Implementation of a Real-Time Attitude System for the Stabilization Control of a Quad-rotor Robot

Sérgio Ronaldo Barros dos Santos*, Cairo Lúcio Nascimento Júnior*, Sidney N. Givigi Jr †,
Tiago Henrique Medeiros Mercante*, Neusa Maria Franco de Oliveira*.
* Division of Electronic Engineering
Instituto Tecnológico de Aeronáutica
São José dos Campos, Brazil.
Email: sronaldo@ita.br, cairo@ita.br, tiagoh@ita.br,neusa@ita.br
† Department of Electrical and Computer Engineering
Royal Military College of Canada
Kingston, ON. Canada.
Email: sidney.givigi@rmc.ca

***Abstract*—The real-time estimation of the attitude and heading is a fundamental task for an autonomous vehicle to operate properly. Nowadays, the determination of this information using low cost sensors is a very active research area in robotics applications. Therefore, a new method in sensory fusion to attitude estimation using low cost devices is addressed in this paper. The Attitude and Heading Reference System (AHRS) consists of three-axes solid-state or MEMS gyroscopes, accelerometers and magnetometers and also an on-board processing system. The algorithm uses direction cosine matrix theory to obtain the desired information from the low cost sensors. Using the computed attitude, a simple approach to the stabilization of a quad-rotor robot while in flight was implemented. Towards the stabilization of the quad-rotor, AHRS must be used as a feedback for the PD controllers. Experimental results showed that the AHRS and PD controllers implemented can be used to stabilize the quad-rotor during flight. The responses of the system can be visualized in graphics plotted in Matlab or through a virtual graphic interface developed.**

Index Terms - Attitude Controllers; IMU; AHRS; UAV.

I. INTRODUCTION

Nowadays, there are a multitude of types of sensors that can be used to measure the attitude of an aircraft. The most common ones are inertial sensors, infrared sensors and optical sensors. These three types of sensors are based on different principles and have their special features. All of them can be implemented on autonomous vehicles, such as terrestrial, maritime or aerial, as the primary navigation sensors [1]. Inertial sensors can provide very accurate attitude measurements, but they have the most complicated configurations. An inertial sensor system is usually called inertial measurement unit (IMU). An IMU usually contains a module with accelerometers, gyroscopes and sometimes other components, such as magnetometer. The accelerometers are used to measure the linear acceleration of the platform and the gyroscopes are used to measure the angular velocity. With those measurements, the orientation of a vehicle can be estimated in the inertial reference frame. Due to their accurate navigation performance, IMUs have been broadly applied on unmanned aerial vehicles (UAV), ground robots and in others types of vehicles [1 and 2].

Attitude is the term used to describe the vehicle's orientation in space. The Attitude and Heading Reference System (AHRS) is a device used to determine a vehicle's attitude and heading [3]. The attitude information generated by an AHRS is used in many navigation, guidance, and control applications such as pilot-in-the-loop control of manned aircrafts. The key difference between an IMU and an AHRS is the addition of an on-board processing system. In an AHRS the attitude and heading angles are directly provided, while an IMU just delivers sensor data to an additional device that computes the attitude angles [4]. In order to control a quad-rotor correctly, the attitude is one of the inputs used by the low-level control system. Then, the developed system may be used to resolve the stabilization control problem for a quad-rotor. Of course, an AHRS can be applied to vehicles with less degrees-of-freedom.

Our goal is to implement an Attitude and Heading Reference System using low-cost inertial and magnetic sensors that is able to provide accurate attitude information to stabilize the quad-rotor robot during flight. Using the proposed method the Euler angles are computed through the angular rates obtained from the gyroscope. Due to the fast update rate of the measured gyro data, drift and also integration errors are expected. So, the fusion with others sensors such as a magnetometer and an accelerometer is necessary. The experimental validation has been conducted using the AHRS to get the real-time attitude estimation of the quad-rotor. The attitude data are used to feedback all the controllers implemented in the embedded controller board. The control loop approach used in this experiment is based on PD control. The computed attitude angles are sent to the ground station by means of XBee RF modules. The reference signals for the attitude control loop are transmitted to the UAV through its own radio control using an update rate of 50Hz. This experimental environment allows not only testing the hardware and software but also conducting the flight tests with minimal cost and effort.

