

A Methodology for Design and Analysis of Sensor Fusion with Real Data in UAV platforms

Jesús García, Jose Manuel Molina, Jorge Trincado

jgherrer@inf.uc3m.es, molina@ia.uc3m.es, jtrincad@di.uc3m.es

Grupo de Inteligencia Artificial Aplicada of the University Carlos III de Madrid

Abstract— This paper presents a methodology to design sensor fusion parameters using real performance indicators of navigation in UAVs based on PixHawk flight controller and peripherals. This methodology and the selected performance indicators allows to find the best parameters for the fusion system of a determined configuration of sensors and a predefined real mission. The selected real platform is described with stress on available sensors and data processing software, and the experimental methodology is proposed to characterize sensor data fusion output and determine the best choice of parameters using quality measurements of tracking output with performance metrics not requiring ground truth.

Keywords— UAVs sensor fusion, EKF, Real Data Analysis, System Design

I. INTRODUCTION

Unmanned aerial systems (UAS) have recently evolved with exponential growths thanks to development of technologies of navigation, sensing or obstacle avoidance based on available micro-electro-mechanical systems (MEMS). The research has focused in getting higher degrees of autonomy thanks to enhanced navigation, but also in robust methods to protect missions against intended attacks, mainly interference (jamming) and supplant (spoofing) of GNSS signal.

The main problem in navigation focuses on improving GPS with the ability to provide accurate navigation output when GPS data becomes unavailable due to unexpected outages or intentional problems (jamming or spoofing) in certain environments. Therefore, an approach based on the fusion of complementary sensors is essential, resorting to the fundamental equations of navigation and the characterization of the errors committed by each data source. This area has become popular due to the ubiquity of GPS and the availability of inertial sensors based on inexpensive MEMS components [1], [2], [3]. The integration of these complementary technologies allows compact and robust navigation solutions to determine attitude and location, so that the vehicle can determine its state in a robust way and use appropriate control techniques for autonomy. Other more drastic options for non-dependence on the GPS signal involve the deployment of autonomous localization systems such as the recognition of the environment by artificial vision [4] or location by means of electromagnetic beacons [5], with the associated cost of developing a complementary infrastructure.

Complementary to navigation technologies, the use of lasers in combination with other range detection sensors (sonar, radar, video), allows to extend the navigation conditions and obstacle avoidance. In the air vehicles (UAVs), the integration requirements (consume, weight, dimensions) are much more restrictive but, even so, it is a line in continuous development [6],[7],[8]. Regarding protection against spoofing, there is also a noticeable research activity [9]. Protections include diverse strategies ranging from simple actions such as monitor the communication channel, to cryptographic authentication, discrimination based on the level of signal, time of arrival, multi-antenna systems, wave polarization, Doppler shift and arrival angle, etc., as described in [10],[11].

Therefore, research of robust and general techniques to integrate complementary data sources has become essential for this type of systems. In addition to theoretical developments, it is of vital importance the availability of equipment and experimental environments to validate the robustness of the solutions working in real-world conditions. The integration of sensors has to be based on the definition of parameters of the tracking system that should be adjusted to improve performance in a predefined set of missions with a defined set of sensors. The methodology proposed in this paper assumes that real system adjustments will be based on a real platform with predefined flight missions so that, in this context, the best parameters could be obtained analyzing the real operation of sensors and real output. Simulation of UAV environments are not enough to evaluate in a thorough way these systems in real conditions of real missions.

This paper presents the selected environment for real experimentation, the proposed methodology and a selection of available data sources and effects of data processing techniques on the quality of the navigation solution. The main contribution of this paper is the briefing of a methodology used for adapting filter parameters to real conditions, and further systematic analysis of available real data. Section II introduces the selected working platform, detailing the architecture of its software and the vehicles we have made to test its capacities and collect data. Section III presents the proposed methodology and evaluation metrics, section IV explains experimental environment and analyses the recorded sensor data, and section V contains the characterization and analysis of the PixHawk Px4 system filter and fusion algorithms following the presented methodology. Finally, section VI contains summarizes the conclusions derived from this work.

II. THE PIXHAWK AND PX4 UAV SYSTEM

A. Architecture

Unmanned vehicles must be able to control their attitude and position by means of automatic control algorithms, they are controlled by a computer that integrates data from some electro-mechanical sensors and any local or global positioning system, and applies any output control system to change its location using any locomotion system. This controller is usually an embedded microcontroller that performs the core of all vehicle components.

This research is based in the study of the PixHawk flight controller performance. An open-hardware computer designed by 3D Robotics specifically to create autopilot vehicles, that arises from the combination of PX4FMU and PX4IO boards. Both cards, from their version v2, are integrated in the same PCB (Printed Circuit Board) giving origin to PixHawk.

B. Sensors and data sources

The PixHawk board has several sensors integrated, shown in Table I, which serve as data sources to the PX4 stack and include some processing functions¹.

¹ MPU-6000 includes a Digital Motion Processing (DMP) with programable low pass filters

TABLE I
SENSORS INTEGRATED IN THE PIXHAWK BOARD

Sensor	Type	Axes	Scale	ADC accuracy	Data rate
L3GD20H	gyroscope	3	2000 dps	16 bits	760 Hz
LSM303D	accelerometer/ magnetometer	6	$\pm 16g$ / $\pm 2\text{gauss}$	16 bits	1600 Hz/ 100 Hz
MPU-6000	accelerometer/ gyroscope	6	$\pm 16g$ / 2000 dps	16 bits	1000 Hz/ 8000 Hz
MS5611	barometer	1	1200 mbar	24 bits	1000 Hz

C. Connectivity and external sensors

Besides, PixHawk counts with high connectivity for external devices and peripherals that may increase the vehicle capabilities. At the right side, there are some specific connectors for certain peripherals, such as: 2 connectors for telemetry communication, 1 for GPS, and 1 Spektrum receiver socket. At the left side, there are some general-purpose ports and buses like 2 UART ports, 2 SPI, 1 I2C connector, 1 USB connector, 1 CAN bus connector and 3.3V and 6.6V ADC connectors. The standard configuration of a PixHawk UAV counts with interfaces to external sensors specified in the next table:

TABLE II
ADDITIONAL SENSORS WIRED TO THE PIXHAWK

Sensor	Type	Interface	Range	Accuracy	Data rate
Ublox Neo 6M	GPS	GPS port	x	2.5m	<3s
HMC5883 ¹	Magnetometer	I2C	$\pm 16G$	$\pm 12MG$	1600 Hz
Flow Sensor shield	Optical flow/gyroscope	I2C	x	<0.5m	250Hz
Range finder	Ultrasonic sonar	I2C	0-6m	0.5cm	1000 Hz
Lidar lite	Pointer lidar	I2C	0-40m	$\pm 2.5\text{cm}$	1-500Hz

¹ The GPS sensor and the HMC magnetometer are integrated in the same external shield.

These sensors allow enhancing navigation capabilities and increase the accuracy of the stabilization system measurements, what is quite important when we want to create an unmanned vehicle, because allows a more faithful image of the flying environment. The following diagram shows the main configuration of sensors used in our researching vehicles:

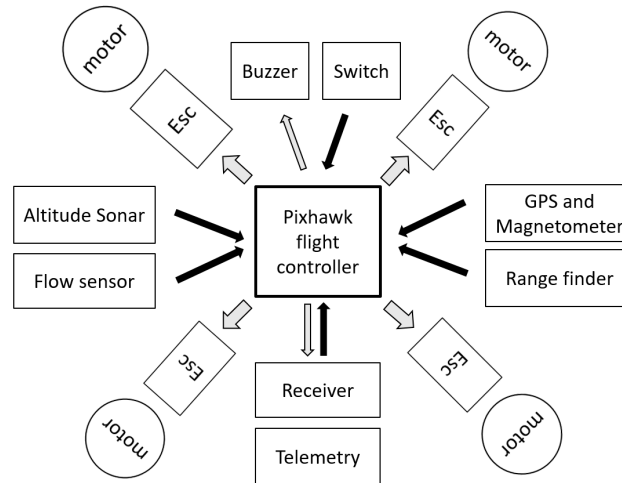


Figure 1: UAV components schema

D. Software for Flight Control and Data Processing

PX4 is the control software of PixHawk processor. It is a real-time operating system based on NuttX² and consists of two main layers: PX4 Flight Stack and PX4 Middleware. PX4 Flight Stack is the complete collection of applications embedded in PixHawk hardware for drone control, while PX4 Middleware is the interface that allows the flow of data from sensors to applications through a publish/subscribe system called uORB. uORB allows to publish the data coming from the sensors and make them available to the applications of the Flight Stack, obtaining a reactive system and totally parallelized. The outstanding modules are flight controller and sensor data processing [12].

Regarding the data processing, Px4 implements an AHRS (attitude and heading reference system) that implements different algorithms to estimate the vehicle attitude and creates a direction vector that allows the unmanned displacement. In this section we will overview some basic algorithms that run into the system during the flights.

a) Direction Cosine Matrix (DCM)

This program allows the analysis of the triaxial accelerometers and gyroscopes data to obtain a Direction Cosine Matrix [13]. It makes possible the conversion of real-time measurements into instantaneous orientation parameters of the vehicle to deliver roll, pitch and yaw angles or variations:

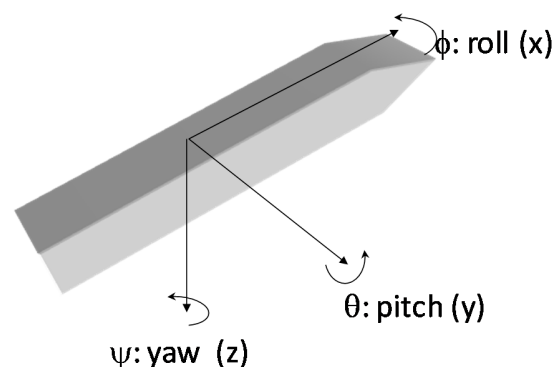


Figure 2: DCM inertial measurement conversion

² www.nuttX.org

b) *Inertial Navigation System (INS)*

This algorithm calculates the trajectories and corrections that allows the vehicle to move between single points using the DCM data. It is used to estimate the vehicle attitude with high frequency, so it is especially useful to complement the global position obtained from the GPS data.

c) *Extended Kalman Filter (EKF)*

All measurements are affected by noise that should be taken into account in the estimation of attitude and cinematic parameters. The Px4 system counts with several Extended Kalman Filter algorithms to process all sensor data in a compensation function that depends of the specific noise and accuracy characterization of each sensor, throwing high accuracy estimations of the vehicle attitude. The Px4 application counts with the possibility of applying different EKF solutions running in parallel, using different sensor measurements and states. With this implementation, it is possible to increase the accuracy and consistence of estimates even if the vehicle losses the GPS signal in certain time intervals. Next table shows the three different available EKF modes:

TABLE III
PX4 EXTENDED KALMAN FILTERS

Name	Specification
EKF1	Only use the DCM for attitude control and the Inertial navigation for AHRS reckoning for position control
EKF2	Use the GPS for 3D velocity and position. The GPS altitude could be used if barometer data is very noisy.
EKF3	If there is no GPS, it can use optical flow to estimate 3D velocity and position.

