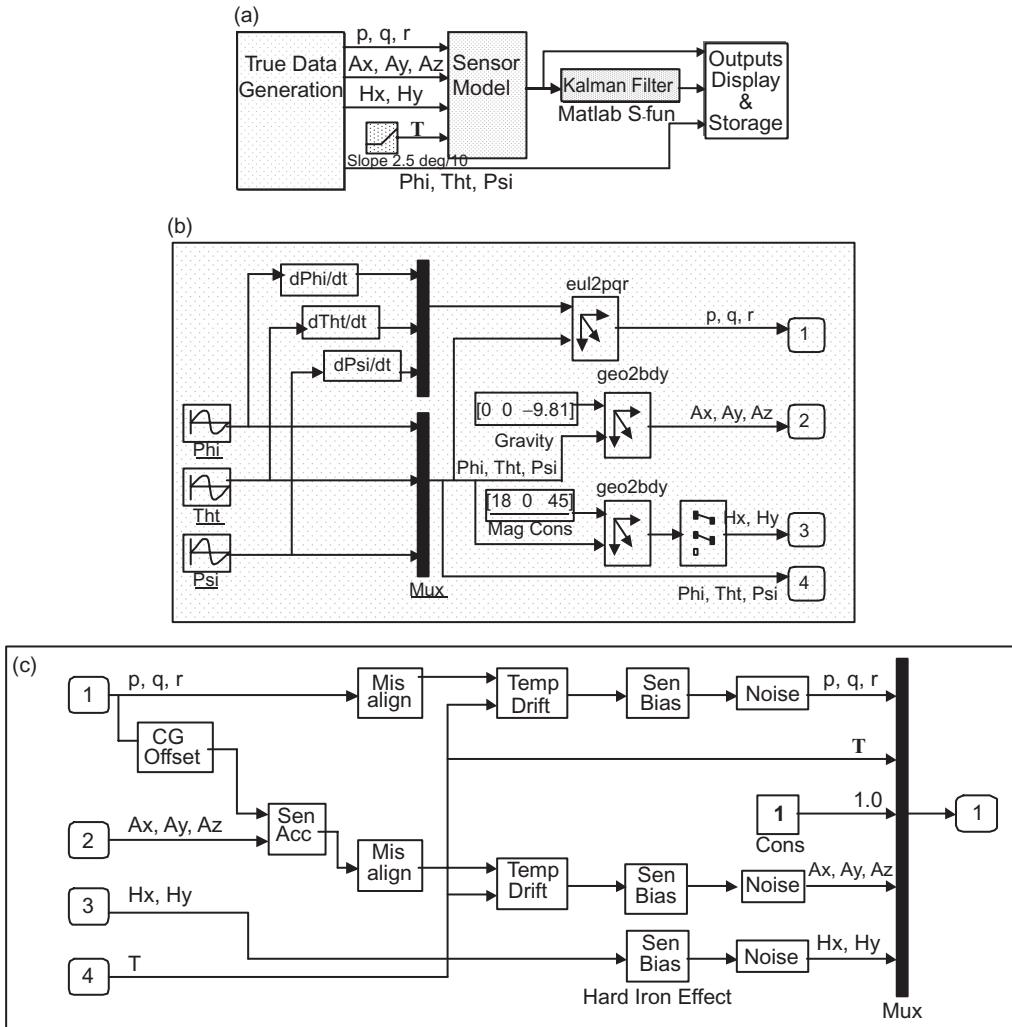
The Kalman Filter-based fusion algorithm scheme for the derivation of attitudes from the MIP is first realized and studied in Matlab/Simulink environment. The SIMULINK blocks are shown in figure 3. Kalman Filter algorithm is introduced into the SIMULINK block as S-function. For improved accuracy and to avoid singularities when dealing with cosine rotation matrix, a quaternion formulation is used in the filter model for attitude propagation.

### 6.1 Kalman Filter performance with 12-state model

The 12-state mathematical model consisting of 4 quaternion states and 8 augmented states with 6 observations is proposed initially for attitude estimation as given below.

*State model:*

$$\begin{aligned} \dot{q}_0 &= \frac{1}{2} [-q_1 p_b - q_2 q_b - q_3 r_b], \\ \dot{q}_1 &= \frac{1}{2} [q_0 p_b + q_2 r_b - q_3 q_b], \\ \dot{q}_2 &= \frac{1}{2} [q_0 q_b - q_1 r_b + q_3 p_b], \\ \dot{q}_3 &= \frac{1}{2} [q_0 r_b + q_1 q_b - q_2 p_b], \\ \dot{B}_p &= 0.0, \\ \dot{B}_q &= 0.0, \\ \dot{B}_r &= 0.0, \\ \dot{B}_{Ax} &= 0.0, \\ \dot{B}_{Ay} &= 0.0, \\ \dot{B}_{Az} &= 0.0, \\ \dot{B}_{Hx} &= 0.0, \\ \dot{B}_{Hy} &= 0.0, \end{aligned}$$



**Figure 3.** (a) Main Simulink Blocks with Kalman Filter as Matlab S-function. (b) Simulink blocks for true data generation. (c) Simulink blocks for sensor model.

where,

$$\begin{bmatrix} p_b \\ q_b \\ r_b \end{bmatrix} = \begin{bmatrix} 0.9713 & 0.0019 & -0.0155 \\ -0.0134 & 0.9862 & -0.0098 \\ 0.0086 & 0.0255 & 1.0161 \end{bmatrix} \begin{bmatrix} p_2 \\ q_2 \\ r_2 \end{bmatrix},$$

(accounting for misalignment, inverse of (1)),

$$\begin{bmatrix} p_2 \\ q_2 \\ r_2 \end{bmatrix} = \begin{bmatrix} p_1 \\ q_1 \\ r_1 \end{bmatrix} - \begin{bmatrix} 4.1420 \times 10^{-5} T_m^3 & -0.0039 T_m^2 & +0.0876 T_m & -0.3803 \\ & 0.0018 T_m^2 & -0.0236 T_m & -0.5370 \\ 1.7881 \times 10^{-4} T_m^3 & -0.0150 T_m^2 & +0.3564 T_m & -2.2526 \end{bmatrix},$$