978-1-4673-3108-1/13/$31.00

The paper is organized as follows. In section II the problem addressed is described and the devices that will be used in the solution are presented. Section III is devoted to the presentation of the proposed algorithm for the attitude estimator system. Section IV presents the implementation of AHRS and the visualization setup. In section V the experimental setup and obtained results using the implemented AHRS and control system are presented. Our conclusions are found in section VI.

II. Problem Statement

Quad-rotor robots have become popular due to their relatively simple model and the low-cost involved in operating experimental platforms. A quad-rotor is an underactuated aircraft with four rotors using fixed pitch angle [5]. However, in spite of the four actuators, the quad-rotor is a dynamically unstable system that has to be stabilized through a suitable attitude and heading measurements and also a control system [6]. Fig. 1 shows the system architecture of quad-rotor.

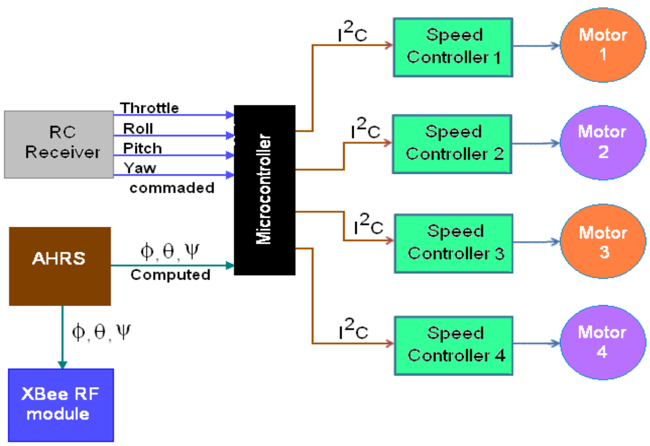

Figure 1. Overview of system architecture of the quad-rotor.

The RC module is the receiver device from standard RC transmitter. Using this device the command signal for the attitude loop control implemented in the on-board microcontroller are received. The computed attitude by means of developed AHRS is used for closed loop feedback. The microcontroller produces four I^2C outputs to control the brushless motor speeds. These command signals are sent to brushless controllers using a 1kHz update rate [6]. All of the movements of quad-rotor can be controlled by the changes of each rotor speed that results in different thrust and angular motions.

A significant challenge in developing UAV is to extract and fuse the useful information in a robust manner to provide stable flight [7]. In this paper, we address the problem of attitude estimation for the stabilization control of the quad-rotor by the fact that the fulfillment of this task is a precondition for further implementation of other functionalities. For a better understanding of the addressed problem, it is necessary to provide some details about the system to be developed and the intrinsic difficulties of the problem. Low cost sensors do not commonly provide good information relative to the data measured, due to their high noise levels and signal drift. Thus, these sensors have their filtered signals and also are usually used together with other devices before being applied to autonomous vehicles. In this approach, attitude and heading are estimated using the magnetic force, angular rate, and gravity measurements through the three-axes magnetic sensors, three-axes gyro sensors, and three-axes accelerometers sensors, respectively [8].

During the estimation process, the measured angular rates are integrated using a fixed update rate. However, this method produces small errors which are accumulated over time. To reduce the error, the integrated signal from the gyroscope will be combined with the measured magnetic force and gravity through a fusion algorithm. Fig. 2 shows the fusion algorithm architecture for the attitude and heading estimation.

