UNSYS digital

Journal of Unmanned System Technology

Position Estimation for Autonomous Hover of a Mini-Helicopter

Vinodhini C.[†], Puneet Singh[§], and C. Venkatesan[§]

[†]Department of Mechanical Engineering, Indian Institute of Technology, Kharagpur, India

[§]Department of Aerospace Engineering, Indian Institute of Technology, Kanpur, India

***Abstract*—**The state estimation of Unmanned Aerial Vehicles (UAVs) is crucial to control their orientation and navigation. The ability of an unmanned autonomous helicopter to hover enables one to operate in areas inaccessible or hazardous to other vehicles. In order to achieve stable hovering at a particular attitude and height, accurate estimations of orientation and position are essential. This paper presents the research activity taken up at IIT Kanpur on the position estimation of an autonomous mini-helicopter. A study was conducted on the calibration of the onboard Inertial Measurement Unit (IMU) and its data was used to estimate position. The next step is aimed at integrating Global Positioning System (GPS) with the IMU to obtain more accurate/reliable data by implementing Kalman filtering.

***Keywords*—**Autonomous mini-helicopter, position estimation, IMU, GPS, Kalman filter.

NOMENCLATURE

UAV	=	Unmanned Air Vehicle
IMU	=	Inertial Measurement Unit
GPS	=	Global Positioning System
XYZ	=	ground-fixed reference frame
$\vec{a}_H$	=	linear acceleration of helicopter with respect to the ground
$\vec{a}_{IMU}$	=	linear acceleration of IMU with respect to the ground
$a_{H,X}$	=	X component of acceleration of the helicopter
$a_{H,Y}$	=	Y component of acceleration of the helicopter
$a_{H,Z}$	=	Z component of acceleration of the helicopter
$a_{IMU,X}$	=	X component of acceleration of the IMU
$a_{IMU,Y}$	=	Y component of acceleration of the IMU
$a_{IMU,Z}$	=	Z component of acceleration of the IMU
$\vec{\omega}$	=	angular velocity/rotation rate of IMU
$\dot{\vec{\omega}}$; $\vec{\alpha}$	=	angular acceleration of IMU
p	=	angular velocity of the IMU/helicopter about X axis
q	=	angular velocity of the IMU/helicopter about Y axis
r	=	angular velocity of the IMU/helicopter about Z axis
$\vec{d}$	=	relative displacement between the IMU and CG of the helicopter with respect to the ground
d_X	=	X component of relative displacement
d_Z	=	Z component of relative displacement
dt	=	time step
X	=	displacement of the IMU/helicopter along X direction
Y	=	displacement of the IMU/helicopter along Y direction
Z	=	displacement of the IMU/helicopter along Z direction
$\vec{s_i}$	=	initial position of the helicopter with respect to the ground
$\vec{s_f}$	=	final position of the helicopter with respect to the ground
φ	=	roll angle of the IMU/helicopter
θ	=	pitch angle of the IMU/helicopter
ψ	=	roll angle of the IMU/helicopter
Δt	=	discrete time step
g	=	acceleration due to Earth's gravity (9.8 m/s^2)
$\boldsymbol{x}$	=	vector of the states of the system
$\boldsymbol{F}$	=	system dynamics matrix
$\boldsymbol{w}$	=	white noise process vector
$\boldsymbol{Q}$	=	process-noise matrix
$\boldsymbol{z}$	=	measurement vector
$\boldsymbol{H}$	=	measurement matrix
$\boldsymbol{v}$	=	white noise measurement vector
Φ	=	fundamental or transition matrix
$\boldsymbol{P}$	=	covariance matrix after an update
$\boldsymbol{M}$	=	covariance matrix before an update
$\boldsymbol{K}$	=	Kalman gain matrix
$\hat{\boldsymbol{x}}_k$	=	current state estimate
Φ_s	=	noise spectral density
$\hat{\boldsymbol{x}}_k^-$	=	a priori estimate of current state
$\boldsymbol{e}_k^-$	=	a priori error in the current estimate
$\boldsymbol{e}_k$	=	a posteriori error in the current estimate

I. INTRODUCTION

DEVELOPMENT of Unmanned Aerial Vehicles (UAVs) is a promising research area due to their advanced capabilities and great flexibility. A. M. Low's "Aerial Target" of 1916 was the first known attempt at a powered UAV [1]. Autonomous mini helicopters are one such classification of UAVs. The ability of an unmanned autonomous helicopter to

Corresponding author: Vinodhini C. (e-mail: vinodhini.830@gmail.com).
This paper was submitted on December 01, 2013; revised on January 19, 2014; and accepted on February 24, 2014.

vertically take-off and land as well as hover enables one to operate in areas inaccessible by other vehicles. Furthermore, it can perform tasks which would be exceedingly difficult or hazardous for a manned vehicle. There are several possible applications for this technology, including terrain surveying, close-up inspection of power lines, military operations, monitoring traffic, search-and-rescue missions, filming movies, and the investigation and clean-up of hazardous waste sites.

However, one of the major difficulties of an autonomous helicopter is to keep it hovering at a particular position and attitude in a stable manner. Hence, the position estimation of the vehicle is highly essential. Unfortunately, most of the available positioning technologies have limitations either in accuracy of the absolute position (eg. GPS with Selective Availability), accumulated error (eg. dead-reckoning systems such as odometry), or availability (eg. GPS with Differential Correction) [2]. A plausible solution to determining the vehicle's position in some convenient coordinate system is by using an Inertial Measurement Unit (IMU).

In this study, experiments were carried out using an IMU and its data was analyzed. Suitable calibrations and corrections in the output were made and the device limitations were realized. The possible error contributions and their corrections were investigated. Subsequently, the concept of Kalman filtering and its applications were studied. Assuming two models in Kalman filtering and using test flight data (GPS and IMU), velocity and position estimations were analyzed.

The paper is organized as follows. The basic equations for obtaining the helicopter position, which involve the IMU parameters, are described in Section II. The calibration of the IMU and its associated experiment, including observations, are presented in Section III. The results and graphs of the IMU experiment are included in Section IV. Kalman filtering fundamentals and the models considered for state estimation are mentioned in Section V. The Kalman filtering simulation results are presented in Section VI. Finally, the concluding remarks end the paper.

II. State Estimation using IMU Data

An IMU is an electronic device containing sensors and it reports the following parameters of a craft –

1. Linear accelerations along X, Y, Z using accelerometers (by measuring specific forces)
2. Angular rates about X, Y, Z using gyroscopes
3. Orientation i.e., roll φ, pitch θ, yaw ψ, using magnetometers (by measuring the magnetic field to determine the magnetic heading)

From the linear acceleration and rotation rate data provided by the IMU, the acceleration of the helicopter with respect to the ground can be written as

$$\vec{a}_H = \vec{a}_{IMU} - \{(\dot{\vec{\omega}} \times \vec{d}) + \vec{\omega} \times (\vec{\omega} \times \vec{d})\} \quad (1)$$

where

$$\vec{a}_H = a_{H,X}\hat{\imath} + a_{H,Y}\hat{\jmath} + a_{H,Z}\hat{k} \quad (2)$$

$$\vec{a}_{IMU} = a_{IMU,X}\hat{\imath} + a_{IMU,Y}\hat{\jmath} + a_{IMU,Z}\hat{k} \quad (3)$$

$$\vec{\omega} = p\hat{\imath} + q\hat{\jmath} + r\hat{k} \quad (4)$$

$$\vec{d} = d_X\hat{\imath} + d_Z\hat{k} \quad (5)$$

The above equations are written with respect to the ground reference frame XYZ (after transformation of IMU data from body reference frame to ground-fixed reference frame). Since the IMU is placed close to the nose and along the central line of the helicopter, the relative displacement between the CG of the helicopter and the IMU, has components only along X and Z directions as indicated by (5).