(accounting for temperature variation, (2)),



$$\begin{bmatrix} p_1 \\ q_1 \\ r_1 \end{bmatrix} = \begin{bmatrix} p_m \\ q_m \\ r_m \end{bmatrix} - \begin{bmatrix} B_p \\ B_q \\ B_r \end{bmatrix}, \text{ (accounting for estimated sensor bias).}$$

Measurement model:

$$\begin{aligned} 1 \cdot 0 &= [q_0^2 + q_1^2 + q_2^2 + q_3^2] \\ A_{x_m} &= A_{x_3} + B_{Ax} \\ A_{y_m} &= A_{y_3} + B_{Ay} \\ A_{z_m} &= A_{z_3} + B_{Az} \\ H_{x_m} &= H_{x_b} + B_{Hx} \\ H_{y_m} &= H_{y_b} + B_{Hy} \end{aligned}$$

where,

$$\begin{bmatrix} A_{x_3} \\ A_{y_3} \\ A_{z_3} \end{bmatrix} = \begin{bmatrix} A_{x_2} \\ A_{y_2} \\ A_{z_2} \end{bmatrix} + \begin{bmatrix} -0.0049T_m & +0.2955 \\ -0.0179T_m & +0.4348 \\ -0.0005T_m^2 & +0.0425T_m & -1.0268 \end{bmatrix},$$

(accounting for temperature variation, (2))

$$\begin{bmatrix} A_{x_2} \\ A_{y_2} \\ A_{z_2} \end{bmatrix} = \begin{bmatrix} 1.0021 & 0.0188 & -0.0146 \\ -0.0093 & 1.0008 & 0.0160 \\ 0.0401 & -0.0137 & 0.9998 \end{bmatrix} \begin{bmatrix} A_{x_1} \\ A_{y_1} \\ A_{z_1} \end{bmatrix},$$

(accounting for misalignment, (1)),

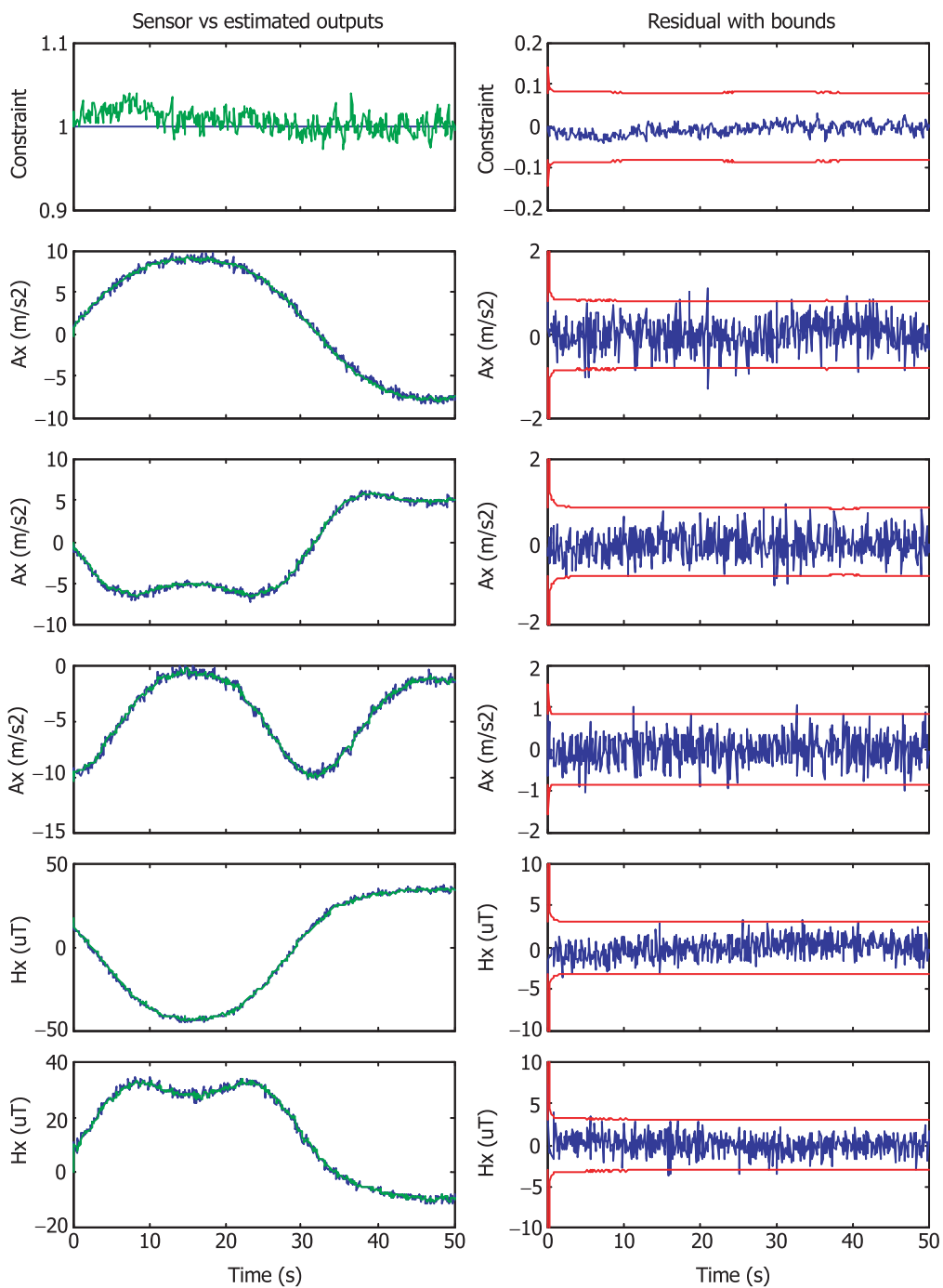
$$\begin{bmatrix} A_{x_1} \\ A_{y_1} \\ A_{z_1} \end{bmatrix} = \begin{bmatrix} A_{x_b} \\ A_{y_b} \\ A_{z_b} \end{bmatrix} - \begin{bmatrix} (q^2 + r^2) X_{Ax} \\ (p^2 + r^2) Y_{Ay} \\ (p^2 + q^2) Z_{Ay} \end{bmatrix}, \text{ (accounting for CG offset, (3)),}$$