AHRS is able to select the best EKF mode for each situation and execute both EKF1 and

III. PROPOSED METHODOLOGY

In many real problems, simulated environments are used to define UAV sensors and the system parameters to optimize the system performance [14], [15]. Some problems appear with this kind of methodologies, basically how to represent in simulation all effects appearing in real conditions and the way to evaluate the parameters configuration. UAV simulation have been applied to design the control subsystem for predefined missions, but simulation of real sensors is a major problem in this kind of approach. Real UAV conditions are not easy to model in simulators. UAV are affected for atmospheric conditions and random movements of UAV platforms, so accurate simulation of input data is extremely complicated for designing system parameters. Besides, in simulation, evaluation of system is based on ground truth, comparing simulated output with desired real output, and in real conditions the ground truth is not available. On the other hand, there are analysis done with real data [16], but a systematic analysis to compare alternative configurations is of high value to take design decisions.

In this scenario, the proposed methodology tries to avoid simulated evaluation and test directly the system parameters under real conditions. The first step of this methodology is the definition of the UAV platform, the type, cinematic characteristics, set of sensors and the tracking algorithms. Once the UAV platform is defined, the methodology is composed by the following steps, as displayed in Fig. 3:

1.- Mission definition. Parameters are selected from a set of possible values for a specific mission, defined by means of several waypoints. These waypoints are used to repeat the same mission every time that a new evaluation is done.

2.- Each time a flight of the UAV passing through the predefined waypoints (mission) is carried out, the values of the sensor data are stored (position and velocity taken from GPS, inertial data, magnetometers, etc.)

3.- A set of flights, with the same waypoints defining the mission, are carried out and the corresponding sets of sensors data are stored together with system output, using several configurations of parameters for filtering. These values are postprocessed offline.

4.- The best configuration of parameters is selected, analyzing the performance metrics for the set of missions carried out over the same waypoints. This point is crucial because evaluation needs to evaluate the sensor fusion performance based on real data not in ground truth.

5.- The selected parameters are introduced in UAV system to perform real mission

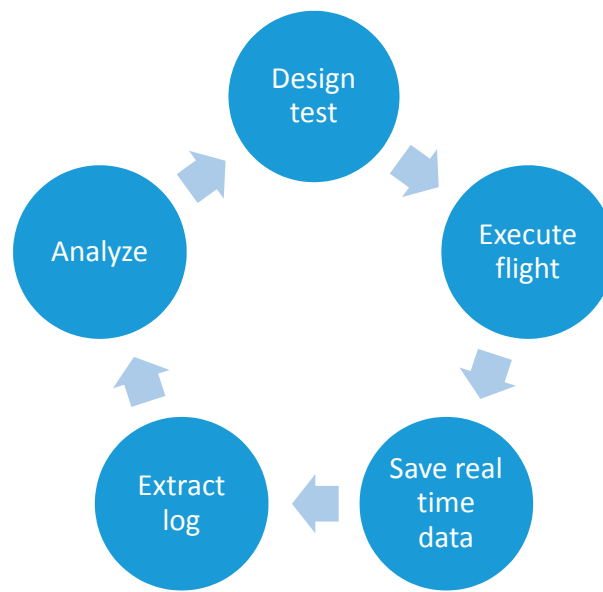


Figure 3: Proposed methodology

In order to assess the sensor fusion performance, with no available ground truth, the following indicators have been computed:

- fusion break rate
- normalized innovations

1) Averaged innovation

The innovation, or residual, is computed in the tracking filter each time an update is done for the prediction. For instance, for horizontal XY position, denoting with sub-index p the predicted track and sub-index m for measurement, and considering the average along a time window with N_T measurements, it is defined as:

$$r = \frac{1}{N_T} \sum_{i,k} (\hat{x}_{pi}[k] - x_{mi}[k])^2 + (\hat{y}_{pi}[k] - y_{mi}[k])^2 \quad (1)$$

The normalized innovation also employs the covariance matrices considering both the predicted and observation uncertainties, matrix S. The averaged value of normalized innovations defined as:

$$r_n = \frac{1}{N_T} \sum_i \begin{bmatrix} \hat{x}_{pi}[k] - x_{mi}[k] & \hat{y}_{pi}[k] - y_{mi}[k] \end{bmatrix} S_i[k]^{-1} \begin{bmatrix} \hat{x}_{pi}[k] - x_{mi}[k] \\ \hat{y}_{pi}[k] - y_{mi}[k] \end{bmatrix} \quad (2)$$

This value is a-dimensional, and represents the discrepancy between observations and predictions, averaged along the measurements contained in N_T .

Sometimes, if full covariance matrices are not available, a simplification is done and only the variances (diagonal terms) are considered:

$$r_n = \frac{1}{N_T} \sum_i \left(\frac{(\hat{x}_{pi}[k] - x_{mi}[k])^2}{\sigma_{xp}^2 + \sigma_{xm}^2} + \frac{(\hat{y}_{pi}[k] - y_{mi}[k])^2}{\sigma_{yp}^2 + \sigma_{ym}^2} \right) \quad (3)$$

2) Averaged innovation Fusion break rate

The rate of fusion break, t_{FB} , is the number of times some navigation source is declared as inconsistent in the integrity analysis and therefore de-fused (the less consistent component is removed from the system track). This may lead to tracker re-initialization or keep the system track with a component less, the faulty sensor. It is computed as:

$$t_{FB} = \frac{\sum_i \{gt_i \mid gt_i \text{ is inconsistent track}\}}{N_T} \quad (4)$$

The value, averaged along N_T measurements, is obtained counting the total number of fusion break events during the interval. The test to decide the removal of a data source is done using the innovation, this time normalized by a higher number of standard deviations, typically 5.

IV. EXPERIMENTAL ENVIRONMENT AND SENSOR DATA ANALYSIS

A. Data acquisition

The process of data acquisition was based on several flight test missions that have taken place on circuits like the one sketched in Fig. 4:

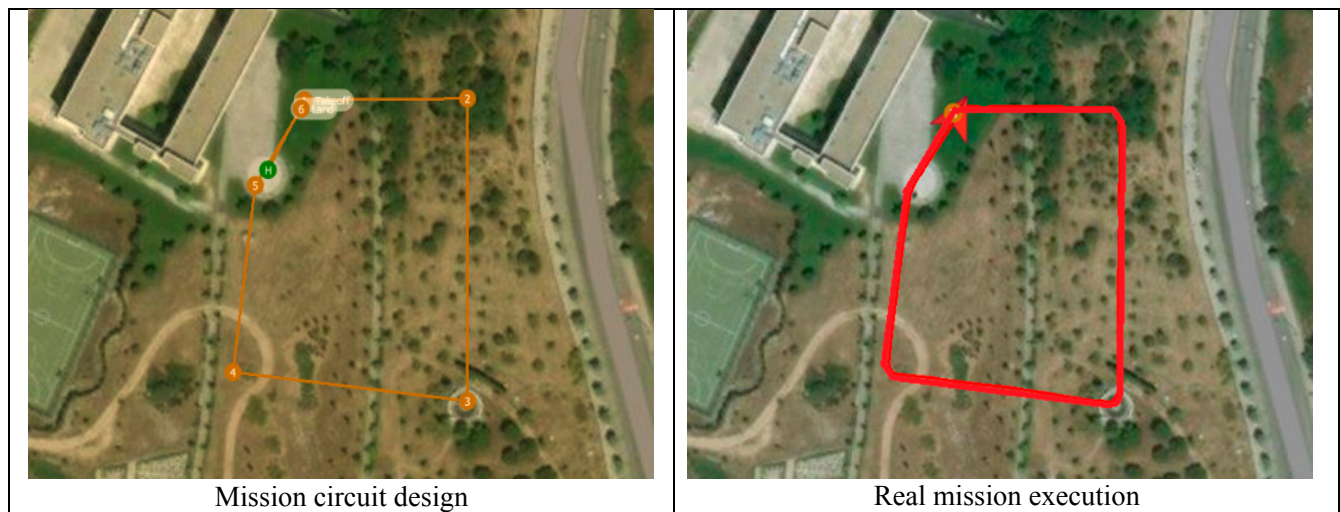


Figure 4: Test circuit diagram

For each specified mission, we applied different configuration parameters in order to analyze the performance differences between each setting up of tracking filters. The flight controller log data was saved together with the configuration settings to be analyzed. For instance, table IV, shows different tests carried out on the previous circuit and the analyzed elements of the navigation system (sensors and controllers)³

³ The static test has been performed on the floor to avoid displacements and vibrations that could disturb the readings

TABLE IV
TEST PERFORMED OVER THE SAME CIRCUIT

Name	Specification
Static test without propellers	Accelerometers and gyroscopes, noise
Static test on idle	Accelerometers and gyroscopes, noise
Unmanned flight test	GPS, inertial navigation system
Unmanned hold test	GPS, Flow Sensor, DCM
Manual flight test	PID and configuration parameters

B. Data characterization

Before exploiting sensor data for navigation enhancement, it is important to know the behaviour, limitations and main characteristics of the data we are working with, to establish a measure of the confidence and the consistence of the data. These properties are conditioned by the sensors and measurement processes. The logged data can be useful to debug the estimation algorithms and the flight performance but, in this work we will only pay attention to those data that could explain the local position estimations, the navigation system performance or the filter capabilities. Table V shows the data analysis for the main data sources available [17].

TABLE V
DATA CHARACTERIZATION

Sensor	Source	range	Accuracy	Data rate
Raw accelerometer	X,Y,ZL3GD20H MPU-6000	[-32768, +32767] DoT	$\pm 7 \text{ DoT}^2$	8000Hz
Raw X Y gyroscope	ZL3GD20H MPU-6000	[± 2000] deg/s	$\pm 3 \text{ deg/s}$	8000Hz
Roll, Pitch, Yaw	DCM over Accel/Gyro	[0,180] $^\circ$	$\pm 0.05 \text{ deg}$	1600Hz
Heading	DCM over magnetometer	[0,360] $^\circ$	$\pm 1.5 \text{ deg}$	1000 Hz
Atm. Pressure	MS5611	[10,1200]HPa	$\pm 1.2 \text{ Hpa}$	1000Hz
Sonar Floor		[0,6]m	$\pm 0.02 \text{ m}$	
Estimated position terms	localINS	Different		1000Hz
Estimated speed	Optical flow sensor	[0,16]m/s	$\pm 3.5 \text{ m/S}$	760 Hz
Speed N/E	GPS	[0, 999] mph	5%	<3s
Global position ³	GPS	Lat[-90,90] $^\circ$	3m	<3s

C. Data analysis

The logged data from different sensors are presented in this section. Data from triad of gyroscopes along body axes of UAV are displayed in Fig. 5, with different colors corresponding to six flights with the same programmed mission. We can appreciate some outliers and no significant biases