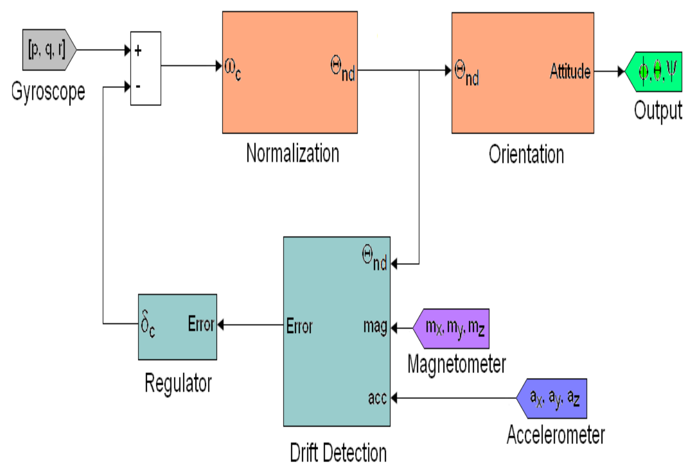

Figure 2. Architecture of the fusion algorithm for the AHRS

The presented algorithm can be described as a standard feedback loop. The measured angular rates are compared to the produced signal from the feedback loop, generating refined angular rate vectors. This approach uses discrete samples of the angular rates to obtain the attitude and heading estimation through an integration process. Thus, the attitude estimation depends not only on current measurement but also depends on the previous angular rates.

The reference signals are composed of the measured angular rates about the x, y and z axes of the body frame. The corrected angular rate is obtained by the subtracting between the feedback signal and the reference signal. Using the refined angular rate, the Euler angles (roll φ, pitch θ, and yaw ψ) can be computed through the process of enforcing the orthogonality conditions. The feedback loop uses the computed attitude, magnetic force and gravity vectors to model the drift error using the sensory fusion. The generated error signal is used to compute the adjustment vector. This adjustment vector is compared to the reference signal, generating the corrected angular rate as described previously.

III. Algorithm Description

Sensor fusion is a common method to increase the accuracy and remove noise in sensory systems. Usually, two orthogonal coordinate frames are used, the body frame denoted by superscript "b", and the inertial frame denoted by superscript "n".

The Direction Cosine Matrix (DCM), applied to transform a vector in the body frame to the inertial frame, with s and c used to abbreviate sin (.) and cos(.) functions, is shown in (1).

$$R_b^n = \begin{bmatrix} c(\theta)c(\psi) & -c(\phi)s(\psi)+s(\phi)s(\theta)c(\psi) & s(\phi)s(\psi)+c(\phi)s(\theta)c(\psi) \\ c(\theta)s(\psi) & c(\phi)c(\psi)+s(\phi)s(\theta)s(\psi) & -s(\phi)c(\psi)+c(\phi)s(\theta)s(\psi) \\ -s(\theta) & s(\phi)c(\theta) & c(\phi)c(\theta) \end{bmatrix} \quad (1)$$

From (1), assuming that the measured Euler angles are very small, the values of $\sin\phi$, $\sin\theta$, $\sin\psi$ is approximately equal to ϕ, θ, ψ, respectively, and $\cos\phi$, $\cos\theta$, $\cos\psi$ is nearly equal to one [9]. Then, the Direction Cosine Matrix can be rewrite as:

$$\Theta_b^n = R_b^n = \begin{bmatrix} 1 & -\psi & \theta \\ \psi & 1 & -\phi \\ -\theta & \phi & 1 \end{bmatrix} \quad (2)$$

where initially ϕ, θ, ψ are equal to zero.