$$\begin{bmatrix} A_{x_b} \\ A_{y_b} \\ A_{z_b} \end{bmatrix} = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_0q_3 + q_1q_2) & 2(q_1q_3 - q_0q_2) \\ 2(q_1q_2 - q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_0q_1 + q_2q_3) \\ 2(q_0q_2 + q_1q_3) & 2(q_2q_3 - q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \begin{bmatrix} A_{x_g} \\ A_{y_g} \\ A_{z_g} \end{bmatrix},$$

$$\begin{bmatrix} H_{x_b} \\ H_{y_b} \end{bmatrix} = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_0q_3 + q_1q_2) & 2(q_1q_3 - q_0q_2) \\ 2(q_1q_2 - q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_0q_1 + q_2q_3) \end{bmatrix} \begin{bmatrix} H_{x_g} \\ H_{y_g} \\ H_{z_g} \end{bmatrix}.$$

Even though the sensor models are derived accurately from the calibration data, additional states are provided in the filter model to account for error in sensor model, if any, as sensor bias. In the measurement model, the first observation is the quaternion constraint equation (Collinson 1998). It plays a very important role in containing the quaternion states always in the range of  $-1$  to  $+1$ , thus easing the scaling problems in the computation, and has been assigned highest priority by assigning low measurement covariance.

Figures 4a to 4c show the filter performance with the attitudes changing at a rate of  $0.1$  rad/s and filter update time of  $0.1$  s.



**Figure 4(a).** Comparison of filter outputs with sensor outputs and the resulting residual with bounds.

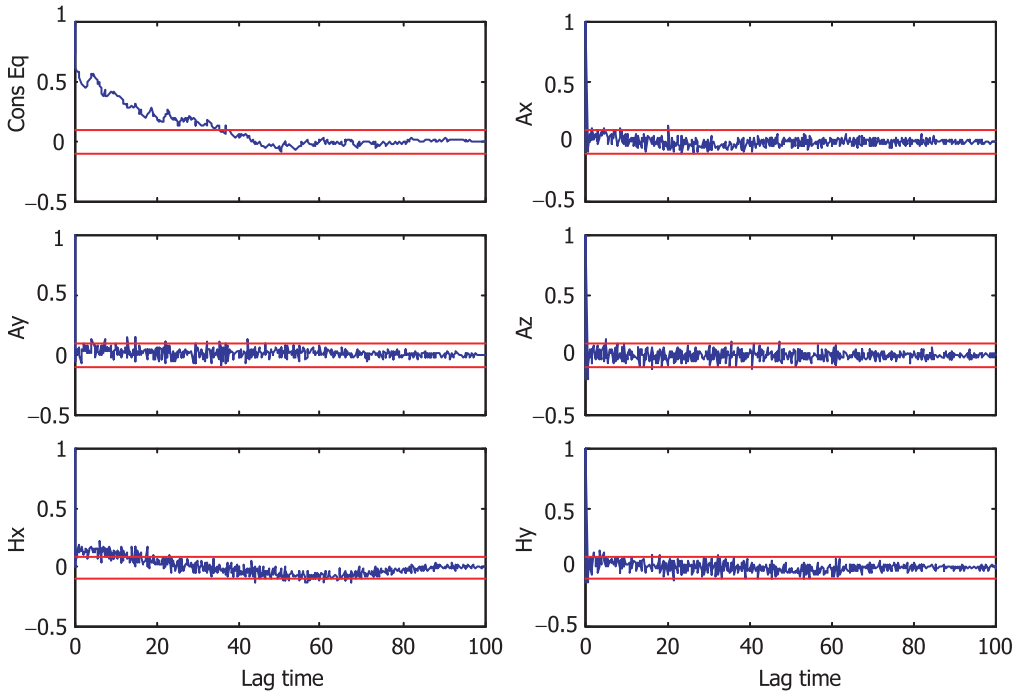


Figure 4(b). Autocorrelation of residual with bounds (whiteness test).

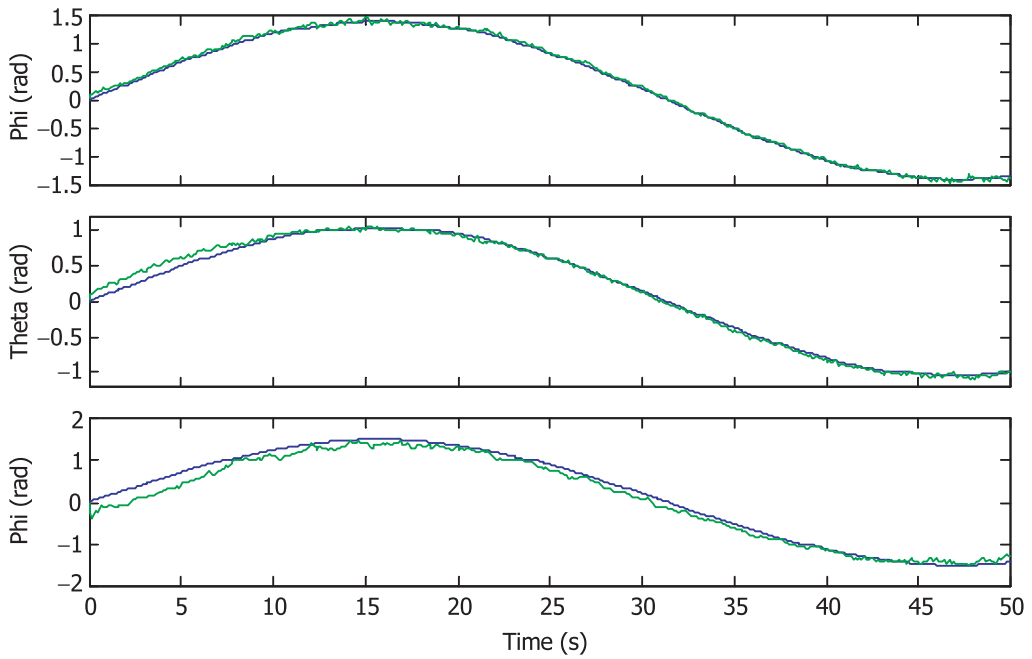


Figure 4(c). Comparison of estimated attitudes (computed from estimated quaternion states) with true attitudes.