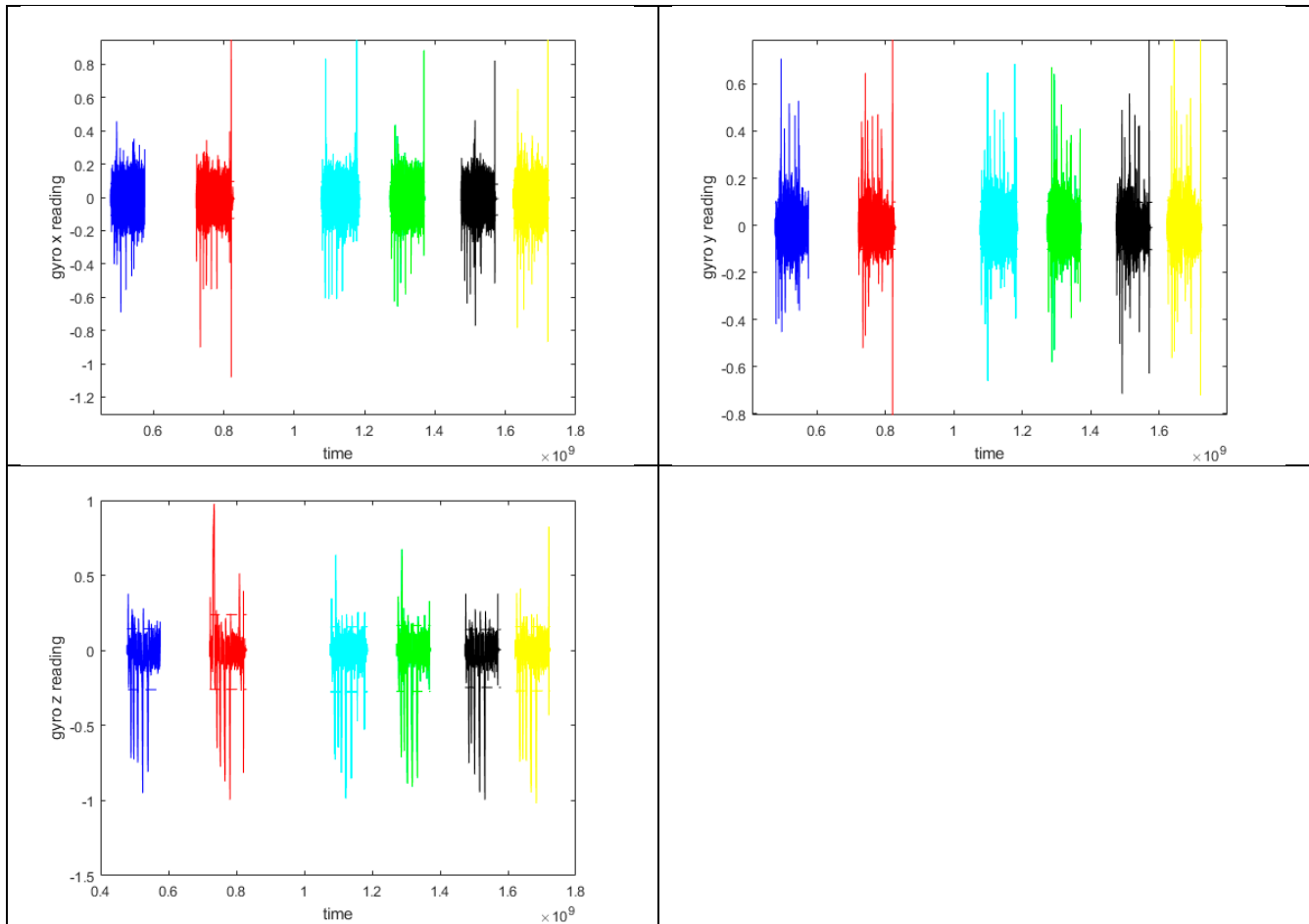
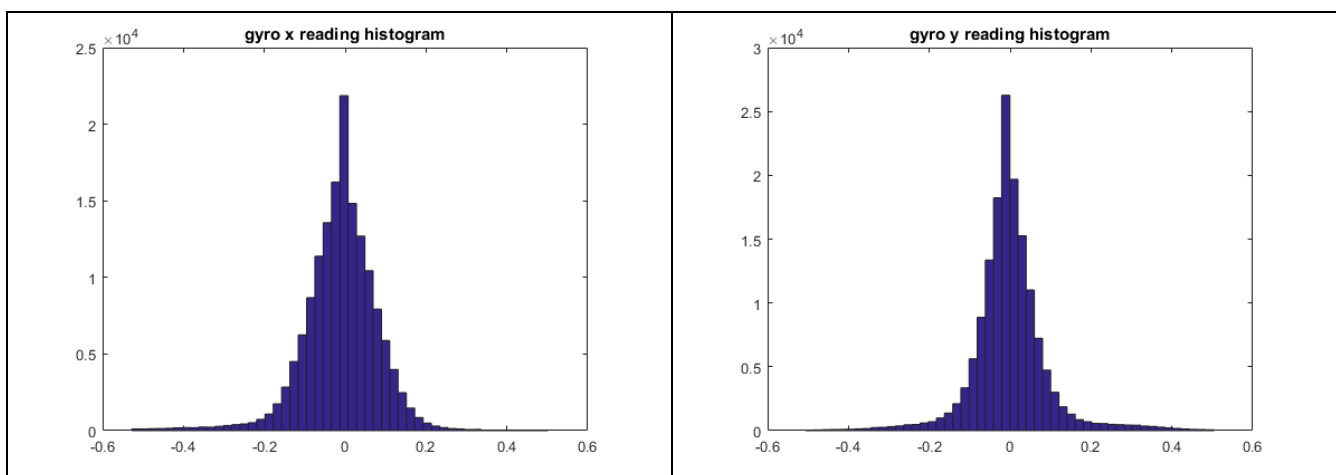


Figure 5: Gyroscope data of test flights

The corresponding histograms of previous data (with outliers removed) are presented in Fig. 6, here we can see the data is affected of Gaussian shape noise, and the mean value of turns corresponding to maneuvers can be seen in the z axis.



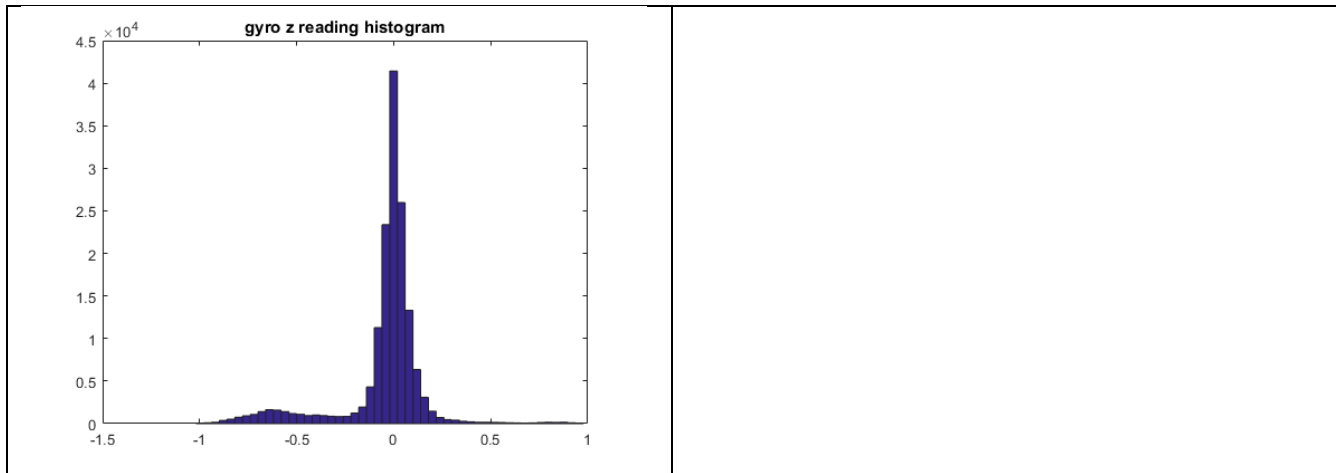


Figure 6: Gyroscope data histograms

Similar comments can be said about data from triad of accelerometers, presented in Fig. 7, with some significant outliers which were removed from tracking filter process. The histograms (outliers removed) are presented in Fig. 8, in this case the turns performed in flights are reflected in the accelerations done along the y axis.