Using the linear acceleration values of the helicopter, the displacement of the vehicle along X, Y, Z ground axes (fixed reference frame) can be determined as follows.

$$\iint_0^{\Delta t} a_{H,X}dt = X$$
$$\iint_0^{\Delta t} a_{H,Y}dt = Y$$
$$\iint_0^{\Delta t} a_{H,Z}dt = Z \quad (6)$$

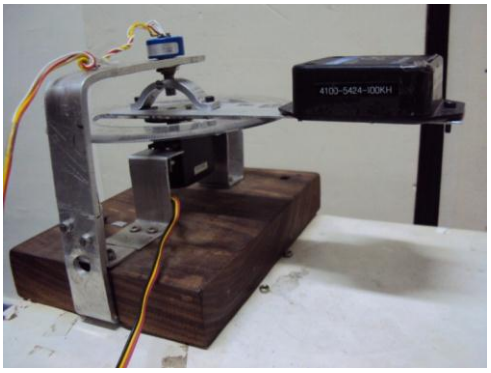
With $\vec{s_i}$ known (which is the lift-off position for the helicopter), the helicopter's position $\vec{s_f}$ at any instant in ground reference frame can be obtained in the following manner –

$$\vec{s_f} = \vec{s_i} + (X\hat{\imath} + Y\hat{\jmath} + Z\hat{k}) \quad (7)$$

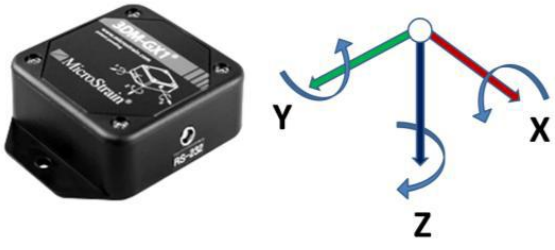
III. IMU Calibration and Experiment

The following section elaborates the calibration of the IMU 3DM-GX1 and the experiment carried out for analyzing the velocity and displacement outputs.

A. Setup and Components Used



(a)



(b)

Figure 1 (a) Experimental setup, (b) MicroStrain 3DM-GX1 IMU [6]

1. IMU: MicroStrain 3DM-GX1
2. Servo motor: Futaba S3151
3. Potentiometer: Rotary potentiometer
4. Power Supply and DAQ: NI PXI-1050, NI PXI-6289

The experimental setup was tested multiple times and calibrated before the actual experiment was carried out to ensure that the error contribution from the test setup is minimal and insignificant. The mechanical misalignment in the setup, the servo motor errors and the potentiometer errors were checked and suitably rectified.

B. Device Calibration

IMU: The IMU was calibrated to show zero angular rates along X, Y, Z axes when held stationary and acceleration due to gravity along the vertical axis as 9.8 m/s^2, through a LabVIEW program. The sensor misalignment errors were also taken into account during calibration.

Potentiometer: In a rotary potentiometer, $V = f(\theta)$, where θ is the angle of rotation of the potentiometer wiper and V is the voltage output (potential difference between the wiper and ground). In the experimental setup indicated by **Figure 1**, the trim screw of the potentiometer, the armature of the servo motor and the bar (to which the IMU is attached) rotate about the same axis. Thus, θ can be considered as the angle of rotation of the arm. The exact relation between θ and v (voltage ratio) is determined by

$$v = \frac{V}{Maximum\ voltage\ output} \tag{8}$$

Using a LabVIEW program, the duty cycle supplied to the servo motor was increased in steps of 0.25% over its entire range and the angle at every position was noted from the protractor and recorded. Simultaneously, the voltage ratio was calculated at every position. The subsequent plots were obtained.

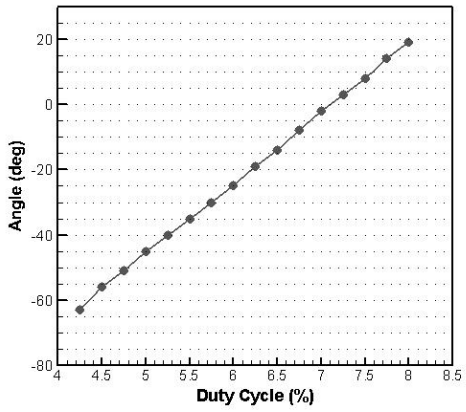


Figure 2 Plot of angle vs duty cycle

The exact relation between θ and v is modeled as a linear variation given by $\theta = mv + c$, where m is the slope and c is the y-axis intercept. The values of m and c were found to be 336.526 (deg) and -134.435 (deg) respectively. This θ is taken as the reference for angle of rotation of the IMU and its derivatives for angular velocity and angular acceleration respectively.

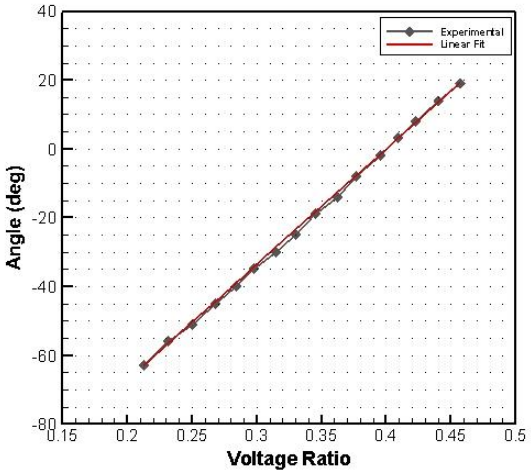


Figure 3 Plot of angle vs voltage ratio

C. Experiment

The following steps were executed:

1. A pulse-like input was given to the servo motor (to and fro motion), with duty cycle varying over a range within 2-11% (minimum and maximum limits possible). The upper limit in each case was achieved in 30 steps starting from the lower limit and vice-versa (refer **Figure 4**).
2. The arm connected to the servo armature rotated to and fro (once), thereby causing the IMU, which is attached to it in a particular orientation, to also rotate.
3. Data was recorded from the potentiometer and IMU. The data included time, voltage ratio, accelerations along the 3 axes, angular velocities about the 3 axes, and the roll, pitch, yaw angles.
4. This procedure was repeated with different IMU orientations as well as duty cycle ranges.

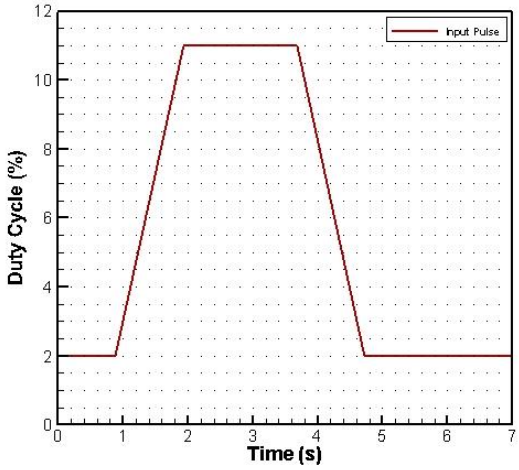


Figure 4 Plot of duty cycle input to servo motor vs time

Figure 4 is an example of one of the inputs given to the servo motor in terms of the duty cycle variation with time. In this case, after approximately 1 second, the duty cycle changed progressively from 2% (initial) to 11% (final) and after a brief pause, returned to 2% and continued to remain in that position till the end.

Plots of the following data were made and compared (for each duty cycle range and orientation).

1. Angle of rotation from:
   i. Potentiometer data [$\theta_1 = mv + c$]
   ii. Euler angle from IMU [θ_2]
   iii. Angular rate data from IMU [$\theta_3 = \theta_3' + r\Delta t$]
   iv. Tangential acceleration data from IMU [$\theta_4 = \theta_4' + \Omega'\Delta t + (a_t/l)\Delta t^2/2$]
2. Angular velocity from:
   i. Potentiometer data [$\omega_1 = \Delta\theta_1/\Delta t$]
   ii. Euler angle data from IMU [$\omega_2 = \Delta\theta_2/\Delta t$]
   iii. Angular rate from IMU [ω_3]
   iv. Tangential acceleration data from IMU [$\omega_4 = \omega_4' + (a_t/l)\Delta t$]
   v. Radial acceleration data from IMU [$\omega_5 = \sqrt{a_r/l}$]
3. Angular acceleration from:
   i. Potentiometer data [$\alpha_1 = \Delta\omega_1/\Delta t$]
   ii. Euler angle data from IMU [$\alpha_2 = \Delta\omega_2/\Delta t$]
   iii. Angular rate data from IMU [$\alpha_3 = \Delta\omega_3/\Delta t$]
   iv. Tangential acceleration data from IMU [$\alpha_4 = a_t/l$]

Δt is the time step between two readings and l is the length of the arm on which the IMU was fixed. The angle of rotation, angular velocity and angular acceleration calculated above using different data correspond to the parameters of one particular orientation during any of the experimental runs i.e., either roll, pitch, or yaw motion with the remaining two angles kept constant.