### 6.2 *Effect of fast maneuvering and slower rate of filter update*

Though the plots shown in figures 4a to 4c confirm the reasonably good performance of the filter algorithm, when it is implemented on the micro-controller the filter performance is found to deteriorate. This is mainly because of the slower computational speed of the micro-controller (16 MHz clock speed). With the 12-state model each filter update cycle on the micro-controller takes nearly 2 seconds.

Further studies with the Matlab/Simulink setup show that, because the system model is nonlinear, the filter performance deteriorates when the filter update rate is slow. Figures 5a and 5b show the filter performance when the update time is increased from 0.1 s to 0.5 s. Therefore, this factor is critically considered while implementing and testing the filter algorithm on the micro-controller and efforts are made to reduce the filter update time.

### 6.3 *Kalman Filter performance with 4-state model*

To facilitate the faster filter update, the filter model size is reduced from 12-state to 4-state retaining only the quaternion states. This is done with the assumption that the sensor model is accurate enough and needs no sensor bias estimation.

Before implementing the filter with the 4-state model on the micro-controller, it is thoroughly validated in Matlab/Simulink. Figure 6a shows the Kalman Filter performance with the 4-state model when there is no sensor bias. Figure 6b presents the comparison of attitude estimates from the 12-state and 4-state models and shows almost similar performance. Figure 7a shows the Kalman Filter performance with the 4-state model when sensor bias is present. This figure shows that the residual and its autocorrelation cross their respective bounds, indicating that there is still some information in the measurements, which has not been extracted by the filter. However, the comparison of attitude estimates from 12-state and 4-state models, shown in figure 7b, is reasonably good.

These comparisons have been made to show the validity of the 4-state model, which though not as good as the 12-state model (only in the presence of bias in the sensor output), yet estimates attitudes reasonably well. Based on these results it was finally decided to implement the 4-state model on the micro-controller and model the sensors as accurately as possible, thus reducing the unaccounted biases in their outputs.

## 7. **Micro-controller implementation**

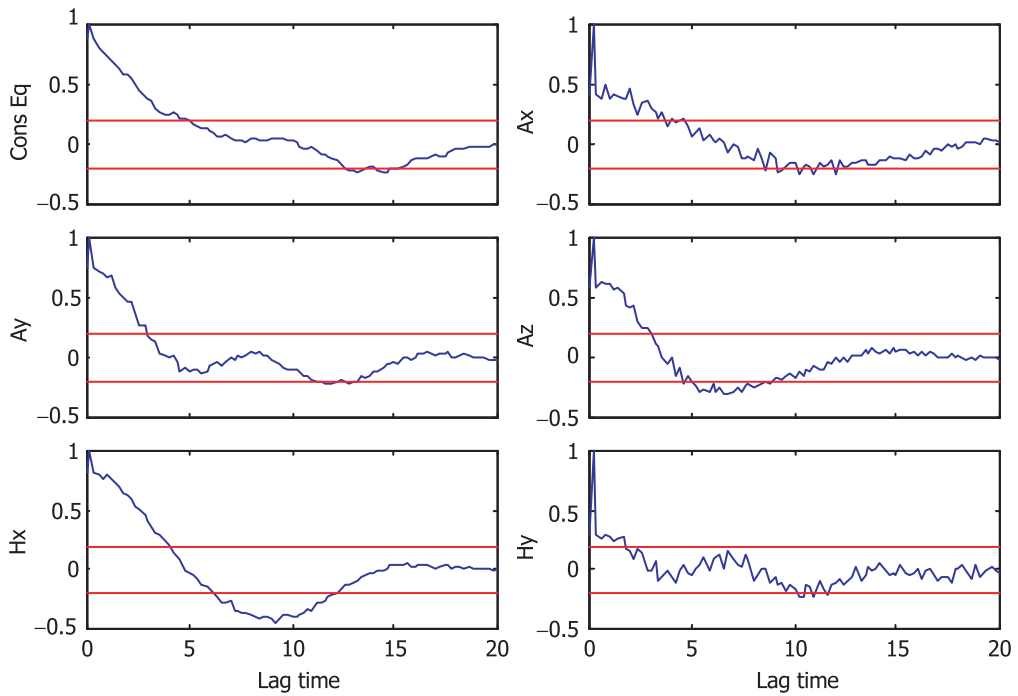
### 7.1 *Data acquisition*

7.1a *ADC outputs:* The outputs of rate gyros, accelerometers and temperature sensor are analog and are acquired through a 12-bit ADC. An assembly code has been developed to acquire the ADC output data.