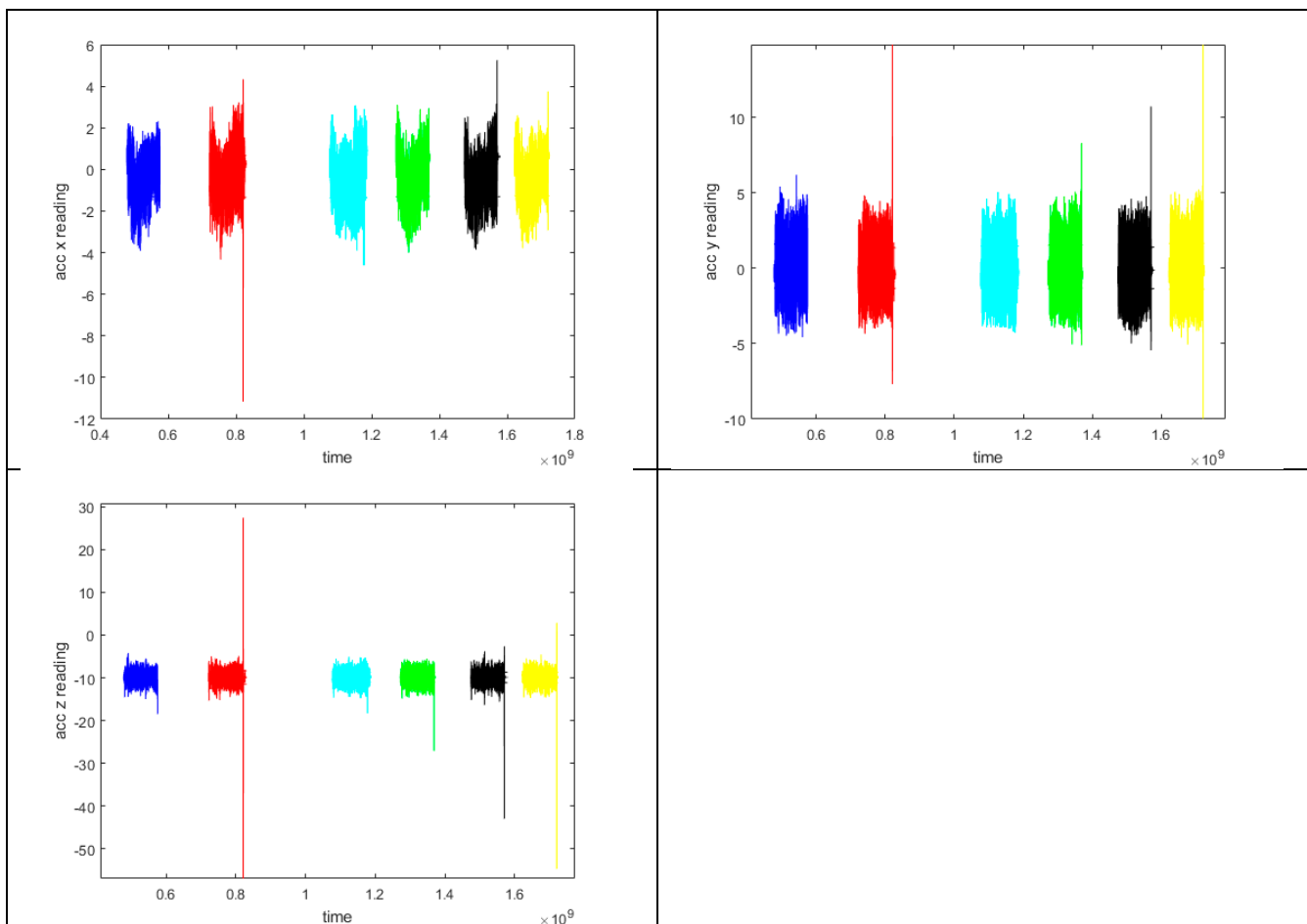


Figure 7: Accelerometer data for all flights

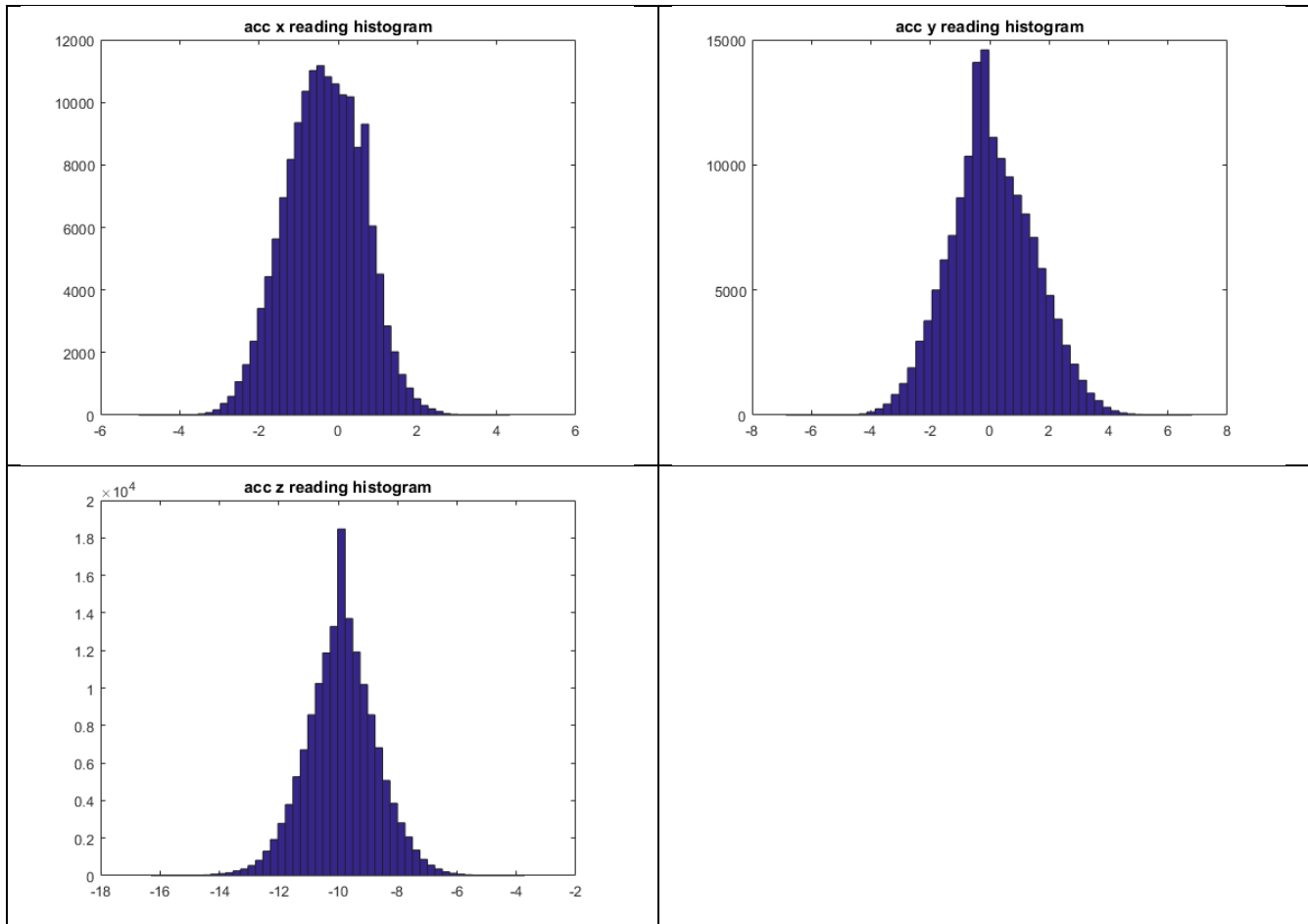


Figure 8: accelerometer data histograms

Similar analysis can be done for the other sensors (magnetometer, barometer, range finder, etc.), not presented here to save space.

D. Data filtering evaluation

Finally, other aspect related with sensor data is the availability in Px4 of some complementary filters that decrease the ambient noise by applying a low pass filter that gets only those data with an oscillation frequency that underneath the cutoff barrier and discard all data with a higher frequency. This approach works based in the principle that the vehicle movements that we wish to register, never surpass a certain speed, and those that overtake it can be directly considered as environment noise and vibrations. The use of complementary filters may increase the accuracy of the estimation and the efficiency of the system by reducing the amount data that the other algorithms like the EKF and the INS would have to deal.

Due to this relevance, during the configuration and calibrating process of our vehicles, we have performed an analysis of the max motor vibration frequency and those perturbations that travel through the frame of the multicopter up to the flight controller sensors. After this analysis we have tested the performance with different cutoff values to optimize the adjustment of the Low Pass Filter and ensure not to acquire useless data.

Figure 9 shows the band of dominant frequencies collected by the gyroscope sensor during an real flight. At lower frequencies we can observe the vehicle intentioned movements with frequencies that oscillate between [0-2] Hz, then with values that reach 90Hz are the movements produced by the fast speed corrections that are made on each motor to maintain the stability of the vehicle. The higher values than this frequency performs the vibration noise, introduced

by the small unbalances of the motors, the propeller turbulence shocking with the chassis and a phenomenon of resonance between propellers denominated propwash.

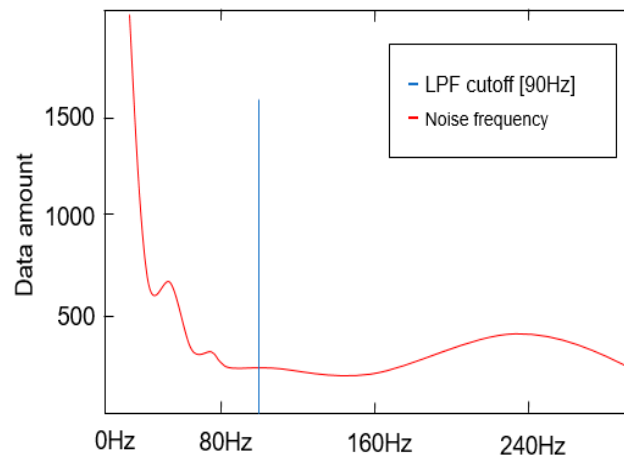


Figure 9: Gyroscope data frequency analysis
(y-axis is amount of data)

To reduce the effect of the environment noise, the Px4 system counts with some Extended Kalman Filter algorithm that process all readed data using estimation states that increases the precision of the data thanks to the application of specific corrections on each measure based in known characteristic error of the sensors.

V. EXPERIMENTAL EVALUATION OF TRACKING FILTERS FOR SENSOR FUSION

A. GPS and INS local position integration:

The most typical task carried out by the data fusion process of the Pixhawk (EKF2 filter) is the attitude estimation using magnetometer, gyroscope and accelerometer data (attitude and heading reference system), and then fuse with accelerometers and GPS data to estimate position and velocity. Fig. 10 a) shows an illustrative example of position estimation on the tangent plane XY (North-East) using the integrated measurements of accelerometers. We can see how the attitude changes in a superposition of the X and Y projections, especially on the last values.

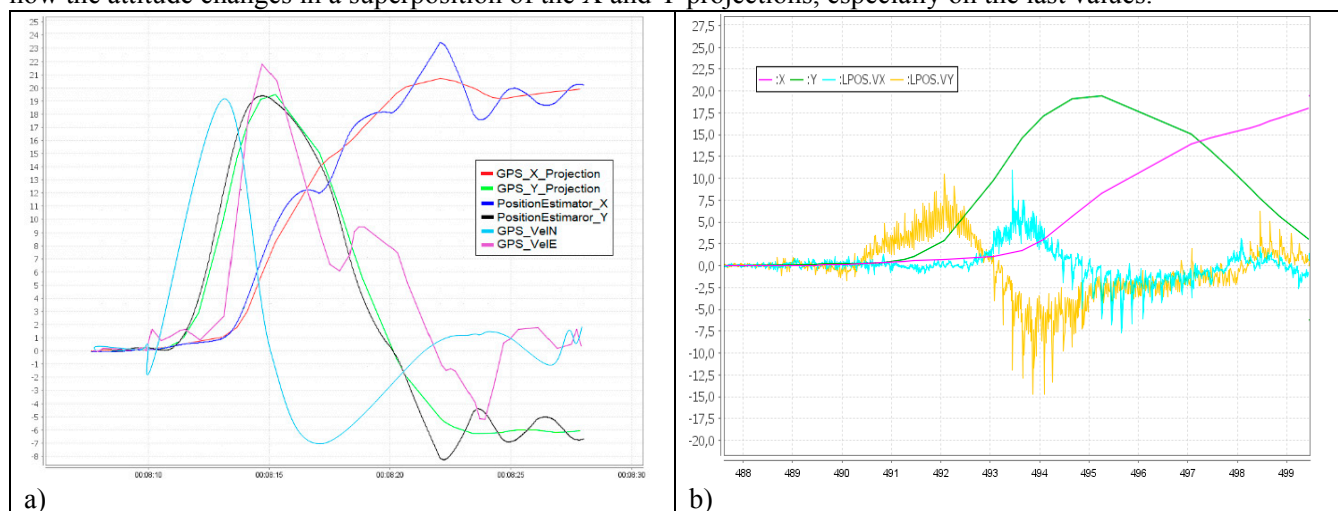


Figure 10: Attitude estimation through GPS, INS and magnetometer data fusion

As shown on Fig. 10 b), the INS computes local position by integration of accelerometer and gyroscope data expressed into XY plane, with a faster time update.

So, the sensor fusion system is based on a loosely coupled architecture which uses GPS position and velocity measurements to aid the INS, typically used in navigation solutions [18]. In this way, the IMU sensors are used extrapolate position, velocity, and attitude at high frequency (50 Hz), while updates from GPS measurements at low frequency (1 Hz) allows refinement of cinematic estimates and inertial sensor biases. Typically, the estimated state vector resulting in the output for the GNSS/INS filter contains the attitude vector represented with a quaternion, 3D position and velocity, three gyro biases and three accelerometer biases. The selected coordinate frame for position and velocity is the ENU frame (East, North, Down) with respect to the tangent plan with origin defined by the arming point in the start of the mission

Both the estimation of cinematic parameters and sensor biases critically depend on the filter process noise, whose parameters affect to the predicted error covariance and impact in the weights given to the observations with respect to the predicted estimates. A higher value for these parameters imply higher values of predicted covariance and so higher gain to observations (since the confidence on prediction decreases). Conversely, lower values imply lower gain to observations (higher confidence on predictions). The values have been systematically analyzed in the six scenarios used, with flights repeating the programmed mission (waypoints), but changing the parameters affecting to EKF performance shown in table VI.