In this approach the attitude is constantly updated through the body rotation rate. The rotation rate is represented by the $[p \;\; q \;\; r]^T$ vector measured around the x, y and z axis, respectively. Using the DCM, these values can be expressed as:

$$\Omega_b^n = \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix} \quad (3)$$

The current rotation rate can be computed, [9 and 10] using (4).

$$\dot{R}_b^n(k) = \Theta_b^n(k-1)\cdot\Omega_b^n(k) \quad (4)$$

Using (5), the rotation matrix can be calculated at discrete-time, every 0.020 seconds.

$$\Theta_b^n(k) = \Theta_b^n(k-1) + \dot{R}_b^n(k) \quad (5)$$

A. Normalization and Orientation

Usually, the numerical errors due to the integration process gradually reduce the orthogonality between the components of the rotation vectors around the x, y and z axis. Thus the body and inertial frame axes no more describe a rigid body. The correction of this error can be made through the process of enforcing the orthogonality conditions, as described in [9 and 10]. Consider the rotation matrix as given in (6).

$$\Theta(k) = \begin{bmatrix} R_{xx} & R_{xy} & R_{xz} \\ R_{yx} & R_{yy} & R_{yz} \\ R_{zx} & R_{zy} & R_{zz} \end{bmatrix} \quad (6)$$

Firstly, is computed the dot product between the x and y vectors of the rotation matrix. The results inform how much the x and y vectors lag each others [9 and 10].

$$E_{ort} = [R_{xx} \quad R_{xy} \quad R_{xz}] \bullet \begin{bmatrix} R_{yx} \\ R_{yy} \\ R_{yz} \end{bmatrix}^T \quad (7)$$

Using the orthogonality error, new x and y vectors are derived through the (8) and (9).

$$\begin{bmatrix} R_{xx} \\ R_{xy} \\ R_{xz} \end{bmatrix}_{ort}^T = \begin{bmatrix} R_{xx} \\ R_{xy} \\ R_{xz} \end{bmatrix}^T - \frac{E_{ort}}{2}\cdot\begin{bmatrix} R_{yx} \\ R_{yy} \\ R_{yz} \end{bmatrix}^T \quad (8)$$

$$\begin{bmatrix} R_{yx} \\ R_{yy} \\ R_{yz} \end{bmatrix}_{ort}^T = \begin{bmatrix} R_{yx} \\ R_{yy} \\ R_{yz} \end{bmatrix}^T - \frac{E_{ort}}{2}\cdot\begin{bmatrix} R_{xx} \\ R_{xy} \\ R_{xz} \end{bmatrix}^T \quad (9)$$

It can be verified that the orthogonality error is reduced, substituting (9) and (8) into (7), knowing that the magnitude among the vectors should be approximately unitary. In the next step, the z vector is also adjusted to be orthogonal to the x and y vectors, using (10).

$$\begin{bmatrix} R_{zx} \\ R_{zy} \\ R_{zz} \end{bmatrix}_{ort}^T = \begin{bmatrix} R_{xx} \\ R_{xy} \\ R_{xz} \end{bmatrix}_{ort}^T \times \begin{bmatrix} R_{yx} \\ R_{yy} \\ R_{yz} \end{bmatrix}_{ort}^T \quad (10)$$

Immediately after the updates, it should be assured that the x, y and z vectors are of unitary magnitude. One way to do that is using the Taylor's expansion for each vector as shown in (11), (12) and (13).

$$\begin{bmatrix} R_{xx} \\ R_{xy} \\ R_{xz} \end{bmatrix}_{nd}^T = \left(\frac{1}{2}\left(3-\left(\begin{bmatrix} R_{xx} \\ R_{xy} \\ R_{xz} \end{bmatrix}_{ort}^T \bullet \begin{bmatrix} R_{xx} \\ R_{xy} \\ R_{xz} \end{bmatrix}_{ort}^T\right)\right)\right)\cdot\begin{bmatrix} R_{xx} \\ R_{xy} \\ R_{xz} \end{bmatrix}_{ort}^T \quad (11)$$