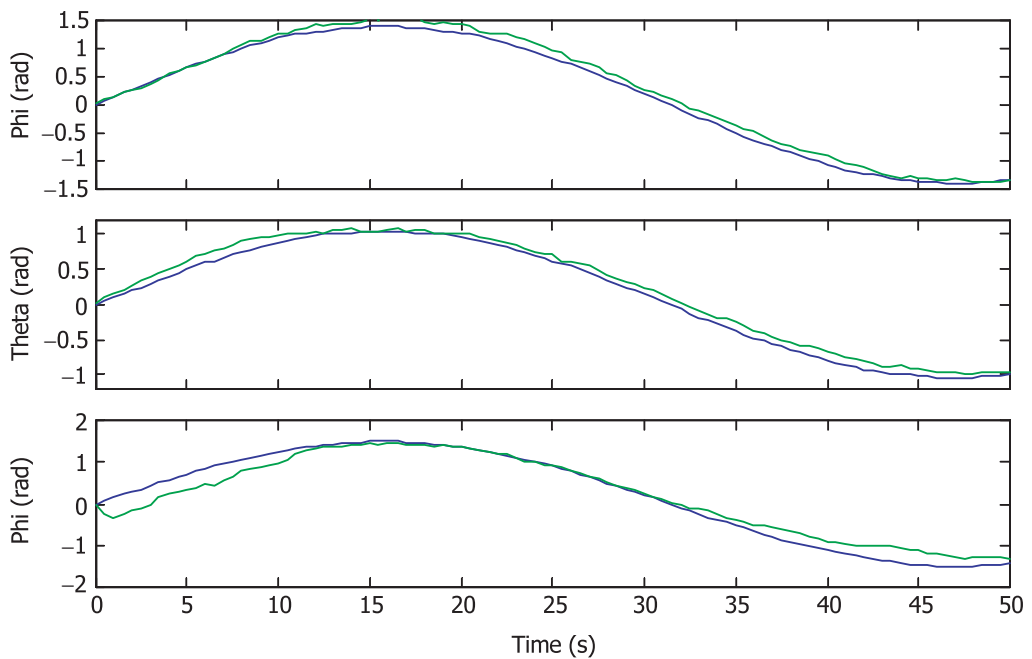
7.1b *Magnetometer outputs:* Magnetometer V2X, which is operational in the raw mode, gives digital output and hence is connected directly to the HC12 compact. Each time the magnetometer provides the reading after certain events take place at particular intervals of time. An interrupt routine is used to execute each event as and when it occurs, instead of waiting for the complete cycle to be over.

### 7.2 *Kalman Filter*

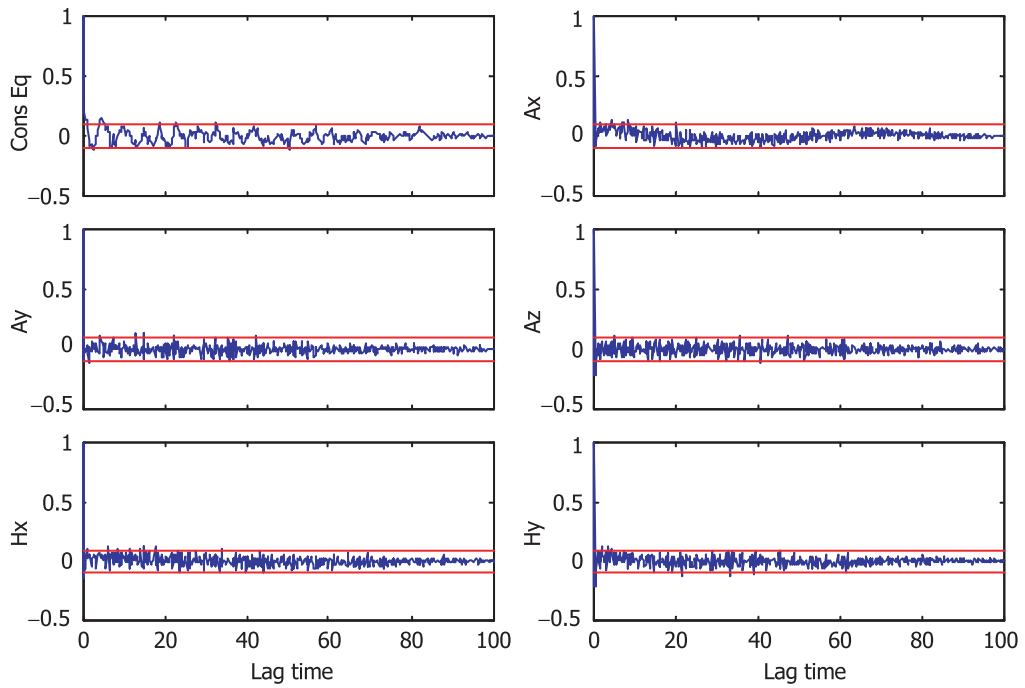
To overcome numerical divergence problem, the Kalman Filter algorithm is implemented in UD factorization form (Bierman 1977; Brown & Huang 1997). The UD factorization based



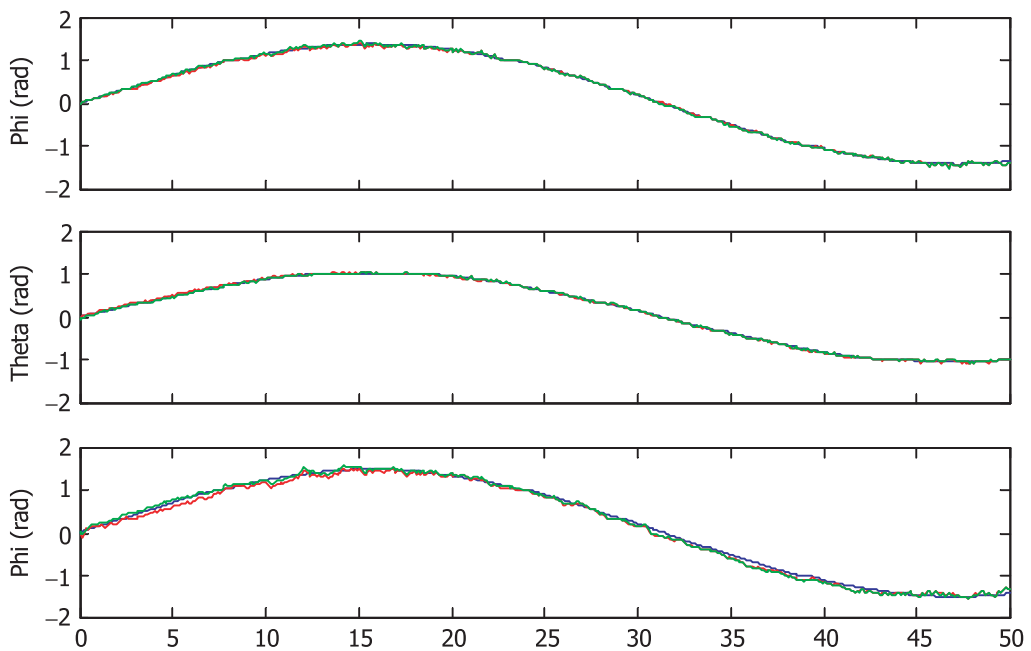
**Figure 5(a).** Autocorrelation of residual with bounds (whiteness test).