TABLE VI
PROCESS NOISE PARAMETERS OF EKF

Name	Description	Flight1	Flight2	Flight3	Flight5	Flight6	Flight6
EKF2_ACC_B_NOISE	Process noise for IMU accel. bias prediction	0.003	0.001	0.003	0.007	0.01	0.003
		m/s^3	m/s^3	m/s^3	m/s^3	m/s^3	m/s^3
EKF2_GYR_B_NOISE	Process noise for IMU gyro bias prediction	0.001	0.001	0.003	0.007	0.01	0.001
		rad/s^2	rad/s^2	rad/s^2	rad/s^2	rad/s^2	rad/s^2
EKF2_ACC_NOISE	Accelerometer noise for covariance prediction	0.35	0.1	0.3	0.7	1.0	0.35
		$m/s/s$	$m/s/s$	$m/s/s$	$m/s/s$	$m/s/s$	$m/s/s$
EKF2_GYR_NOISE	Rate gyro noise for covariance prediction	0.015	0.01	0.03	0.07	0.1	0.015
		rad/s	rad/s	rad/s	rad/s	rad/s	rad/s

These parameters are complemented with those used to characterize the measurement noise. The default values for sensor noise were used in all flights, as presented in table VII

TABLE VII
SENSOR MEASUREMENT PARAMETERS OF EKF

Name	Description	Values
EKF2_BARO_NOISE	barometric altitude	2.0m
EKF2_GPS_P_NOISE	gps position	0.5m
EKF2_GPS_V_NOISE	gps horizontal velocity	0.5 m/s
EKF2_MAG_NOISE	magnetometer axis	3- 5.0e-2 Gauss
LPE_ACC_XY	Accelerometer noise density xy	0.012 m/s ² /sqrt(Hz)
LPE_ACC_Z	Accelerometer noise density z	0.02 m/s ² /sqrt(Hz)

Finally, the sensor fusion logic includes protection values for self-assessment of performance, as mentioned in section 3, and the sensors will be removed from fusion each time the threshold value for residuals are exceeded, the default value is 5 times the standard deviation, as shown in table VIII.

TABLE VIII
DATA CHARACTERIZATION

Name	Description	Values
EKF2_GPS_P_GATE	Gate size for GPS horizontal position fusion	5.0SD
EKF2_GPS_V_GATE	Gate size for GPS velocity fusion	5.0 SD
EKF2_BARO_GATE	Gate size for barometric and GPS height fusion	5.0 SD
EKF2_RNG_GATE	Gate size for range finder fusion	5.0 SD
EKF2_EV_GATE	Gate size for	5.0 SD

The analysis was carried out by postprocessing the data collected from six flight tests using the described UAV system and operated autonomously (besides, the UAV was monitored remotely by a pilot in visual line of sight for safety reasons). Figure 11 present the horizontal position estimated by tracking filter and GPS observations (circles). As can be appreciated in the zoomed figure 12, the flights corresponding to higher values of parameters (like flight 1, in blue, flight 5, in black) present lower deviations during turns, and, conversely, are affected more by the GPS noise.

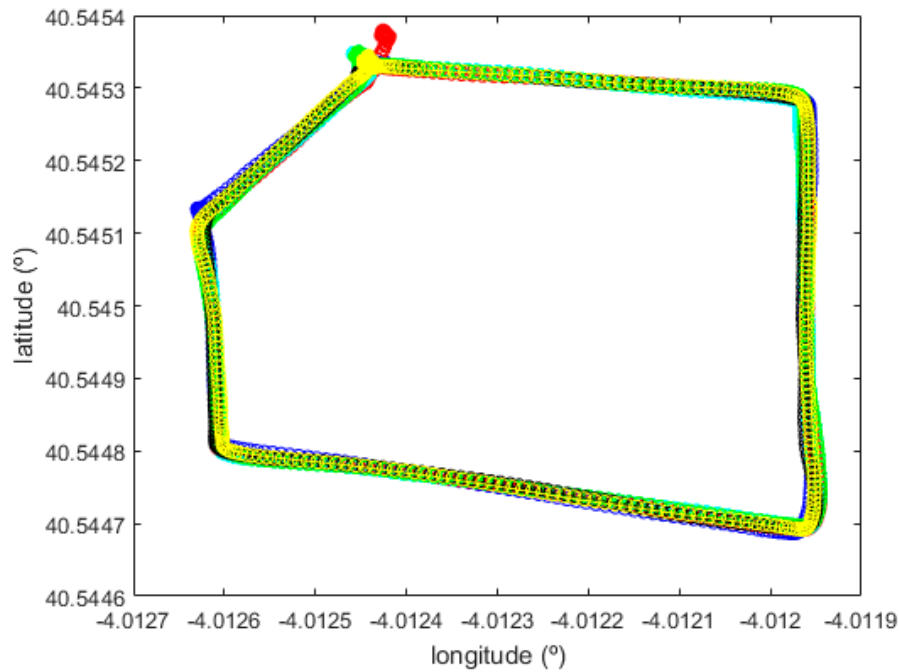


Figure 11: Lat-lon GPS input and EKF output for all flights (1-blue, 2-red, 3-cyan, 4-green, 5-black, 6-yellow)

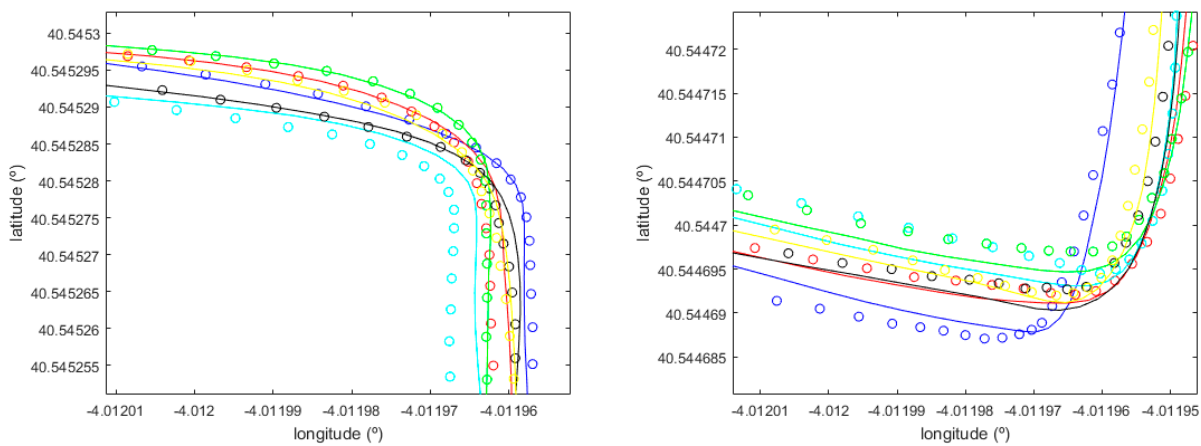


Figure 12: Details of lat-lon GPS input and EKF output for all flights (1-blue, 2-red, 3-cyan, 4-green, 5-black, 6-yellow)

The same effects can be observed with the GPS velocity observations. Figures 13 and 14 present the observations and filtered values, with zoomed graphics for flights 2,6 to appreciate the difference response with respect to observations.

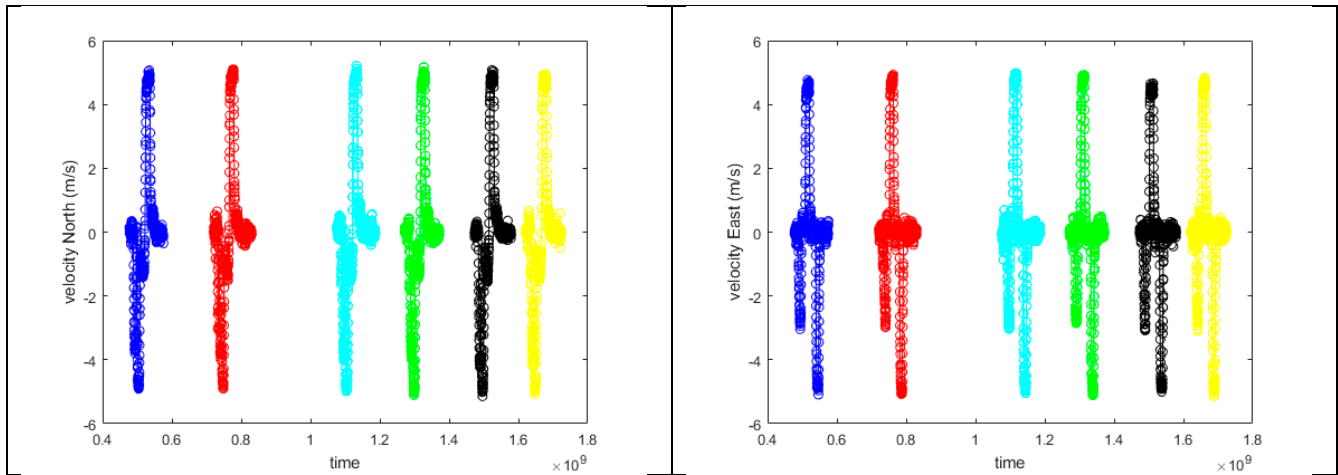


Figure 13: Horizontal GPS velocity input and EKF output for all flights (1-blue, 2-red, 3-cyan, 4-green, 5-black, 6-yellow)

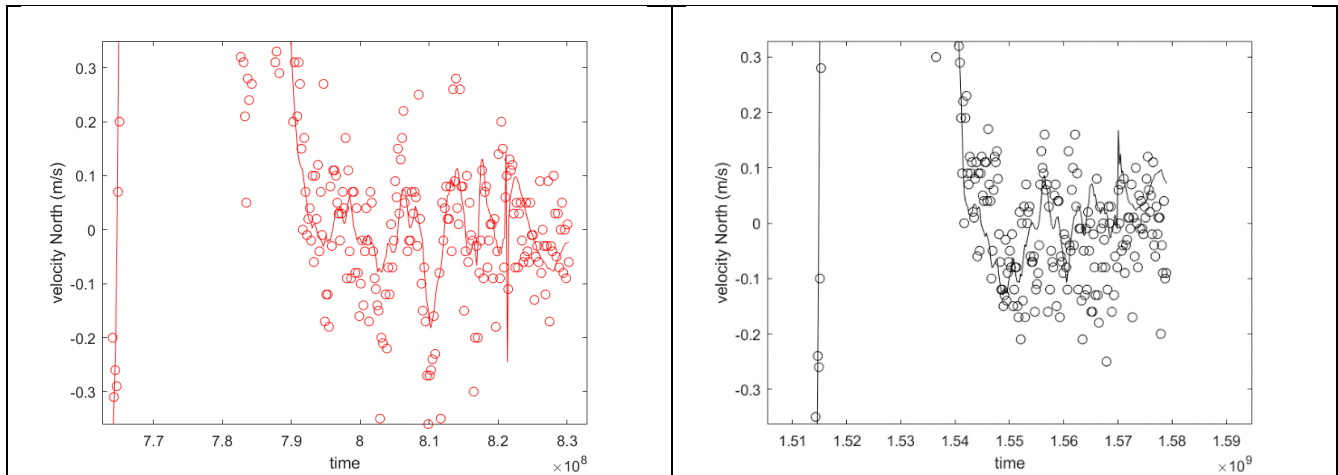


Figure 14: Details of lat-lon GPS input and EKF output for two flights (2-red, 5-black)

B. EKF vertical filter performance

Regarding vertical estimation, it can be obtained from different sensors, like barometer, GPS and range finders. Figure 15 a) shows an example of values estimated GPS and barometer. Notice that, the GPS value describes a discrete scale with high precision but low resolution and a large latency. On the other side, the barometer measurement allows higher frequency data but requires a calibration with respect to the typical atmospheric pressure. The result of the fusion process throws an estimated value with the accuracy benefits of both measurements, allowing a consistent altitude.