Various combinations of orientations and duty cycle ranges were tested. The angle of orientation of the IMU ranged from 0^o to 90^o about all three axes while the duty cycle ranged from 2% to 11%. It was found that 1% change in duty cycle of the servo motor corresponds to approximately 20° rotation of the IMU.

D. Observations and Corrections Implemented

1. Plots of the data from potentiometer and Euler Angles (IMU) consistently matched in all the graphs.
2. Data from the rate gyroscope (IMU) were ½ of the expected values. This error was rectified by changing the values of GyroGainScale and AccelGainScale to 8500 and 10000 respectively in the LabVIEW program for IMU Data Acquisition. These GainScales are constants (calibration factors) for each of the sensors i.e., Rate Gyroscopes and Accelerometers. It is known that the rotation rate of the Earth about its axis is 7.292×10^{-5} rad/s. Since this value is extremely small compared to the angular velocities obtained in the experiment using the IMU gyroscopes, the Earth's rotation rate has been ignored.
3. Data from the accelerometers (IMU) were inconsistent in all the graphs. The inconsistencies were reduced by changing the integration method to trapezoidal from rectangular. Euler Angle errors were duly noted and the appropriate corrections were made to the accelerometer data.
4. Disturbances/deviations in the graphs were caused by noise in the data and experimental vibrations. The noise in the data was corrected by implementing 10 point moving average while the experimental vibrations were lowered by changing the arm on which the IMU was attached from aluminum to composite carbon fiber material.
5. Similar results were observed for other IMU orientations and duty cycle ranges.
6. Angle restrictions of $-90° < \varphi, \theta < +90°$ had to be followed. There were no such restrictions in ψ.

IV. RESULTS

A. Graphs of IMU Parameters vs Time

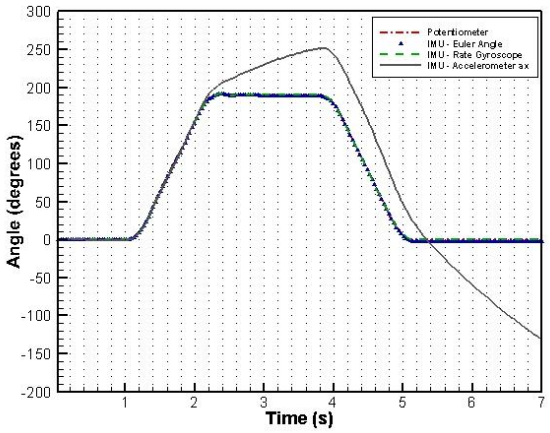


Figure 5 Yaw angle vs time for 2-11% duty cycle

In **Figure 5**, three plots match quite accurately while the fourth plot of accelerometer data is deviating. The data is consistent during the rise but subsequently, the deviation is large. This is possibly because as angular acceleration is integrated to obtain angle, the accumulated errors are also integrated leading to further deviation from original plot. 2-11% duty cycle change corresponds to approximately 180°, which is the largest possible range of motion in the experiment. Despite the deviation, **Figure 5** is the best plot among all the ranges of angular motion. This is mostly because the large angle change or slower motion rate allows the sensors to detect changes in motion more accurately.

On observing **Figure 6**, it can be concluded that three plots match perfectly, namely data from the Potentiometer, IMU Euler Angle and IMU rate gyroscope. The remaining two are accelerometer data plots. The data plot of accelerometer a_y (magenta) follows the expected trend in the peak regions while that of a_x (dark grey) approximately replicates the first peak only. It can also be noted that after 2 s, the grey plot continues to deviate with a brief match in the second peak rise. This deviation again can be attributed to accumulated error, after integrating from acceleration. **Figure 6** is relatively the best plot among all the angle ranges for reasons being similar to those mentioned with **Figure 5**.

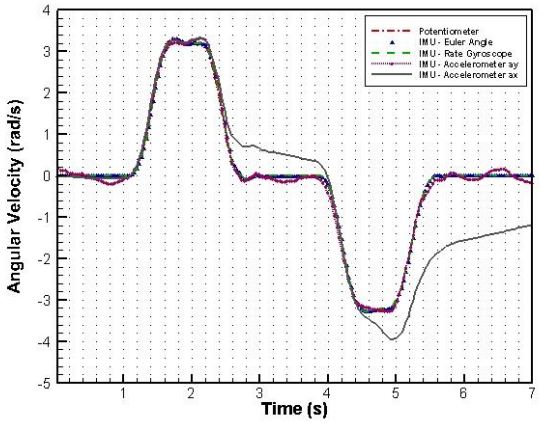


Figure 6 Angular velocity (ω_z) vs time for 2-11% duty cycle

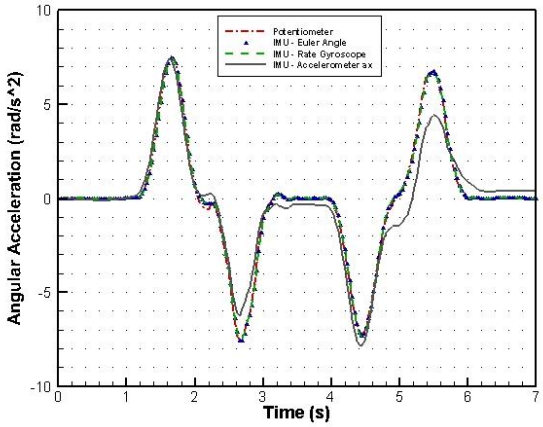


Figure 7 Angular acceleration (α_z) vs time for 2-11% duty cycle

Figure 7 depicts the best accelerometer data graph, with the exception of one plot being mismatched. It was observed that in most of the graphs where accelerometer data was quite consistent with other plots, one of the peaks never synchronized. The peak value was slightly higher or lower and it occurred at any one of the four peaks. The exact reason for this anomaly is still unknown. It is possible that at one peak, the accelerometer did not respond to the change as expected but once a change in motion occurred, it was able to pick up data more accurately.

B. Bias Estimation of Accelerometer a_z

Consider the acceleration along Z axis of the IMU a_z, as indicated by the accelerometer of the IMU. If we consider Roll motion of the IMU i.e., the IMU to be rotated by an angle φ about its X axis and then kept stationary, at this instant, the acceleration along Z axis can be modeled as

$$a_z = g\cos\varphi + \{B + SF(g\cos\varphi) + K(g\cos\varphi)^2\} \quad (9)$$

where g is the acceleration due to gravity (9.81 m/s^2), B is the Bias error, SF is the Scale Factor and K is g-squared sensitive Drift [3 , pp. 82-83].

Theoretically, it is expected that the accelerometer output will only contain the first term i.e., $g\cos\varphi$ (vertical angle dependent gravity term). In reality, the term within the curly brackets is also added, which is the Error term. This error term is a second order polynomial which is a function of $g\cos\varphi$.

Bias error is an offset that remains constant in all measured data which, in this case, is acceleration. Scale Factor is proportionality constant between the measured and actual acceleration, which appears as the coefficient of the first order term in the error. g-squared sensitive Drift is the coefficient of the second order error term. 'g-squared' implies that the error term depends on either the square of a single acceleration, or the product of two orthogonal accelerations [4].

In this section, the attempt made to estimate the above errors for the Z axis accelerometer (by only changing roll angle) has been described. The following steps were performed –

1. The IMU was mounted on a test rig which allowed roll, pitch as well as yaw motions. For this experiment, the orientation of the IMU was changed only about its X axis (roll motion), while the pitch and yaw motions were arrested.
2. The IMU was rotated from -40° to +40°, with data recorded at every 5°.
3. Using the data recorded, a graph of a_z vs $g\cos\varphi$ as well as the general fit polynomial curve (2nd order) for the same were plotted.
4. The coefficients of the polynomial function were noted.
5. This experiment was performed 5 times and the average errors were calculated.

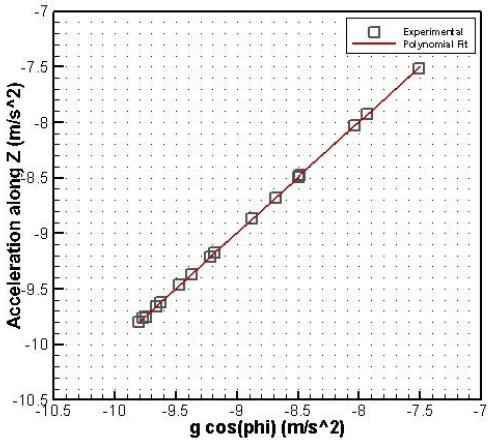


Figure 8 Graph of measured Z axis acceleration vs actual Z axis acceleration

Figure 8 is the result for one set of data. The polynomial coefficients were obtained where 0th order signified B, 1st order signified 1 + SF and 2nd order signified K.