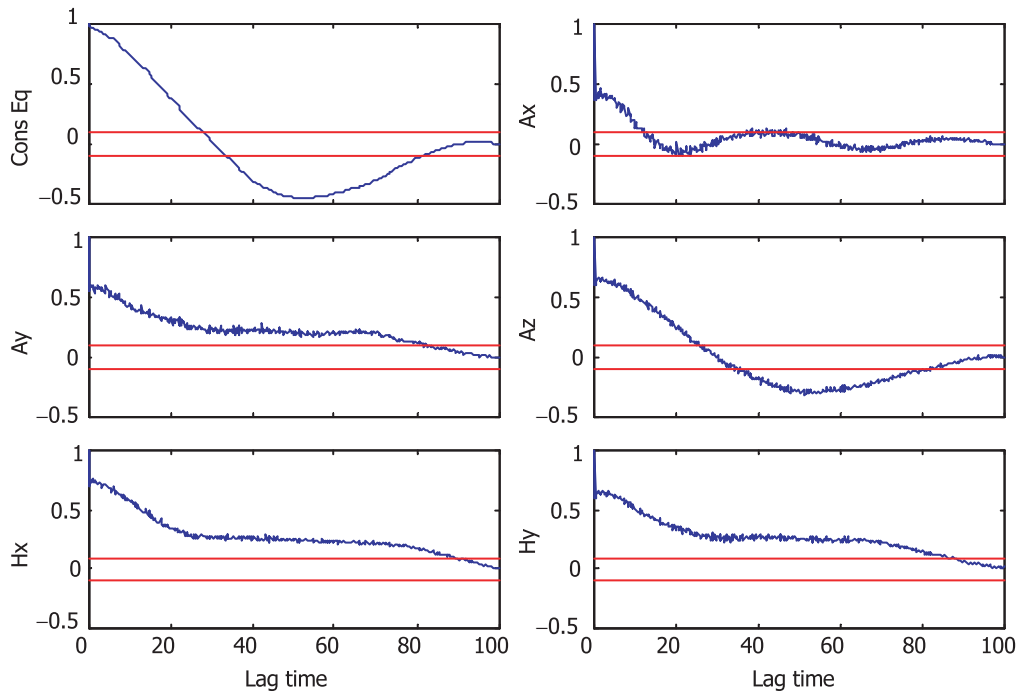
**Figure 5(b).** Comparison of estimated attitude with true attitude (when filter update time is 0.5 s).



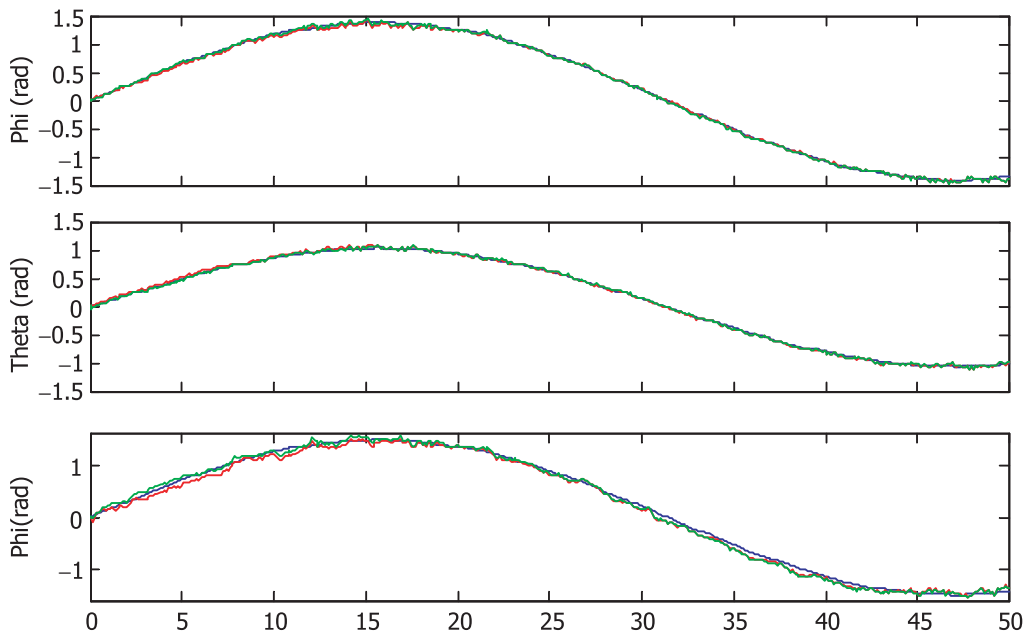
**Figure 6(a).** Autocorrelation of residual with bounds (whiteness test).



**Figure 6(b).** Comparison of attitude estimates from 4-state as well as 12-state models with true attitudes (when there is no bias in sensor outputs).



**Figure 7(a).** Autocorrelation of residual with bounds (whiteness test).



**Figure 7(b).** Comparison of attitude estimates from 4-state as well as 12-state models with true attitudes (with bias in sensors output).