Regarding the use of range finders for height estimation, illustrative data in a flight test are presented in Fig. 15 b). As can be seen, a typical problem with these sensors are the short range of data or the errors introduced when the vehicle changes its pitch angle during flight. For this reason, the height calculation is made using the data from the barometer and the GPS if these sensors pass quality checks, and the measurement of the ground sensors are used only during landing and take-off procedures.

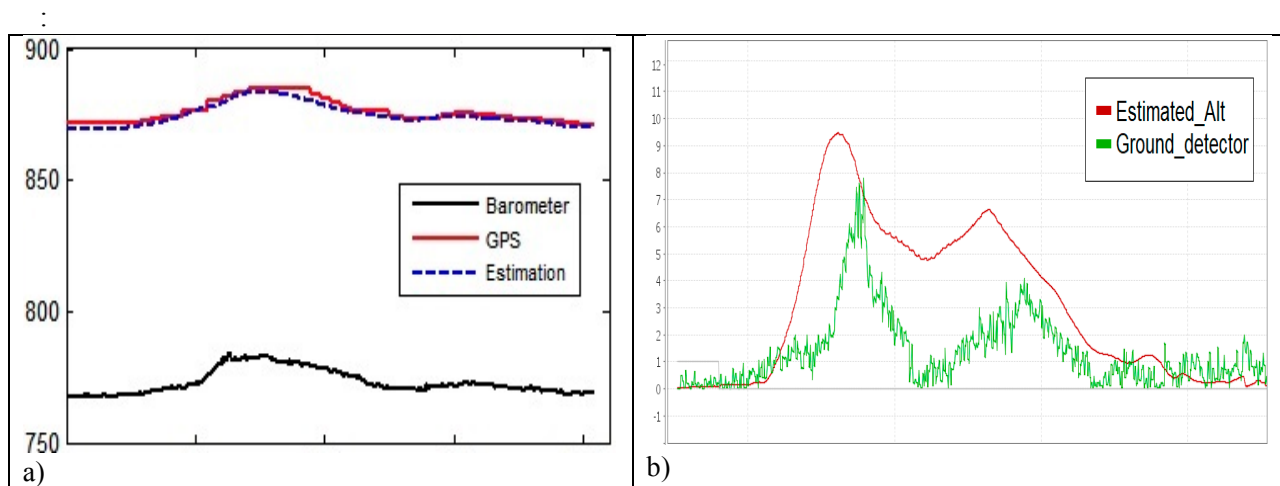


Figure 15: EKF vertical filtering using a) GPS and barometer fusion, b) barometer and range finder fusion

For the flights analyzed in this section, figure 16 presents the GPS and barometer data input and the estimated output. As can be seen in the details in Fig. 17, the weights given to barometer data are higher than GPS, with a smoothing of noise

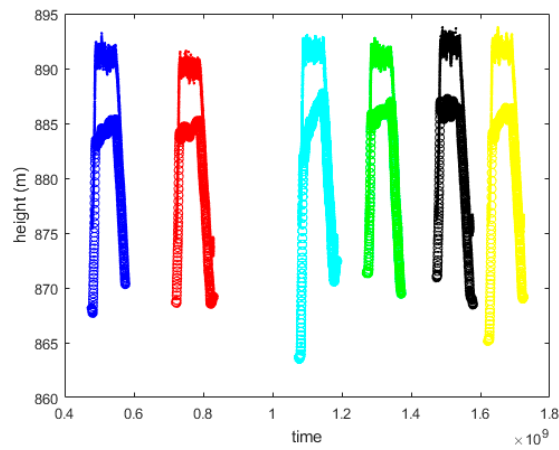


Figure 16: Barometric and GPS height inputs and EKF output for all flights

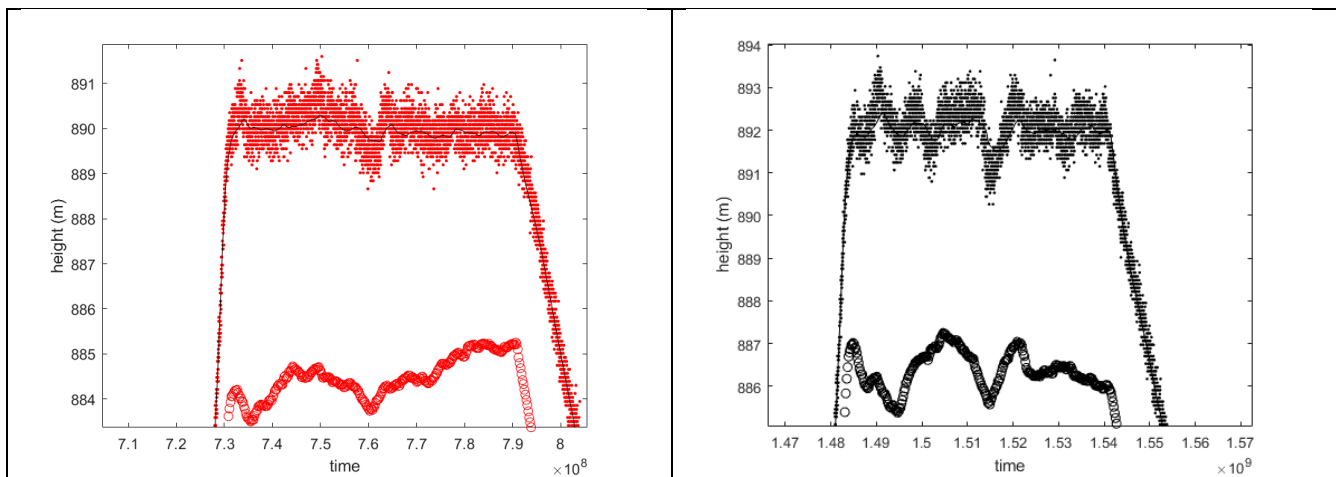


Figure 17: Estimated Details of Barometric and GPS height inputs and EKF output for flights 2-red, 5-black

C. Analysis of innovations

The normalized innovations, as presented in section 3, are presented in this section. The 3D position innovations are presented in Fig. 18, and their distributions are shown in boxplots in Fig. 19. As can be seen, a slight reduction of values appears in configurations of flights 2 and 4. Similar comments can be appreciated in the velocity innovations, presented in Fig. 20, with boxplots in Fig. 21

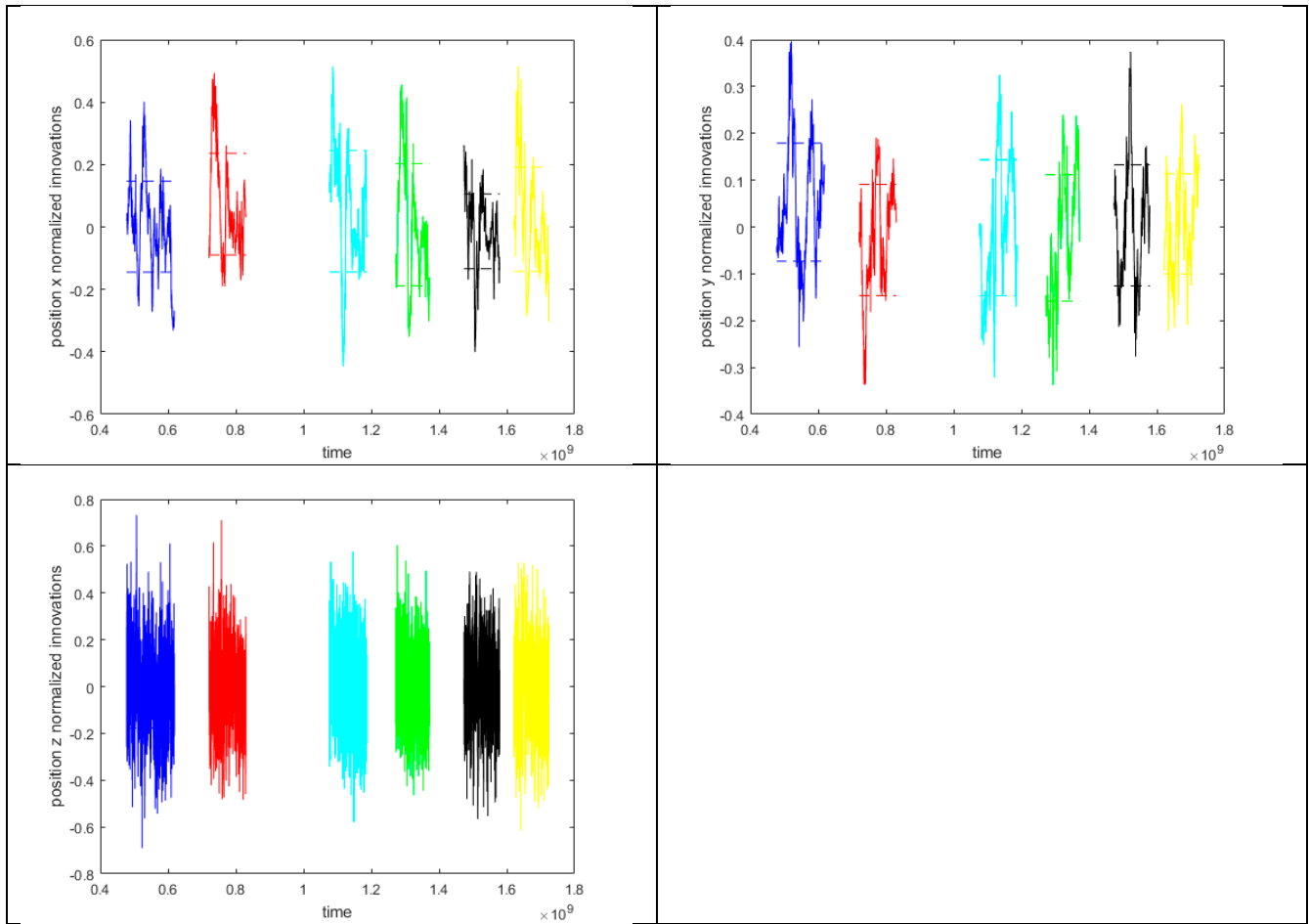
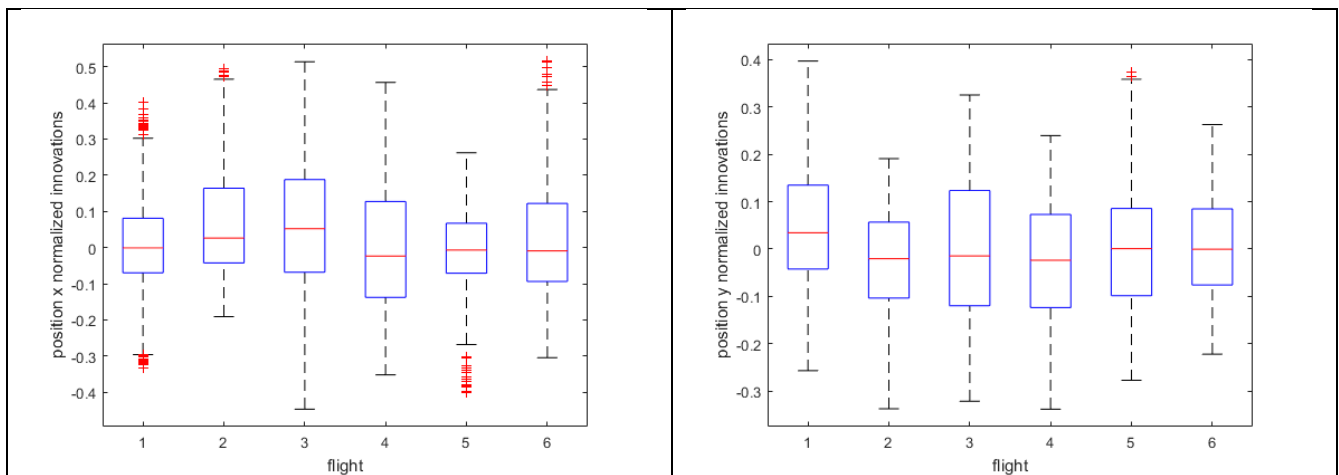