The average error values were calculated and are listed below.

Bias error B = 0.0513538 m/s^2

Scale Factor SF = 0.008326

g-squared sensitive Drift K = 0.0004034 (m/s^2)$^{-1}$

It can be noted that these values are very small, with major contribution from Bias, followed by Scale Factor and lastly, g-dependent Drift.

V. KALMAN FILTERING

The Kalman Filter is an algorithm that uses the measurements (from sensors), which contain noise and other errors, and produces estimates of known as well as unknown

parameters which are more precise than the measured quantities. It is the most important algorithm for state estimation and it optimally combines the incoming measurement data with the predicted filter state. It makes the following assumptions –

1. The model (of the real world) is linear.
2. The system for which the states are to be estimated is excited by process noise.
3. The measurements contain random measurement noise.

It has a wide range of applications including guidance, navigation and control of vehicles like aircrafts, spacecraft and ships, etc.

The working algorithm can be described in 2 steps –

1. **Prediction step**: The estimates of the current state variables, including their uncertainties, are produced.
2. **Correction step**: The estimates are updated when a measurement is available using a weighted average concept.

In order to apply the Kalman filtering theory, our model of the real world is described by a set of linear differential equations which when represented in matrix or state-space form is given by

$$\dot{\boldsymbol{x}} = \boldsymbol{F}\boldsymbol{x} + \boldsymbol{G}\boldsymbol{u} + \boldsymbol{w} \tag{10}$$

where $\boldsymbol{x}$ is a column vector with the states of the system, $\boldsymbol{F}$ is the system dynamics matrix, $\boldsymbol{u}$ is a known or control vector and $\boldsymbol{w}$ is a white-noise process vector [3 , pp. 129]. There is a process-noise matrix $\boldsymbol{Q}$ that is related to the process-noise vector according to

$$\boldsymbol{Q} = E[\boldsymbol{w}\boldsymbol{w}^T] \tag{11}$$

Process noise is an indicator to inform the filter that it is known to us that our filter's model of the real world is not precise. The measurements are linearly related to the states according to

$$\boldsymbol{z} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{v} \tag{12}$$

where $\boldsymbol{z}$ is the measurement vector, $\boldsymbol{H}$ is the measurement matrix that relates the state to the measurement and $\boldsymbol{v}$ is the white noise measurement vector [3 , pp. 130]. The measurement noise matrix $\boldsymbol{R}$ is related to the measurement noise vector according to

$$\boldsymbol{R} = E[\boldsymbol{v}\boldsymbol{v}^T] \tag{13}$$

The fundamental/transition matrix for a time-invariant system can be found from the system dynamics matrix using the relation

$$\Phi(t) = e^{\boldsymbol{F}t} = \boldsymbol{I} + \boldsymbol{F}t + \frac{(\boldsymbol{F}t)^2}{2!} + \ldots + \frac{(\boldsymbol{F}t)^n}{n!} + \ldots \tag{14}$$

A. Discrete Polynomial Kalman Filter Without Control Vector

This filter is applied for a polynomial measurement signal corrupted with noise. The preceding relationships must be discretised to build a discrete Kalman filter. Terms involving the control vector are eliminated. Our simulations use this particular Kalman filter.

If the measurement is taken every T_s seconds (sampling time), the discrete fundamental matrix is given by

$$\Phi_k = \Phi(T_s) \tag{15}$$

Φ_k matrix was obtained by considering the first three-four terms of the Taylor series expansion of Φ (14).

The discrete form of the Kalman filtering measurement equation becomes

$$\boldsymbol{z}_k = \boldsymbol{H}\boldsymbol{x}_k + \boldsymbol{v}_k \tag{16}$$

and

$$\boldsymbol{R}_k = E\{\boldsymbol{v}_k \boldsymbol{v}_k{}^T\} \tag{17}$$

where $\boldsymbol{R}_k$ is a matrix consisting of the variances of each of the measurement noise sources [3 , pp. 130]. The resultant Kalman filtering equation is given by

$$\hat{\boldsymbol{x}}_k = \Phi_k \hat{\boldsymbol{x}}_{k-1} + \boldsymbol{K}_k(\boldsymbol{z}_k - \boldsymbol{H}\Phi_k \hat{\boldsymbol{x}}_{k-1}) \tag{18}$$

where $\boldsymbol{K}_k$ represents the Kalman gain matrix and $\hat{\boldsymbol{x}}_k$ represents the current state estimate [3 , pp. 130]. The Kalman gains are computed (while the filter is operating) from the matrix Riccati equations, which are a set of recursive matrix equations given by

$$\boldsymbol{M}_k = \Phi_k \boldsymbol{P}_{k-1} \Phi_k{}^T + \boldsymbol{Q}_k \tag{19}$$

$$\boldsymbol{K}_k = \boldsymbol{M}_k \boldsymbol{H}^T (\boldsymbol{H}\boldsymbol{M}_k \boldsymbol{H}^T + \boldsymbol{R}_k)^{-1} \tag{20}$$

$$\boldsymbol{P}_k = (\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{H})\boldsymbol{M}_k \tag{21}$$

where $\boldsymbol{P}_k$ is a covariance matrix representing errors in the state estimates (i.e., variance of truth minus estimate) after an update, $\boldsymbol{M}_k$ is a covariance matrix representing errors in the state estimates before an update and $\boldsymbol{I}$ is the identity matrix [3 , pp. 131]. The discrete process-noise matrix $\boldsymbol{Q}_k$ can be found as follows

$$\boldsymbol{Q}_k = \int_0^{T_s} \Phi(\tau) \boldsymbol{Q} \Phi^T(\tau)\, d\tau \tag{22}$$

To start the Riccati equations, an initial covariance matrix $\boldsymbol{P}_0$ is required, which is usually initialized to very high values.

The simulations were carried out assuming 2 models, namely constant acceleration model and constant jerk model. The following sections depict the parameters assumed and calculated, which were instrumental in obtaining the Kalman filter estimations of position, velocity and acceleration of the helicopter.

B. Derivation of Matrix Riccati Equations

Considering that there is no deterministic disturbance or control vector, the discrete model of the real world is given by

$$\boldsymbol{x}_k = \Phi_k \boldsymbol{x}_{k-1} + \boldsymbol{w}_k \tag{23}$$

where Φ_k is the matrix that propagates the states from one sampling instant to the next and $\boldsymbol{w}_k$ is white process noise vector

[3 , pp. 131]. From the previous section, it is known that the Kalman filtering equation (18) is

$$\hat{\boldsymbol{x}}_k = \Phi_k \hat{\boldsymbol{x}}_{k-1} + \boldsymbol{K}_k(\boldsymbol{z}_k - \boldsymbol{H}\Phi_k \hat{\boldsymbol{x}}_{k-1})$$

In the above equation, $\boldsymbol{z}_k$ is the discrete measurement vector (16).