$$\begin{bmatrix} R_{yx} \\ R_{yy} \\ R_{yz} \end{bmatrix}_{nd}^T = \left(\frac{1}{2}\left(3-\left(\begin{bmatrix} R_{yx} \\ R_{yy} \\ R_{yz} \end{bmatrix}_{ort}^T \bullet \begin{bmatrix} R_{yx} \\ R_{yy} \\ R_{yz} \end{bmatrix}_{ort}^T\right)\right)\right)\cdot\begin{bmatrix} R_{yx} \\ R_{yy} \\ R_{yz} \end{bmatrix}_{ort}^T \quad (12)$$

$$\begin{bmatrix} R_{zx} \\ R_{zy} \\ R_{zz} \end{bmatrix}_{nd}^T = \left(\frac{1}{2}\left(3-\left(\begin{bmatrix} R_{zx} \\ R_{zy} \\ R_{zz} \end{bmatrix}_{ort}^T \bullet \begin{bmatrix} R_{zx} \\ R_{zy} \\ R_{zz} \end{bmatrix}_{ort}^T\right)\right)\right)\cdot\begin{bmatrix} R_{zx} \\ R_{zy} \\ R_{zz} \end{bmatrix}_{ort}^T \quad (13)$$

After use of the method of enforcing the orthogonality conditions, the rotation matrix can be given as in (14).

$$\Theta_{nd}(k) = \begin{bmatrix} R_{xx} & R_{xy} & R_{xz} \\ R_{yx} & R_{yy} & R_{yz} \\ R_{zx} & R_{zy} & R_{zz} \end{bmatrix}_{nd} \tag{14}$$

Using $\Theta_{nd}(k)$, the attitude of the quad-rotor robot can be calculated by (15), (16) and (17). The Euler angles are computed every 0.020 seconds [9 and 10].

$$\theta = -\arctan\left(\frac{R_{zx}}{\sqrt{1-(R_{zx})^2}}\right) \tag{15}$$

$$\phi = \arctan 2((R_{zy}),(R_{zz})) \tag{16}$$

$$\psi = \arctan 2((R_{yx}),(R_{xx})) \tag{17}$$

B. Fusion

Although the gyros have a regular performance with an uncorrected offset of few degrees per second, the accuracy of the measured attitude using the AHRS can be improved through drift correction [10]. For this, it has been used the measured magnetic force and gravity from the magnetometer and accelerometer, respectively, to detect the gyro offset and provide a negative feedback loop, as shown in Fig. 2. The feedback loop performs the following steps at each iteration:

a) Uses the computed roll, pitch and yaw angles through the enforcing orthogonality process and also the measured gravity and magnetic force vectors to detect the gyro drift;

b) Computes an adjust vector using the total error vector and also the proportional-integral regulator for correction of the gyro drift; and

c) Adjusts the current angular rates obtained by the gyro using the computed adjust vector.

The measured gravity vector and also the magnetic force vector in the body frame can be written as:

$$a = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} \qquad m = \begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix} \tag{18}$$

Using the measured gravity vector, the components of rotation matrix related to roll and pitch angles are verified [9]. The error vector, E_a is calculated by (19).

$$\begin{bmatrix} E_{ax} \\ E_{ay} \\ E_{az} \end{bmatrix} = \begin{bmatrix} R_{zx} \\ R_{zy} \\ R_{zz} \end{bmatrix}_{nd}^T \times \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} \tag{19}$$

Then, the yaw orientation can be computed from the measured magnetic force $[m_x\ m_y\ m_z]^T$ and also the roll and pitch angles obtained previously, [11], using (20), (21) and (22).