Figure 18: Innovations in position for all flights



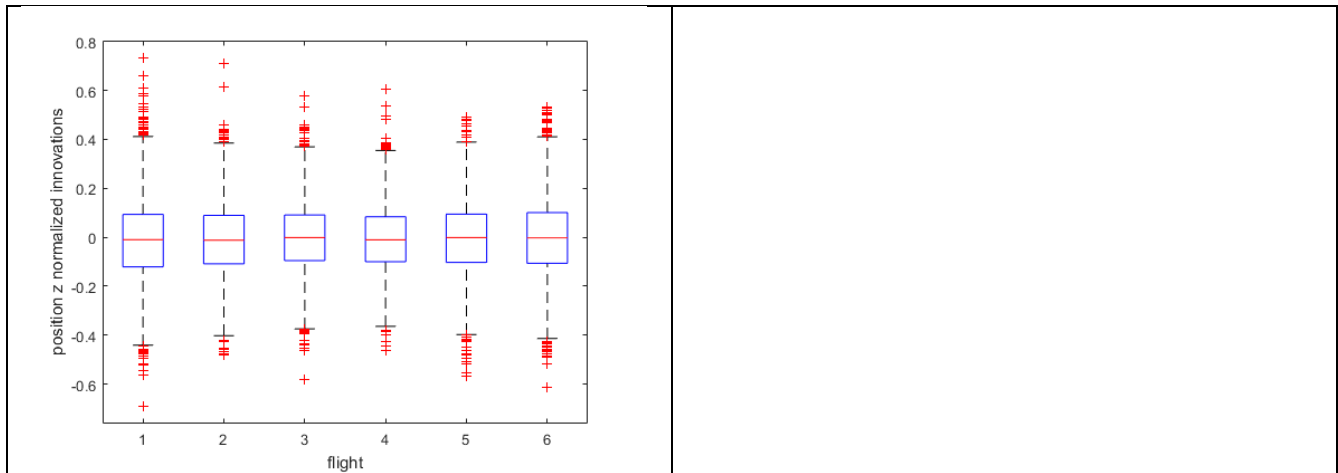


Figure 19: boxplots for position innovations in all flights

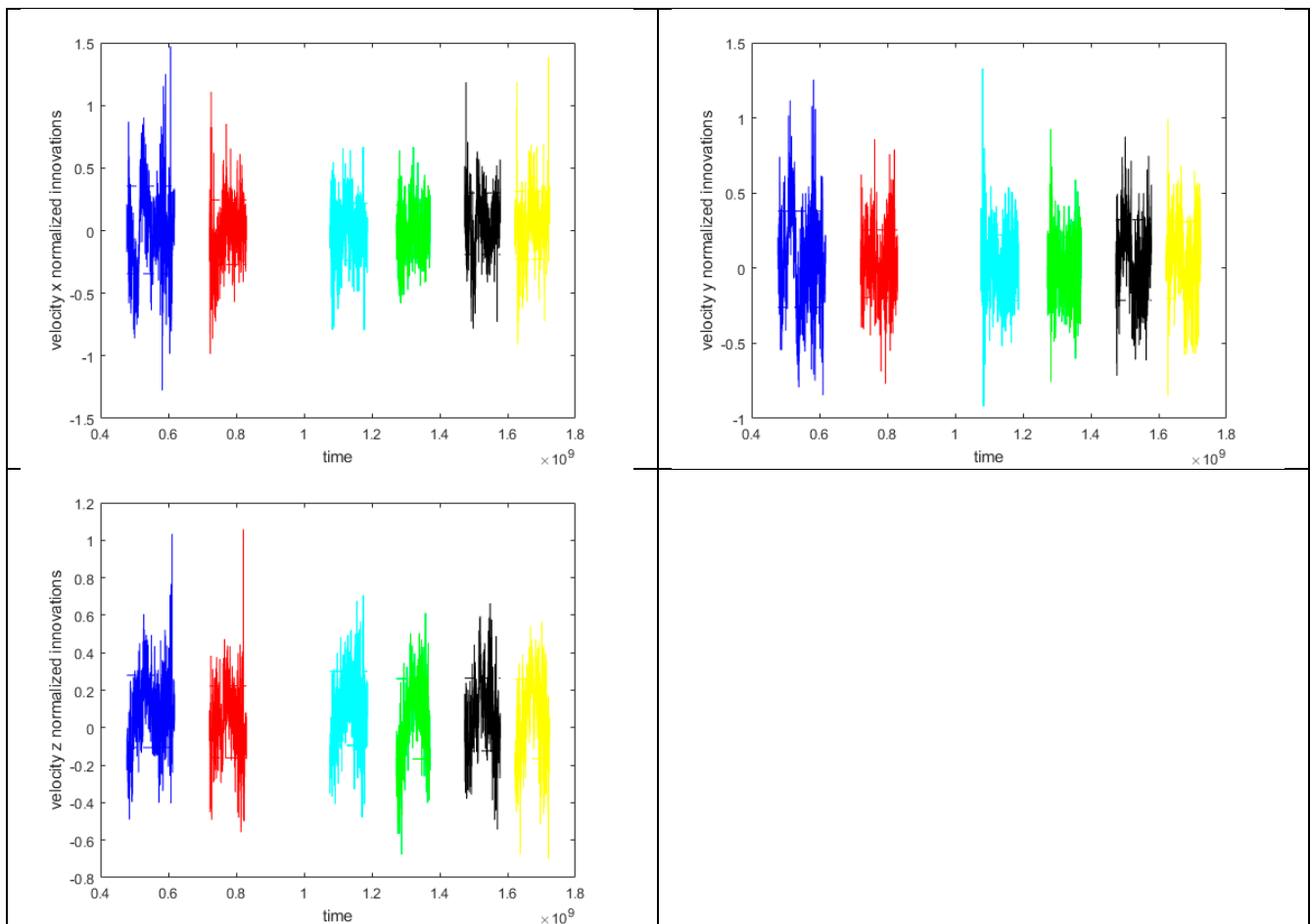


Figure 20: innovations in velocity for all flights

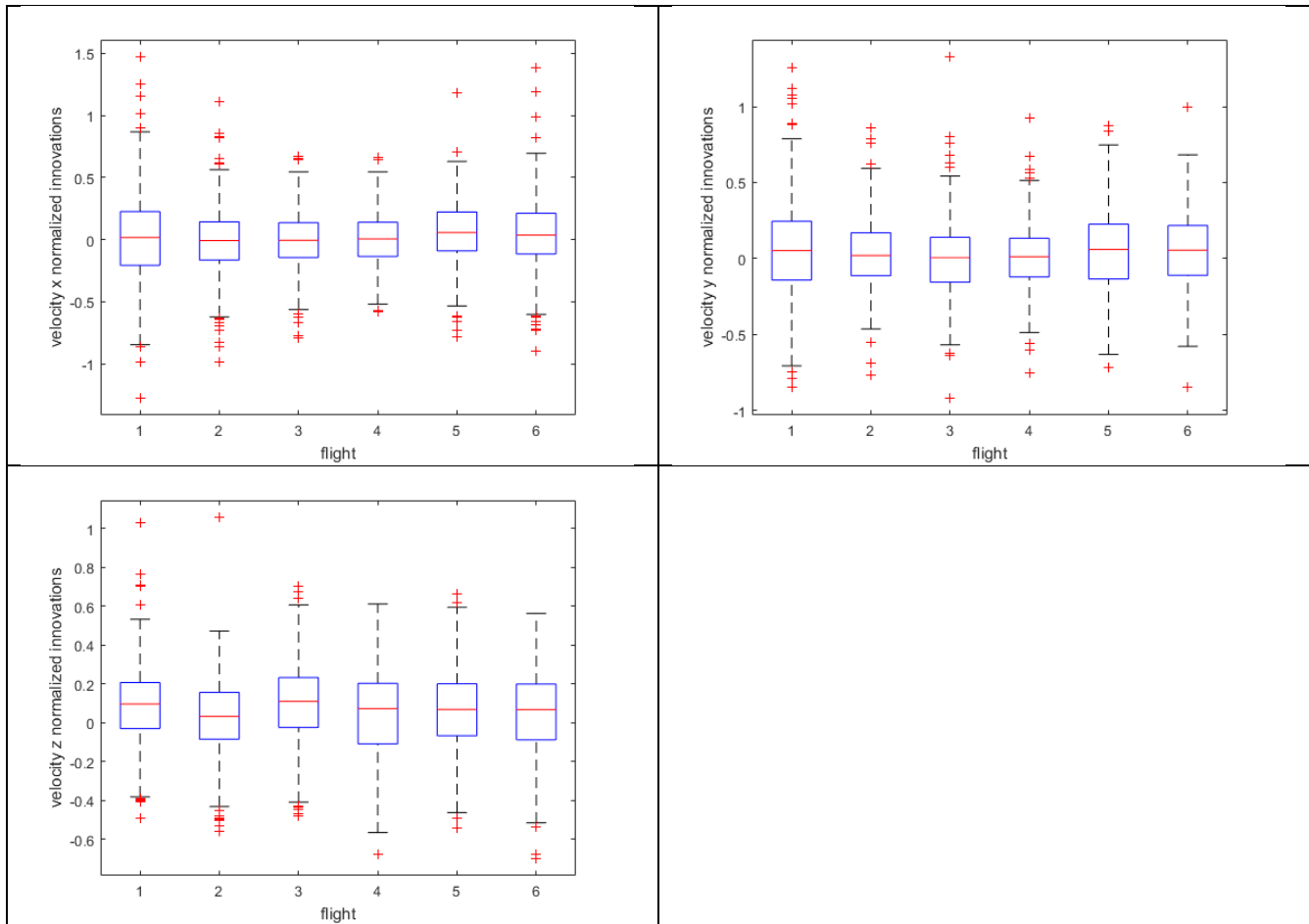


Figure 21: boxplots for velocity innovations in all flights

Finally, Figure 22 presents aggregated position and velocity in a 6D innovation vector, normalized by its covariance matrix. In this case can be seen that flights 4,5 present the lower values.