From the Fundamental Theorem of Estimation Theory,

$$\hat{\boldsymbol{x}}_k^- = \Phi_k \hat{\boldsymbol{x}}_{k-1} \quad (24)$$

where $\hat{\boldsymbol{x}}_k^-$ is the a priori estimate of $\boldsymbol{x}_k$ (before the measurement at t_k) [5]. The error in the current estimate (after the measurement at t_k) is given by

$$\boldsymbol{e}_k = \boldsymbol{x}_k - \hat{\boldsymbol{x}}_k \quad (25)$$

while the a priori error (before the update) in the estimate is given by

$$\boldsymbol{e}_k^- = \boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^- \quad (26)$$

On substituting (23) and (24) and subsequently (25) in (26), $\boldsymbol{e}_k^-$ can be alternatively expressed as

$$\boldsymbol{e}_k^- = \Phi_k \boldsymbol{e}_{k-1} + \boldsymbol{w}_k \quad (27)$$

From the definition of $\boldsymbol{P}_k$,

$$\boldsymbol{P}_k = E\{\boldsymbol{e}_k \boldsymbol{e}_k{}^T\} = E\{(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k)(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k)^T\} \quad (28)$$

and according to the definition of $\boldsymbol{M}_k$,

$$\boldsymbol{M}_k = E\{\boldsymbol{e}_k^- \boldsymbol{e}_k^{-T}\} = E\{(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^-)(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^-)^T\} \quad (29)$$

Substituting (27) in (29) (first part) and using the property of the sum of 2 orthogonal terms i.e.,

$$E\{\boldsymbol{e}_k \boldsymbol{w}_k{}^T\} = [\boldsymbol{0}] \quad (30)$$

the modified expression for $\boldsymbol{M}_k$ is obtained as

$$\boldsymbol{M}_k = \Phi_k E\{\boldsymbol{e}_{k-1} \boldsymbol{e}_{k-1}{}^T\} \Phi_k{}^T + E\{\boldsymbol{w}_k \boldsymbol{w}_k{}^T\} \quad (31)$$

Using the definitions of $\boldsymbol{P}_{k-1}$ and $\boldsymbol{Q}_k$, (31) can be rewritten as

$$\boldsymbol{M}_k = \Phi_k \boldsymbol{P}_{k-1} \Phi_k{}^T + \boldsymbol{Q}_k$$

which is the same as (19) (the first matrix Riccati equation).

Using (24), the Kalman filtering equation (18) can also be expressed as

$$\hat{\boldsymbol{x}}_k = \hat{\boldsymbol{x}}_k^- + \boldsymbol{K}_k(\boldsymbol{z}_k - \boldsymbol{H}\hat{\boldsymbol{x}}_k^-) = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H})\hat{\boldsymbol{x}}_k^- + \boldsymbol{K}_k \boldsymbol{z}_k \quad (32)$$

On substituting the expression for $\boldsymbol{z}_k$ in (32) (second part) and rearranging the terms,

$$\hat{\boldsymbol{x}}_k = \hat{\boldsymbol{x}}_k^- + \boldsymbol{K}_k\boldsymbol{H}(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^-) + \boldsymbol{K}_k \boldsymbol{v}_k \quad (33)$$

On subtracting (33) from $\boldsymbol{x}_k$ and grouping common terms,

$$\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H})(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^-) - \boldsymbol{K}_k \boldsymbol{v}_k \quad (34)$$

Substitution of (34) in the expression for $\boldsymbol{P}_k$ (second part of (28)) gives

$$\boldsymbol{P}_k = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H})E\{(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^-)(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^-)^T\}(\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H})^T + \boldsymbol{K}_k E\{\boldsymbol{v}_k \boldsymbol{v}_k{}^T\} \boldsymbol{K}_k{}^T \quad (35)$$

Using (29) (second part) and (17), (35) can be modified to

$$\boldsymbol{P}_k = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H})\boldsymbol{M}_k(\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H})^T + \boldsymbol{K}_k \boldsymbol{R}_k \boldsymbol{K}_k{}^T \quad (36)$$

On expanding (36),

$$\boldsymbol{P}_k = \boldsymbol{M}_k - \boldsymbol{K}_k\boldsymbol{H}\boldsymbol{M}_k - \boldsymbol{M}_k\boldsymbol{H}^T\boldsymbol{K}_k{}^T + \boldsymbol{K}_k(\boldsymbol{H}\boldsymbol{M}_k\boldsymbol{H}^T + \boldsymbol{R}_k)\boldsymbol{K}_k{}^T \quad (37)$$

According to the rule of minimum mean-square error,

$$\sum_{i=0}^{n} E\{(\boldsymbol{e}_{k_i})^2\} = tr[E\{\boldsymbol{e}_k \boldsymbol{e}_k{}^T\}] = tr[\boldsymbol{P}_k] \quad (38)$$

In order to minimize the variance of the error in the estimate over all possible choices of $\boldsymbol{K}_k$ (gains), the derivative of trace of $\boldsymbol{P}_k$ (37) with respect to the gain matrix must be equated to zero i.e.,

$$\frac{\partial\, tr[\boldsymbol{P}_k]}{\partial \boldsymbol{K}_k} = [\boldsymbol{0}] \quad (39)$$

The basic properties of trace of a matrix and derivative of trace of a matrix i.e.,

$$tr[A] = tr[A^T] \quad (40)$$

$$\frac{d\, tr[AB]}{dA} = B^T \quad (41)$$

$$\frac{d\, tr[ACA^T]}{dA} = 2AC \quad (42)$$

where A and B are square matrices and C is a symmetric matrix, are used in simplifying the expanded form of (39), in which $\boldsymbol{K}_k$ and $\boldsymbol{M}_k$ are square matrices while $(\boldsymbol{H}\boldsymbol{M}_k\boldsymbol{H}^T + \boldsymbol{R}_k)$ is a symmetric matrix. Thus,

$$\frac{\partial\, tr[\boldsymbol{P}_k]}{\partial \boldsymbol{K}_k} = -2\boldsymbol{M}_k\boldsymbol{H}^T + 2\boldsymbol{K}_k(\boldsymbol{H}\boldsymbol{M}_k\boldsymbol{H}^T + \boldsymbol{R}_k) = [\boldsymbol{0}] \quad (43)$$

The simplified form of (43) is given by

$$\boldsymbol{K}_k = \boldsymbol{M}_k\boldsymbol{H}^T(\boldsymbol{H}\boldsymbol{M}_k\boldsymbol{H}^T + \boldsymbol{R}_k)^{-1}$$

which is the same as (20) (the second matrix Riccati equation).

Substitution of the expression for $\boldsymbol{K}_k$ (20) in the last term of (37) i.e., $\boldsymbol{K}_k(\boldsymbol{H}\boldsymbol{M}_k\boldsymbol{H}^T + \boldsymbol{R}_k)\boldsymbol{K}_k{}^T$ gives

$$\boldsymbol{P}_k = \boldsymbol{M}_k - \boldsymbol{K}_k\boldsymbol{H}\boldsymbol{M}_k - \boldsymbol{M}_k\boldsymbol{H}^T\boldsymbol{K}_k{}^T + \boldsymbol{M}_k\boldsymbol{H}^T(\boldsymbol{H}\boldsymbol{M}_k\boldsymbol{H}^T + \boldsymbol{R}_k)^{-1}(\boldsymbol{H}\boldsymbol{M}_k\boldsymbol{H}^T + \boldsymbol{R}_k)\boldsymbol{K}_k{}^T \quad (44)$$

On performing basic matrix operations and further simplification, (44) yields

$$\boldsymbol{P}_k = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H})\boldsymbol{M}_k$$

which is the same as (21) (the third matrix Riccati equation).

C. IMU (Single Sensor) as Measurement Source

For the first case, only acceleration was measured using the IMU's accelerometers. **Tables I** and **II** show the different matrices for the 2 models considered.

In **Table I**, $\sigma_{\ddot{x}}{}^2$ is the variance in acceleration measurement and its value was found to be $0.001(m/s^2)^2$. The IMU was kept stationary and the accelerations were recorded for 5 minutes. Using the acquired data, the variance for Z acceleration was computed. This procedure was repeated three times and the average variance was calculated. T_s is the sampling time of acceleration measurement (of the IMU) and its value is 0.04 s.