$$M_1 = m_x \cos(\theta) + m_y sen(\phi) sen(\theta) + m_z \cos(\phi) sen(\theta) \tag{20}$$

$$M_2 = m_y \cos(\phi) - m_z sen(\phi) \tag{21}$$

$$\psi_m = \arctan 2(-M_2, M_1) \tag{22}$$

The computed yaw orientation calculated in (22) and the components of the rotation matrix related to roll, pitch and yaw angles are used to derive the error vector, E_m using:

$$\begin{bmatrix} E_{mx} \\ E_{my} \\ E_{mz} \end{bmatrix} = \begin{bmatrix} R_{zx} \\ R_{zy} \\ R_{zz} \end{bmatrix}_{nd}^T \cdot \left(sen(\psi_m) \cdot R_{xx} - \cos(\psi_m) \cdot R_{yx} \right) \tag{23}$$

The estimate of the drift defined by a total error vector can be obtained through the (24).

$$\begin{bmatrix} E_x \\ E_y \\ E_z \end{bmatrix} = \begin{bmatrix} E_{ax} \\ E_{ay} \\ E_{az} \end{bmatrix} + \begin{bmatrix} E_{mx} \\ E_{my} \\ E_{mz} \end{bmatrix} \tag{24}$$

Finally, the adjust vector applied for gyro drift correction is given by:

$$\delta_c(k) = Kp. \begin{bmatrix} E_x \\ E_y \\ E_z \end{bmatrix} + Ki \int_0^{0.02} \begin{bmatrix} E_x \\ E_y \\ E_z \end{bmatrix} dt \tag{25}$$

IV. Implementation

In order to control an autonomous vehicle, one of the inputs needed by the autopilot control system is attitude. Attitude is usually represented by 3 rotations, these rotations are roll, pitch and yaw. The AHRS is implemented as an embedded system with the following main components, one 8-bit ATmega as the main processor, two ST's gyroscopes LPR530AL and LY530ALH, one Analog Device's 3 axis accelerometer ADXL345, one Honeywell's 3 axis magnetometer HMC5843, and one XBee RF Pro 60mW wire antenna. Fig 3 shows the structure of the developed system.

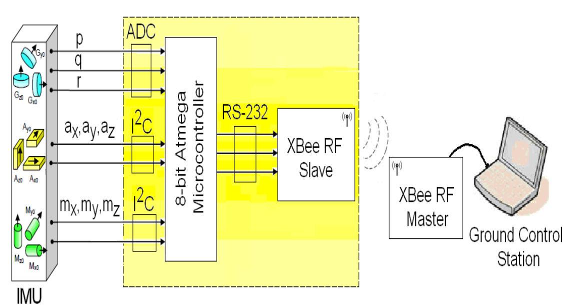

Figure 3. Architecture of the developed AHRS system.

The developed algorithm was implemented in C language, compiled and executed in an 8-bit Atmega microcontroller. Firstly, the microcontroller receives all of the sensors data. The three axis gyroscope sends its measurements in analog format, which is read by an analog digital converter channel in the microcontroller. The three axis accelerometer and magnetometer send their measurements for the microcontroller using I2C communication protocol.

The accelerometers and gyros are sampled in 50Hz and filtered using IIR butterworth 2 order low pass filter. The cut off frequency chosen is 5Hz. This low pass filtering is needed to avoid that the vibration caused by UAV's engine and propellers is measured. On the other hand, the magnetometer is read at a 10 Hz update rate. The attitude estimator is implemented using Euler angle representation. The Euler angles are updated using the gyro data $[p\ q\ r]^T$. The algorithm uses a predefined execution rate of 50Hz. This execution rate was chosen to be matched with the command frequency used to control the quad-rotor. Fig 4 shows the built AHRS board.

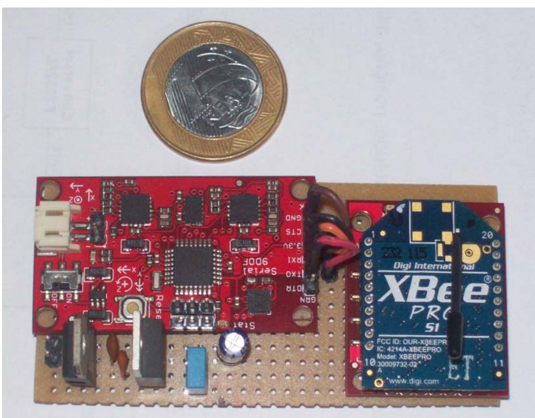

Figure 4. Embedded AHRS board built.