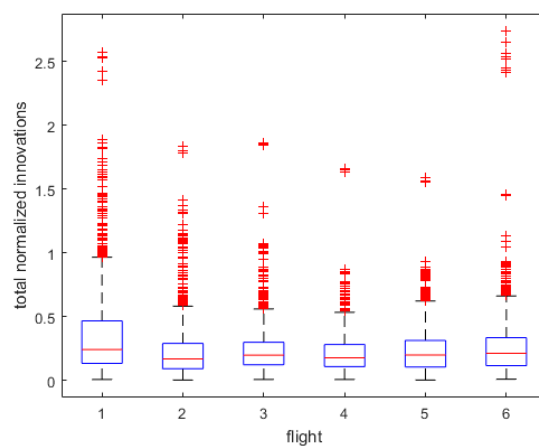


Figure 22: normalized innovations for all flights

Table IX summarizes the innovation analysis for the 6 flights

TABLE IX
RMS OF INNOVATIONS

Variable	Flight1	Flight2	Flight3	Flight4	Flight5	Flight6
pos_x	0.1456	0.1786	0.2011	0.1958	0.1205	0.1684
pos_y	0.1366	0.1220	0.1451	0.1368	0.1294	0.1070
pos_z	0.1674	0.1468	0.1452	0.1414	0.1527	0.1651
vel_x	0.3495	0.2583	0.2280	0.2059	0.2517	0.2746
vel_y	0.3250	0.2276	0.2187	0.2025	0.2732	0.2610
vel_z	0.2115	0.1940	0.2226	0.2192	0.2060	0.2172
aggregated pos-vel	0.4574	0.3040	0.2878	0.2532	0.2926	0.3456

D. Fusion breaks

Finally, the fusion logic did not have any fusion break with the different filter parameters tested. The values of tests carried out (innovation divided by five times the standard deviation) are displayed in Figure 23.

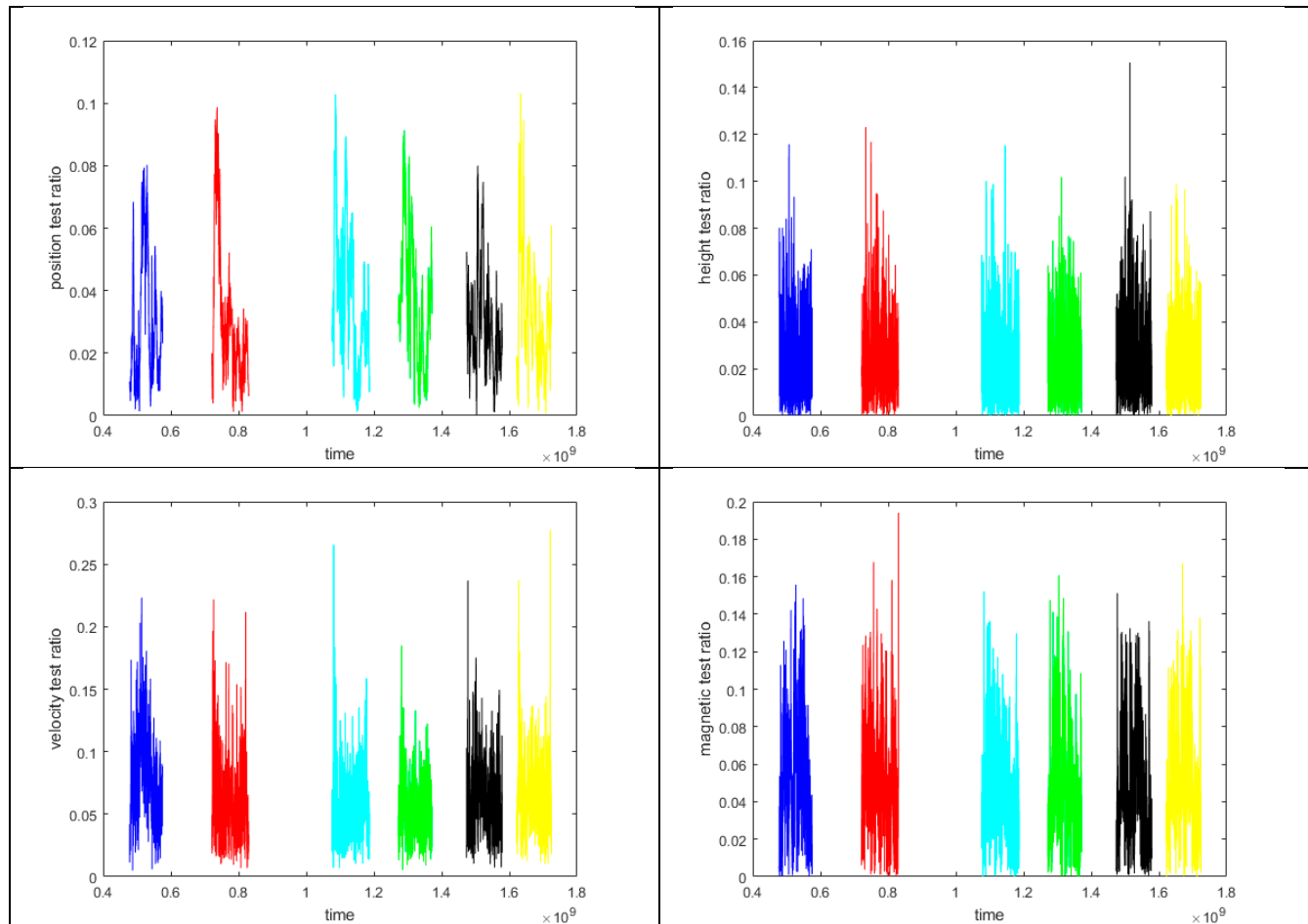


Figure 23: fusion break tests for all flights (horizontal position, height, velocity, attitude)

VI. CONCLUSIONS

This paper presented a platform (Pixhawk PX4) and methodology to experiment with real data for UAV navigation. Based on data analysis and characterization the algorithms can take advantage of available sources. The quality of all inputs was systematically analyzed, and three processing algorithms, DCM, LPF and EKF, were evaluated with different parameters to exploit the data in the appropriate way considering the output analysis and

specific performance metrics not based on ground truth.

So, this work presents a methodology to test and configure UAV navigation systems in real conditions, illustrated with an open environment for experimentation. The analysis of real data in a systematic way will allow successive improvements and parametrization, considering, among others, the following aspects:

- Data filtering to reduce perturbations and remove outliers
- Quality analysis to weight data uncertainty
- Analysis of biases and calibration previous to fusion
- Parameter adjustment to optimize performance (PID gains, filter parameters, observation and plant noises, etc.)

VII. ACKNOWLEDGMENTS

This work was funded by public research projects of Spanish Ministry of Economy and Competitiveness (MINECO), references TEC2014-57022-C2-2-R, RTC-2016-5595-2, RTC-2016-5191-8 and RTC-2016-5059-8

REFERENCES

- [1] Groves P.D. "Navigation using inertial sensors". IEEE AES Magazine (Vol.30, Iss. 2) 42 - 69 Feb. 2015
- [2] Britting, K.R. "Inertial navigation systems analysis". Artech House, 2010
- [3] Farrel J.A., "Aided Navigation: GPS with High Rate Sensors", McGraw-Hill, New York, 2008.
- [4] Choi H., Kim Y.. "UAV guidance using a monocular-vision sensor for aerial target tracking". Control Engineering Practice, 22, 10-19, 2014.
- [5] Torres-González A., Martínez-de Dios J.R., Ollero A. "Robot-Beacon "Distributed Range-Only SLAM for Resource-Constrained Operation". Sensors 2017, 17(4), 903.
- [6] Ferrick A., Fish J., Venator E. and Lee G.S. "UAV Obstacle Avoidance Using Image Processing Techniques Technologies for Practical Robot Applications (TePRA)", 2012 IEEE International Conference on 23-24 April 2012
- [7] Fasano G., Accado D., Moccia A., Moroney D., "Sense and avoid for unmanned aircraft systems". IEEE Aerospace and Electronic Systems Magazine, vol. 31, n° 11, pp. 82-110, 2016.
- [8] Rankin G., Tirkel A., Leukhin A. "Millimeter Wave Array for UAV Imaging". MIMO Radar Radar Symposium (IRS), 2015 16th International 24-26 June 2015
- [9] Broumandan A., Siddakatte R, Lachapelle G. "An Approach to Detect GNSS Spoofing". IEEE AESS Magazine Vol. 32, N8, August 2017. 64-75
- [10] Xiao L., "GNSS Receiver Anti-spoofing Techniques: A Review and Future Prospects". 5th Electronics and Network conference (CECNet 2015) Shanghai, 2015.
- [11] Carson N., Martin S.M., Starling J., and Bevely D.M. "GPS Spoofing Detection and Mitigation Using Cooperative Adaptive Cruise Control System". 2016 IEEE Intelligent Vehicles Symposium (IV) Gothenburg, Sweden, June 19-22, 2016. 1091-1096
- [12] Babister, A. W. (1980). "Aircraft dynamic stability and response" (1st ed.). Oxford: Pergamon Press. ISBN 978-0080247687.
- [13] Hyyti H. and Visala A., "A DCM Based Attitude Estimation Algorithm for Low-Cost MEMS IMUs". International Journal of Navigation and Observation, Volume 2015, Article ID 503814.
- [14] Kim J., Cheng J., Guivant J., Nieto J., "A. Compressed Fusion of GNSS and Inertial Navigation With Simultaneous Localization and Mapping". IEEE AESS Magazine Vol. 32, N8, August 2017. 22-36
- [15] Martí E., García J., Molina J.M. "A Simulation Framework for UAV Sensor Fusion". Hybrid Artificial Intelligence Systems 5th International Conference, HAIS 2010, San Sebastián, Spain, June 23-25, 2010
- [16] Layh T., Demoz G.E. "Design for Graceful Degradation and Recovery from GNSS Interruptions", IEEE AESS Magazine Vol. 32, N9, September 2017. 4-17
- [17] Meier L., PX4 Development Guide. [Online]. Available: <https://dev.px4.io/en/>
- [18] Martí E., García J., Escalera A., Molina J.M., Armingol J.M. "Context-Aided Sensor Fusion for Enhanced Urban Navigation". Sensors Vol. 2012 16802-16837. 2012