TABLE I
IMPORTANT MATRICES FOR DIFFERENT ORDER POLYNOMIAL KALMAN FILTERS
(SINGLE SENSOR OUTPUT)

Model (Polynomial Order)	Systems Dynamics	Fundamental	Measurement	Measurement Noise
Constant Acceleration (2)	$\boldsymbol{F} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$	$\Phi_k = \begin{bmatrix} 1 & T_s & \frac{{T_s}^2}{2} \\ 0 & 1 & T_s \\ 0 & 0 & 1 \end{bmatrix}$	$\boldsymbol{H} = [0 \quad 0 \quad 1]$	$\boldsymbol{R}_k = {\sigma_{\ddot{x}}}^2$
Constant Jerk (3)	$\boldsymbol{F} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$	$\Phi_k = \begin{bmatrix} 1 & T_s & \frac{{T_s}^2}{2} & \frac{{T_s}^3}{6} \\ 0 & 1 & T_s & \frac{{T_s}^2}{2} \\ 0 & 0 & 1 & T_s \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$\boldsymbol{H} = [0 \quad 0 \quad 1 \quad 0]$	$\boldsymbol{R}_k = {\sigma_{\ddot{x}}}^2$

TABLE II
DISCRETE PROCESS-NOISE MATRIX VARIES WITH SYSTEM ORDER (SINGLE SENSOR OUTPUT)

Model (Polynomial Order)	Continuous Process-Noise	Fundamental	Discrete Process-Noise
Constant Acceleration (2)	$\boldsymbol{Q} = \Phi_s \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$\Phi_k = \begin{bmatrix} 1 & T_s & \frac{{T_s}^2}{2} \\ 0 & 1 & T_s \\ 0 & 0 & 1 \end{bmatrix}$	$\boldsymbol{Q}_k = \Phi_s \begin{bmatrix} \frac{{T_s}^5}{20} & \frac{{T_s}^4}{8} & \frac{{T_s}^3}{6} \\ \frac{{T_s}^4}{8} & \frac{{T_s}^3}{3} & \frac{{T_s}^2}{2} \\ \frac{{T_s}^3}{6} & \frac{{T_s}^2}{2} & T_s \end{bmatrix}$
Constant Jerk (3)	$\boldsymbol{Q} = \Phi_s \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$\Phi_k = \begin{bmatrix} 1 & T_s & \frac{{T_s}^2}{2} & \frac{{T_s}^3}{6} \\ 0 & 1 & T_s & \frac{{T_s}^2}{2} \\ 0 & 0 & 1 & T_s \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$\boldsymbol{Q}_k = \Phi_s \begin{bmatrix} \frac{{T_s}^6}{36} & \frac{{T_s}^5}{12} & \frac{{T_s}^4}{6} & \frac{{T_s}^3}{6} \\ \frac{{T_s}^5}{12} & \frac{{T_s}^4}{4} & \frac{{T_s}^3}{2} & \frac{{T_s}^2}{2} \\ \frac{{T_s}^4}{6} & \frac{{T_s}^3}{2} & {T_s}^2 & T_s \\ \frac{{T_s}^3}{6} & \frac{{T_s}^2}{2} & T_s & 1 \end{bmatrix}$

In **Table II**, Φ_s is the noise spectral which is a fudge factor that accounts for the lack of knowledge of the real world and its value was found to be 10^{-11} in case of constant acceleration model and 10^{-16} in case of constant jerk model. Through trial and error, those values were chosen at which the estimates followed the trends of the measurements to a large extent.

D. GPS and IMU (Two Sensors) as Measurement Sources

In this case, acceleration was measured using the IMU's accelerometer and position was obtained using a GPS receiver. **Tables III** and **IV** show the different matrices for the 2 models i.e., constant acceleration and constant jerk.

In **Table III**, ${\sigma_x}^2$ is the variance in position measurement and its value was found to be 9 m^2. In this case, the sampling time for position measurement has a value of 0.22 s. The NavStik module (GPS) was kept stationary and the positions were recorded for 5 minutes. Subsequently, the variance in position was computed.

Table IV is the same as **Table II** with the only difference being in the values assumed for the constants. In **Table IV**, the Φ_s values chosen were the same as mentioned previously. However, the sampling time (T_s) for position measurement has a value of 0.22 s.

VI. Simulation Results

This section shows simulation results of state estimation assuming both the models of Kalman filter and considering both GPS and IMU data. The Kalman filter algorithm was implemented utilizing multiple test flight data, and some of the results are presented here.

TABLE III
IMPORTANT MATRICES FOR DIFFERENT ORDER POLYNOMIAL KALMAN FILTERS (2 SENSORS OUTPUT)

Model (Polynomial Order)	Systems Dynamics	Fundamental	Measurement	Measurement Noise
Constant Acceleration (2)	$\boldsymbol{F} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$	$\Phi_k = \begin{bmatrix} 1 & T_s & \frac{{T_s}^2}{2} \\ 0 & 1 & T_s \\ 0 & 0 & 1 \end{bmatrix}$	$\boldsymbol{H} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$\boldsymbol{R}_k = \begin{bmatrix} {\sigma_x}^2 & \sigma_x\sigma_{\dot{x}} \\ \sigma_{\dot{x}}\sigma_x & {\sigma_{\dot{x}}}^2 \end{bmatrix}$
Constant Jerk (3)	$\boldsymbol{F} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$	$\Phi_k = \begin{bmatrix} 1 & T_s & \frac{{T_s}^2}{2} & \frac{{T_s}^3}{6} \\ 0 & 1 & T_s & \frac{{T_s}^2}{2} \\ 0 & 0 & 1 & T_s \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$\boldsymbol{H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$	$\boldsymbol{R}_k = \begin{bmatrix} {\sigma_x}^2 & \sigma_x\sigma_{\dot{x}} \\ \sigma_{\dot{x}}\sigma_x & {\sigma_{\dot{x}}}^2 \end{bmatrix}$

TABLE IV
DISCRETE PROCESS-NOISE MATRIX VARIES WITH SYSTEM ORDER (2 SENSORS OUTPUT)

Model (Polynomial Order)	Continuous Process-Noise	Fundamental	Discrete Process-Noise
Constant Acceleration (2)	$\boldsymbol{Q} = \Phi_s \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$\Phi_k = \begin{bmatrix} 1 & T_s & \frac{{T_s}^2}{2} \\ 0 & 1 & T_s \\ 0 & 0 & 1 \end{bmatrix}$	$\boldsymbol{Q}_k = \Phi_s \begin{bmatrix} \frac{{T_s}^5}{20} & \frac{{T_s}^4}{8} & \frac{{T_s}^3}{6} \\ \frac{{T_s}^4}{8} & \frac{{T_s}^3}{3} & \frac{{T_s}^2}{2} \\ \frac{{T_s}^3}{6} & \frac{{T_s}^2}{2} & T_s \end{bmatrix}$
Constant Jerk (3)	$\boldsymbol{Q} = \Phi_s \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$\Phi_k = \begin{bmatrix} 1 & T_s & \frac{{T_s}^2}{2} & \frac{{T_s}^3}{6} \\ 0 & 1 & T_s & \frac{{T_s}^2}{2} \\ 0 & 0 & 1 & T_s \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$\boldsymbol{Q}_k = \Phi_s \begin{bmatrix} \frac{{T_s}^6}{36} & \frac{{T_s}^5}{12} & \frac{{T_s}^4}{6} & \frac{{T_s}^3}{6} \\ \frac{{T_s}^5}{12} & \frac{{T_s}^4}{4} & \frac{{T_s}^3}{2} & \frac{{T_s}^2}{2} \\ \frac{{T_s}^4}{6} & \frac{{T_s}^3}{2} & {T_s}^2 & T_s \\ \frac{{T_s}^3}{6} & \frac{{T_s}^2}{2} & T_s & 1 \end{bmatrix}$

A. Constant Acceleration Model Using Only IMU Data

On observing **Figure 9(a)** and **Figure 9(b)**, it can be seen that the estimate of the acceleration considering only IMU data is the same as the measured acceleration i.e., there appears to be no difference between the estimate and the measurement. This indicates that the Kalman filter is not effective.

Since there is no sensor on board the helicopter which gives velocity measurement directly, the reference for comparing velocity estimate is obtained by differentiating the GPS data. It can be seen from **Figure 10** that the velocity estimate using only IMU data is very poor as it deviates from the measurement to a large extent.

Figure 11 shows that the position estimate using only IMU data is highly erroneous (deviates by a maximum value of 900 m) and cannot be used for the position estimation of the helicopter.

B. Observability

The reason for the errors in the Kalman filter estimate in the previous case can be understood by realising the concept of observability and its applications.

A system is said to be observable if, for any possible sequence of state and control vectors, the current state can be determined in finite time using only the outputs, i.e. from the system's outputs it is possible to determine the behaviour of the entire system.

Consider the discrete-time system

$$\boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k) \quad (45)$$

$$\boldsymbol{y}(k) = \boldsymbol{C}\boldsymbol{x}(k) + \boldsymbol{D}\boldsymbol{u}(k) \quad (46)$$

The above system is observable if and only if the observability matrix has a rank equal to *n* where *n* is the order of the system model (number of state variables) [7]. The observability matrix is defined as

$$M_{obs} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix} \quad (47)$$

Consider the constant acceleration Kalman filter model where order $n = 3$.