The computed attitude is sent to embedded controller board and also for a Ground Control Station (GCS) through the serial channel. The computed informations from the on-board sensors are used to feedback the control loops implemented in the embedded microcomputer board. On the other hand, the GCS receives all of the obtained information from the AHRS. The data link transmission between the AHRS and the GCS is established using XBee RF modules. This data can be visualized in graphics plotted in real-time into the Matlab.

Another way to visualize the computed body attitude is using the graphic interface developed in Python. Using this language, a virtual body was built to execute in real time the same rotational movements of the embedded AHRS board. Fig 5 shows the graphic interface used to test the algorithm developed.

Using the graphic interface, it is also possible to verify the importance of the correction feedback loop implemented. After few seconds of the loop being suppressed of the algorithm, the movements of the virtual body do not correspond anymore to the attitude of actual platform. In other words, the virtual body stops its movement while the platform continues to change its attitude.

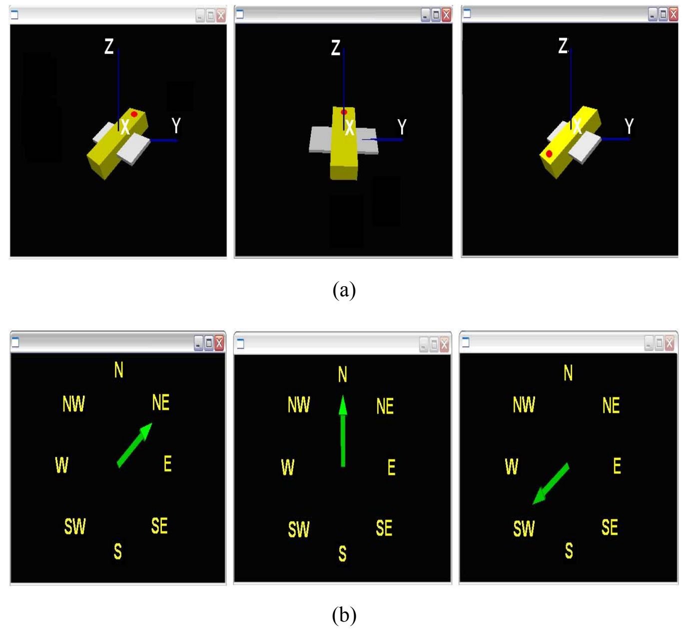

Figure 5. Graphic interface developed to test the proposed sensory fusion algorithm. In (a) Virtual body and (b) Compass

V. Experimental Setup and Results

The validation of the developed AHRS using the process of enforcing the orthogonality conditions and also the sensory fusion is made through the experimental framework showed in Fig 6. The experimental setup consists of a Hummingbird quad-rotor, and a Ground Control Station (GCS). The design, construction, and testing of a UAV requires a large amount of time. Thus, as the objective of this work is not build a quad-rotor, a commercial platform was used for the experimental tests. The Hummingbird fits the physical size requirements: a tip-to-tip wingspan of 55 cm, a height of 8 cm, and a weight of about 500 grams including a battery. The platform is durable enough to survive most crashes while the blades are soft enough not to cause damage during such event. Furthermore, the 20 minute battery life and 200 g payload capacity are also advantageous [5].