Kalman Filter is coded in embedded C programming. The Kalman Filter code before implementing on the micro-controller has been compiled using PC compiler Borland C++, executed and tested to reproduce the results obtained from Matlab/Simulink setup. This exercise has been carried out to ensure that there is no error in filter code conversion from MATLAB to C. An interrupt routine is written also in C to acquire the magnetometer reading. The rate gyro outputs, accelerometer outputs and temperature transducer outputs are acquired via ADC by including the relevant assembly code. All the files are compiled and linked to a machine code using Image Craft C (ICC12) compiler (Image Craft 2001). The compiled machine language code is then downloaded onto the micro-controller using TwinPEEKs monitor program (HC12 Compact 1999) residing in the internal EEPROM of the HC12 Compact.

The ADC outputs thus acquired are of 12-bit resolution and converted into engineering units. Similarly the magnetometer readings are converted into microtesla. The rate gyro and temperature transducer outputs are then passed onto the Kalman Filter algorithm as inputs to the filter model, and accelerometer and magnetometer outputs as observations. The estimated quaternion states are then converted to Euler angles and displayed on the terminal by passing through the SCI (serial communication interface) port.

### 7.3 Testing, validation and observations

With the filter algorithm, the MIP is tested and filter performance is evaluated by simple experiments. While the algorithm is running, the MIP box is rotated and held at different orientations and the estimated attitudes displayed on the terminal are observed. From this simple experiment the following are observed.

- The estimates of roll and pitch attitude match the orientation of the MIP to an accuracy of approximately  $\pm 3^\circ$ .
- The heading angle, which is estimated mainly from the magnetometer output, though working shows large variation due to the effect of local magnetic field (hard iron effect). When the box is held at a distance from the ferromagnetic elements (local magnetic field), the estimates of heading angle are comparatively better.
- Both roll attitude and heading are tested for full  $360^\circ$  rotation. Though roll attitude estimation is reasonably accurate at all orientations, the heading estimation at  $0^\circ$  and  $180^\circ$  shows a jump of approximately  $\pm 10^\circ$
- The pitch attitude is found to work well between  $\pm 90^\circ$ .

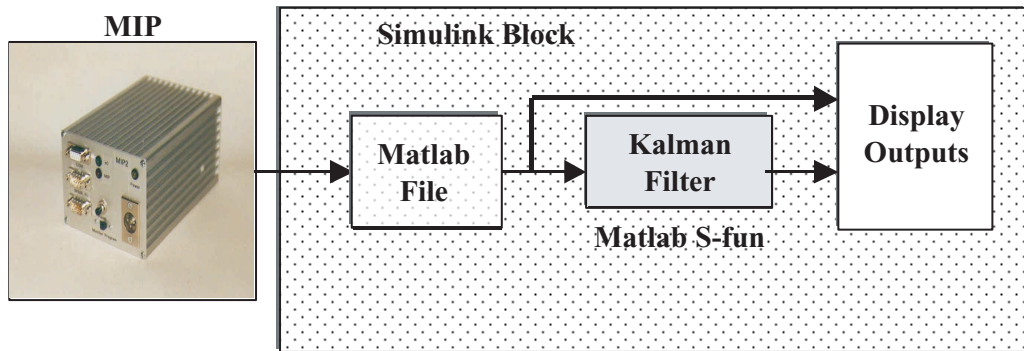
### 7.4 Real data analysis

The sensor outputs from MIP are collected over the serial communication interface into a file and used as sensor data for the Kalman Filter-based fusion algorithm in Matlab/Simulink set up (as shown in figure 8) for testing the filter performance with 4-state model in comparison with 12-state model. The estimates of attitudes from both the 12-state and 4-state models are comparable, particularly roll and pitch attitude as shown in figure 9. The heading, as expected is influenced by hard iron effect, which is compensated by estimating them as bias in 12-state model, but not in 4-state model and hence the difference.

## 8. Concluding remarks

In this work, the attitudes are derived from small, low-cost inertial and magnetometer sensors. An extended Kalman Filter-based estimation scheme is used to fuse the redundant information



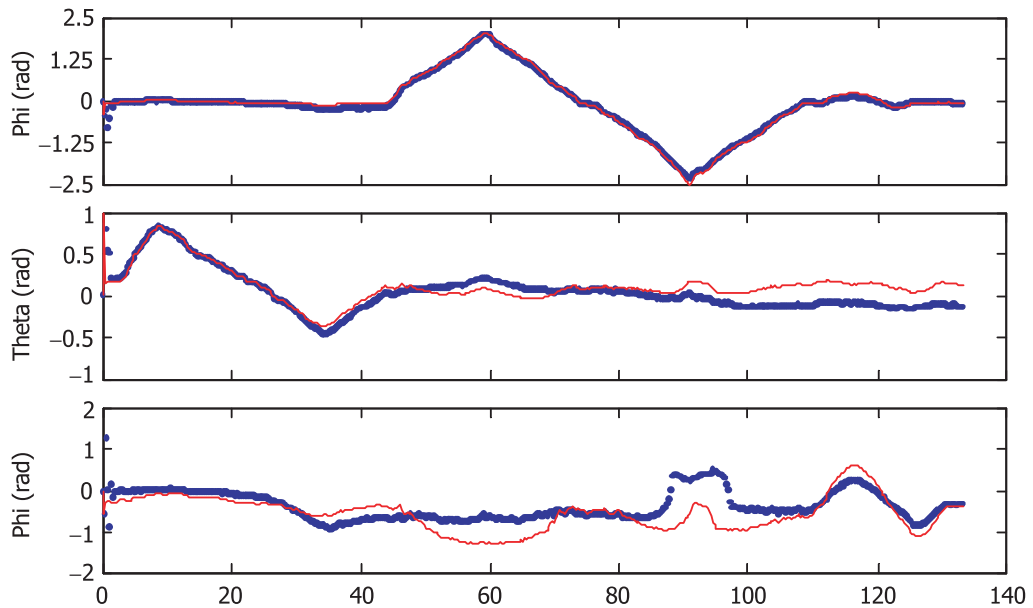


**Figure 8.** Simulink blocks for real data analysis.

coming from three orthogonally mounted accelerometers and a two-axis magnetometer. The accelerometers, together with the magnetometer, provide attitudes reference and the filter provides corrections to the attitude trajectory generated from the integration of the rate gyro outputs. The accuracy of the attitude estimation is improved by including the sensor models in the estimation algorithm.