Case 1: Only IMU output is taken as measurement

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \qquad C = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$$

$$M_{obs} = \begin{bmatrix} C \\ CA \\ CA^2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Since rank(M_{obs}) = 1 < 3, the system is not observable.

Case 2: Both GPS and IMU outputs are taken as measurements

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \qquad C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$M_{obs} = \begin{bmatrix} C \\ CA \\ CA^2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

In this case, rank(M_{obs}) = 3 = 3. Hence, the system is observable.

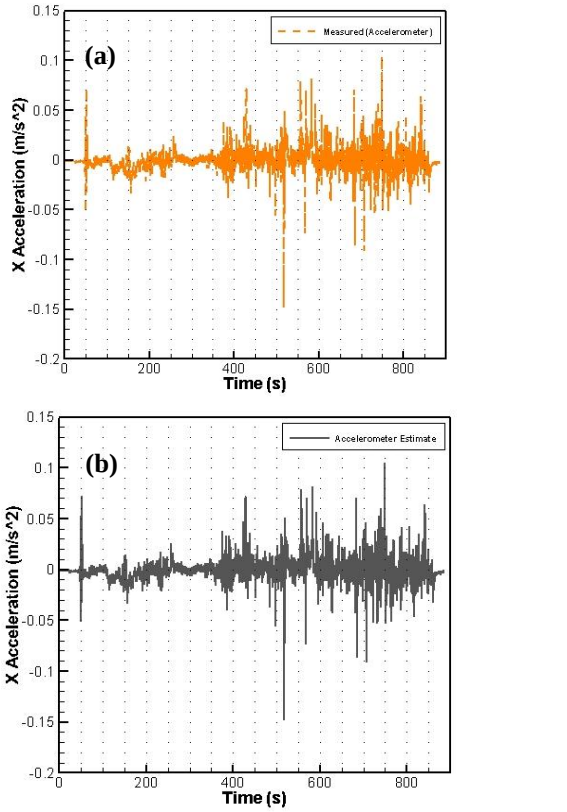


Figure 9 (a) Measured X acceleration vs time; (b) X acceleration estimate vs time

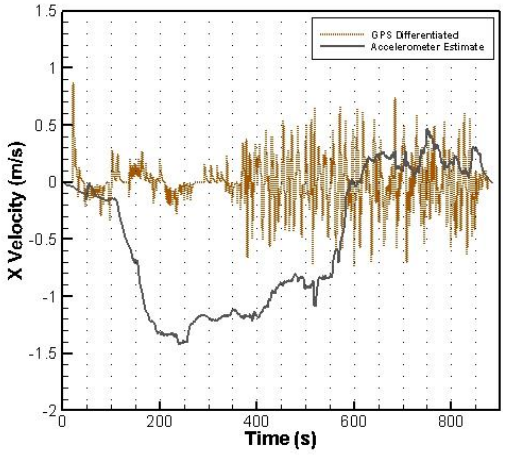


Figure 10 X GPS data differentiated and X velocity estimate vs time

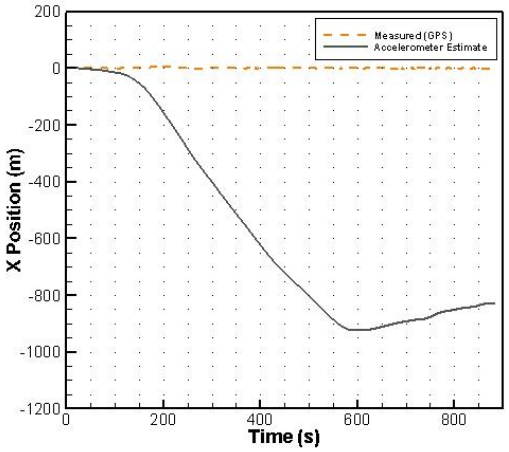


Figure 11 Measured X position and X position estimate vs time

Thus, it can be concluded that the system is observable only when position and acceleration are measured and not when only accelerometer data is provided. It is also possible to observe the system when only GPS data is used but the data acquisition rate from the NavStik is quite low (4.5 Hz) whereas the accelerometer/IMU gives data at 25 Hz. Hence, it is important to consider data from both sensors for better state estimation. These proofs also imply that Kalman filter algorithm works only when the observability criterion is satisfied. The above verification can also be made for constant jerk model.

C. Constant acceleration model using both GPS and IMU data

Figure 12 clearly indicates that the Kalman filter estimate is better than the accelerometer measurement (raw data) as there is no deviation and the estimate follows the trend of the measurement. This means that the algorithm is working properly.

From **Figures 13** and **14**, it can be inferred that the Kalman filter is effective as the estimates follow the trends of the respective measurements. Furthermore, the estimate for velocity appears to be better than the estimate for position as there are no deviations in the former case.

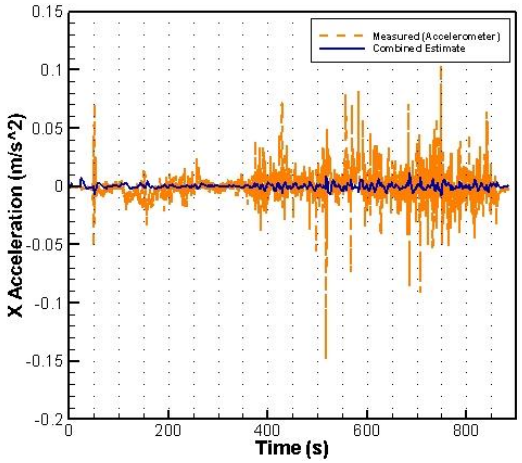


Figure 12 Measured X acceleration and combined estimate of X acceleration vs time

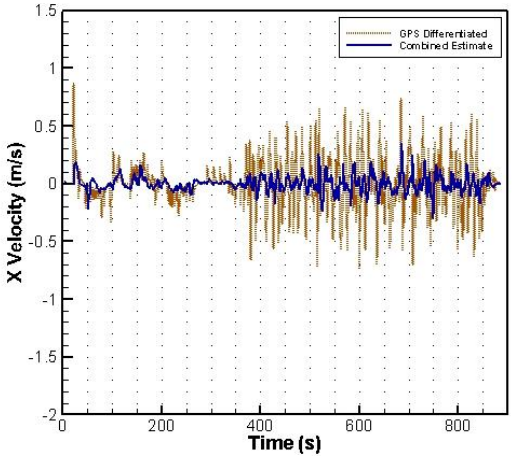


Figure 13 X GPS data differentiated and combined estimate of X velocity vs time

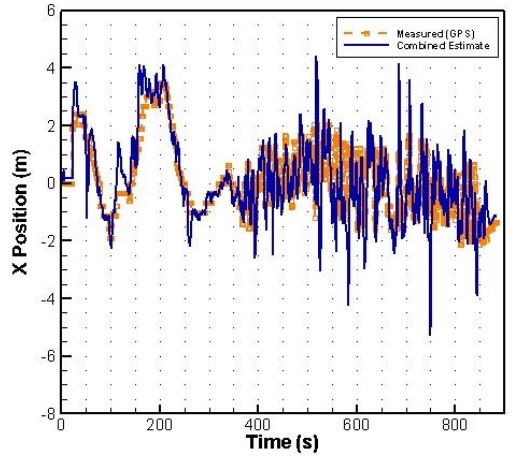


Figure 14 Measured X position and combined estimate of X position vs time

D. Constant Jerk Model Using Both GPS and IMU Data

Although the estimates of constant acceleration model seemed satisfactory, the constant jerk model was also simulated because in reality, the helicopter does not move with a constant acceleration and a better approximation of the real world would be the constant jerk model.