The estimation of attitudes from low-cost MIP is first realized on a Matlab/Simulink platform and proves the capability of the Kalman Filter and multi sensor data fusion algorithm. Since the system model is nonlinear, the performance of the filter algorithm is found to suffer under certain conditions and the same has been highlighted.

Finally the Kalman Filter-based fusion algorithm is implemented on the hardware (MIP), by programming the micro-controller using ICC12 compiler. Owing to the computational



**Figure 9.** Comparison of attitude estimates from 12-state model (thick line) and 4-state (thin line) models from real data.

speed limitations of the micro-controller, no augmented states (sensor bias states) are included in the model. The decision to implement this minimum set of state (only 4 quaternion states) model is arrived at, based on the assumption that the sensor model used in the filter model is almost accurate and their outputs are bias-free. This 4-state model is thoroughly validated in Matlab/Simulink set up before implementing on the micro-controller. The algorithm is tested and evaluated by subjecting the MIP to pure angular motion, at different orientations. From this simple experiment, the accuracy of the estimated roll and pitch attitudes is found to be well within the stipulated accuracy level of  $\pm 5^\circ$  for the ALEX. However the estimation of heading angle, which is mainly derived from magnetometer output, seems to be influenced by variations in local magnetic field.

This work is carried out under the NAL-DLR exchange program.

### List of symbols

(general)

$A$	system state matrix;
$A_x, A_y, A_z$	linear acceleration along body axis;
$B$	system input matrix;
$f$	nonlinear state function;
$G_A$	transition matrix for the process noise;
$h$	nonlinear output function;
$H$	system output matrix;
$k$	as a subscript, it means at $k$ th instant;
$K$	Kalman gain;
$N$	number of samples;
$p, q, r$	angular rates about body axis;
$\dot{p}, \dot{q}, \dot{r}$	time derivative of respective angular rates;
$\tilde{P}$	time updated state error covariance matrix;
$\hat{P}$	measurement updated state error covariance matrix;
$Q$	process noise covariance;
$R$	measurement noise covariance;
$R_e$	covariance of innovation sequence;
$U_m$	input measurement vector;
$v$	measurement noise vector;
$w$	process noise vector;
$X$	state vector;
$\tilde{X}$	time updated state vector;
$\hat{X}$	measurement updated state vector;
$Z_m$	output measurement vector;
$\Phi$	state transition matrix.

(in mathematical model)

$A_{x_b}, A_{y_b}, A_{z_b}$	gravitational acceleration along body fixed frame;
$A_{x_g}, A_{y_g}, A_{z_g}$	gravitational acceleration along geodetic frame, $[0 \quad 0 \quad -9.81] \text{ m/s}^2$ ;

$A_{x_m}, A_{y_m}, A_{z_m}$	accelerometer sensors output;
$B_{A_x}, B_{A_y}, B_{A_z}$	bias in accelerometer sensors output;
$B_{H_x}, B_{H_y}$	bias in magnetometer sensors output;
$B_p, B_q, B_r$	bias in rate gyro sensors output;
$H_{x_b}, H_{y_b}$	earth's magnetic vector along body fixed frame.
$H_{x_g}, H_{y_g}, H_{z_g}$	earth's magnetic vector along geodetic frame, [18 0 45] $\mu$ tesla at Braunschweig, Germany.
$H_{x_m}, H_{y_m}$	magnetometer sensors output;
$p_b, q_b, r_b$	angular rates about the body fixed frame;
$p_m, q_m, r_m$	rate gyro sensors output;
$q_0, q_1, q_2, q_3$	quaternion states;
$T_m$	temperature sensor output;
$X_{A_x}, Y_{A_y}, Z_{A_z}$	distance of $A_x, A_y, A_z$ sensors from CG, [-1.0 2.0 2.0] cm.

### Abbreviations

ALEX	autonomous parafoil landing experiment
DLR	German Aerospace Research Centre
EKF	extended Kalman Filter
GNC	guidance, navigation and control
GPS	global positioning system
ICC12	ImageCrafts C Compiler for Motorola HC12 Compact
IMU	inertial measurement unit
INS	inertial navigation system
MEMS	micro electro mechanical systems
MIP	miniaturized inertial platform
UAV	unmanned aerial vehicle

### References

- Anderson B D, Moore J B 1979 *Optimal filtering* (Englewood Cliffs, NJ: Prentice Hall)
- Bierman G J 1977 *Factorization methods for discrete sequential estimation* (New York: Academic Press)
- Brown R G, Hwang P Y C 1997 *Introduction to random signals and applied Kalman Filtering* (New York: John Wiley and Sons)
- Candy J V 1987 *Signal processing: The model based approach* (Singapore: McGraw Hill)
- Collinson R P G 1998 *Introduction to avionics* (London: Chapman and Hall)
- Doherr K F, Jann T 1997 Test vehicle ALEX for low-cost autonomous parafoil landing experiments. *14th AIAA Aerodynamic Decelerator Systems Technology Conference and Seminar*, AIAA-97-1543
- HC12 Compact 1999 HC12 Compact hardware manual, Version 1.0
- Image Craft C 2001 Compiler and Development Environment for Motorola HC12 Manual, Version 6
- Jann T 2001 Aerodynamic model identification and GNC design for the parafoil-Load System ALEX. *Proc. of AIAA Aerodynamic Systems Decelerator Conference*, AIAA-2001-2015
- Jann T, Doherr K F, Gockel W 1999 Parafoil test vehicle ALEX – Further development and flight test results, AIAA-99-1751
- Schultz D G, Melsa J L 1967 *State functions and linear control systems* (McGraw Hill)
- Valvano J W 2000 *Embedded microcomputer systems* (London: Brooks/Cole)