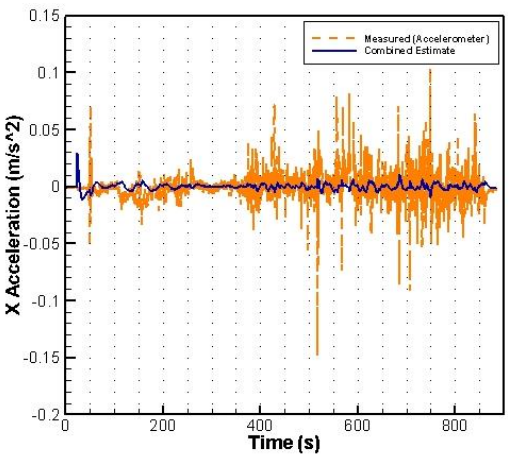


Figure 15 Measured X acceleration and combined estimate of X acceleration vs time

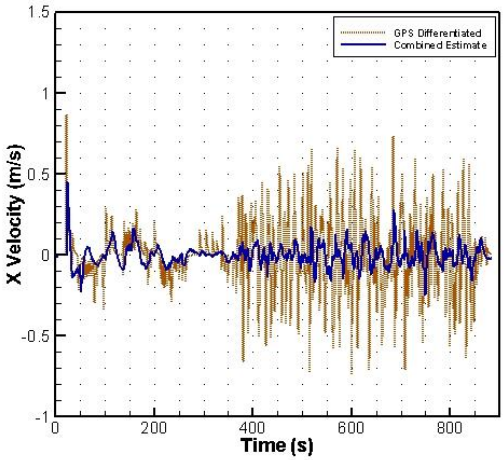


Figure 16 X GPS data differentiated and combined estimate of X velocity vs time

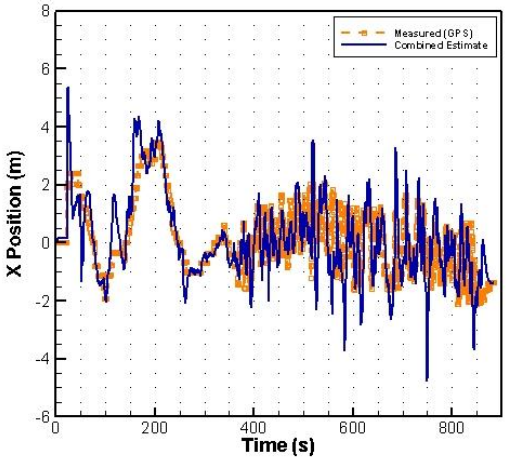


Figure 17 Measured X position and combined estimate of X position vs time

Figures 15, **16** and **17** show that respective estimates follow the expected trends and are marginally better than constant acceleration model estimates (beyond 300 s). This conclusion is further supported by observing the values assumed by Φ_s for constant acceleration model (10^{-11}) and constant jerk model (10^{-16}). The fact that the latter has a smaller value proves that

we are closer to the real world model. The ideal model which replicates the real world would have $\Phi_s = 0$. Hence if the value of Φ_s is closer to 0, it is implied that the model is better. Another observation which can be made is that during the time period 0-300 s, the constant acceleration model estimates (refer **Figures 12**, **13** and **14**) seem better than the constant jerk model estimates. This is because up to 300 s, the helicopter was stationary and this situation is best modeled by constant acceleration. One of the important aspects of Kalman filtering includes appropriate selection of the system model. Consideration of an alternate model would lead to incorrect estimates, which is visible in **Figures 15**, **16** and **17** for the first 300 s (when compared with the respective constant acceleration model results).

The linear Kalman filter implemented for state estimation is the first step to predicting the position of the helicopter. This study will be extended further to more advanced models like extended Kalman filter.

VII. Concluding Remarks

The first portion of the paper presented the calibration results of the IMU Microstrain 3DM-GX1, which was mounted on the helicopter, and the experiments which were conducted to estimate angular acceleration, velocity and acceleration by using the accelerometer, gyroscope and Euler Angle data and their comparison with the potentiometer data. It was found that the gyroscope and Euler Angle data consistently matched with the reference in all the graphs. However, the accelerometer plots were inconsistent and it could be concluded that the accelerometers are quite sensitive and easily affected by noise as well as other possible errors. These errors get propagated on integration and lead to large deviations in velocity and position. Errors like bias, scale factor and drift were found to be quite small and negligible. Furthermore, it was found that accelerometers detect changes and provide accurate data only within certain limits of angular rates of the IMU motion. Similar results were obtained for other orientations of the IMU. The experiments also showed that the IMU yields more accurate results in Yaw motion, with or without small inclinations along other axes, compared to Roll and Pitch motions.

The next section of the paper required the understanding of Kalman filtering and its applications in state estimation. Using previous test flight data, simulations were done to estimate position, velocity and acceleration of the helicopter. Kalman filtering algorithms were implemented using only IMU data as well as both GPS and IMU data. It was found that Kalman filters do not work when only IMU data is measured and this was theoretically verified using the concept of observability. This was followed by comparing two models – constant acceleration and constant jerk while considering both GPS and IMU data. It was found that both models showed good estimations with constant jerk being a better approximation of the real world model. These inferences can be drawn even for Y axis parameters as the results were similar for both X axis and Y axis estimates. The paper presents only the linear Kalman filter implementation which is a relatively simple math model. Further studies will be carried out to estimate position using other sophisticated models like extended Kalman filter.

Acknowledgment

The authors would like to thank the project associates and the workshop staff of the Helicopter Lab at the Department of Aerospace Engineering, IIT Kanpur, India for their help. They would also like to thank the Department of Science and Technology, India, the sponsor of the Autonomous Mini Helicopter project.

References

[1] Wikipedia, "Unmanned aerial vehicle". VIEW ITEM

[2] P. Lamb, and S. Thiébaux, "Avoiding Explicit Map-Matching in Vehicle Location," CSIRO Mathematical and Information Sciences, *6th ITS World Congress (ITS-99)*, Toronto (Canada), November 1999, pp. 1. [Online], VIEW ITEM

[3] P. Zarchan, and H. Musoff, *Fundamentals of Kalman Filtering: A Practical Approach*, 3rd ed., Vol. 232, Progress in Astronautics and Aeronautics, AIAA, Virginia, 2009, Chap. 4.

[4] R. L. Needham, Jr., "Calibration of Strapdown System Accelerometer Dynamic Errors," M.S. Dissertation, Dept. of Aeronautics and Astronautics, MIT, Boston, 1994, pp. 24. [Online], VIEW ITEM

[5] M. S. Fadali, "Discrete Kalman Filter", *Course EE 782: Random Signal Analysis and Estimation Theory*, Dept. of Electrical Engineering, University of Nevada, Reno [Online]. VIEW ITEM

[6] C. Venkatesan, B. B. Swaroop, P. Haritha, and R. Gupta, "Development of Autonomous Mini Helicopter: Challenges faced," IIT Kanpur, *AERO INDIA 2013 International Seminar*, Bangalore (India), February 2013, pp. 5. [Online], VIEW ITEM

[7] B. Southall, B. F. Buxton, and J. A. Marchant, "Controllability and Observability: Tools for Kalman Filter Design," in *Proceedings of the 9th British Machine Vision Conference*, Southampton, September 1998, pp. 1–10. [Online], VIEW ITEM

[8] U. Coppa, A. Guarnieri, F. Pirotti, and A. Vettore, "Accuracy enhancement of unmanned helicopter positioning with low-cost system," *Applied Geomatics*, Springer, Vol. 1, Issue 3, September 2009, pp. 85-95 [Online], VIEW ITEM

[9] J. L. Cookson, "A Method for Testing the Dynamic Accuracy of Micro-Electro-Mechanical Systems (MEMS) Magnetic, Angular Rate, and Gravity (MARG) Sensors for Inertial Navigation Systems (INS) and Human Motion Tracking Applications," M.S. Dissertation, Dept. of Electrical Engineering, Naval Postgraduate School, California, June 2010 [Online]. VIEW ITEM

[10] George Arshal, "Error Equations of Inertial Navigation," *Journal of Guidanec, Control, and Dynamics*, Vol. 10, No. 4, July-August 1987, pp. 351-358. [Online], VIEW ITEM

[11] B. Friedland, "On the Calibration Problem," *IEEE Transactions on Automatic Control*, Vol. AC-22, No. 6, December 1977, pp. 899-905. [Online], VIEW ITEM

[12] A. Kourepenis, J. Borentein, J. Connelly, R. Elliott, P. Ward, and M. Weinberg, "Performance of MEMS inertial sensors," *IEEE Position Location and Navigation Symposium*, April 1998, pp. 1–8. [Online], VIEW ITEM

[13] M. S. Grewal, V. Henderson, and R. Miyasako, "Application of Kalman Filtering to the Calibration and Alignment of Inertial Navigation Systems," *IEEE Transactions on Automatic Control*, Vol. 36, No. 1, January 1991, pp. 4-13. [Online], VIEW ITEM

[14] K. Maenaka, "MEMS inertial sensors and their applications," *5th International Conference on Networked Sensing Systems*, June 2008, pp. 71–73. [Online], VIEW ITEM