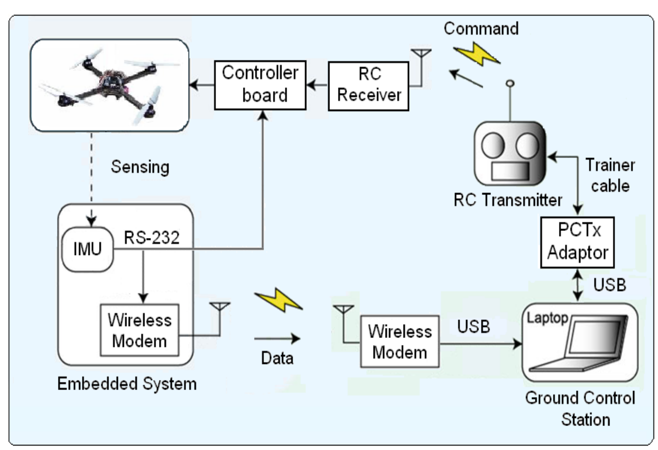

Figure 6. Experimental setup used.

During the flight, the quad-rotor's rotational motions are sensed by the IMU. The collected information is processed using an on-board processing system. Then, the computed attitude is transmitted to the embedded microcontroller board and also to GCS using the established communication protocols.

The Ground Control Station allows the user to define the input command to the attitude control loop. The reference signal can be generated by the standard RC Transmitter or through a position control loop implemented in Simulink. In this paper, experiments have been conducted using the RC transmitter signal. The produced command by the RC transmitter is sent to Simulink before being transmitted to the UAV. The RC transmitter is attached to the USB port of the laptop through a commercial interface, Endurance's PCTx, used to transmit and to receive the command signal of the Ground Control Station. Thus, the received telemetry data from the on-board AHRS and also the transmitted input command for the quad-rotor can be visualized in graphics plotted into the GCS.

An approach based on Proportional-Derivative (PD) controller was used to stabilize the aircraft due to their simplicity and easy implementation. The estimated attitude angles of the quad-rotor are applied as feedback to the PD controllers. The error signals are used to produce the control signals for the actuators. Using a mixing of the computed control signals, the desired speed to each rotor can be obtained correctly. The desired rotor speeds are sent to the motor controllers through the I^2C bus. The control signals calculation is executed on the embedded microcontroller board. Fig. 7 shows the architecture of the controller module implemented for the stabilization of the quad-rotor robot.

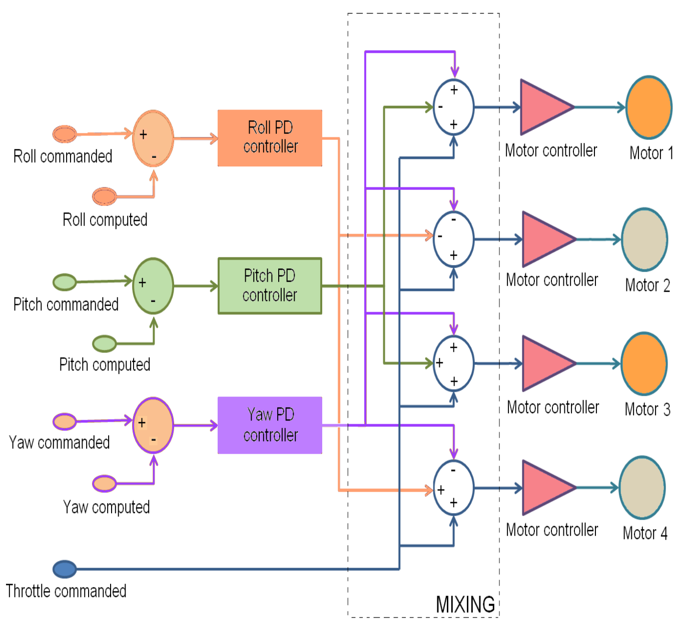

Figure 7. PD controller block diagram.

During the experimental tests, the quad-rotor height is controlled manually from the RC transmitter. The validation of Attitude and Heading Reference System applied to stabilization control of the aircraft during hovering is verified through the experimental results. Fig 8 (a), (b), and (c) illustrate the roll, pitch and yaw responses, respectively, obtained during the evaluation of the AHRS developed.

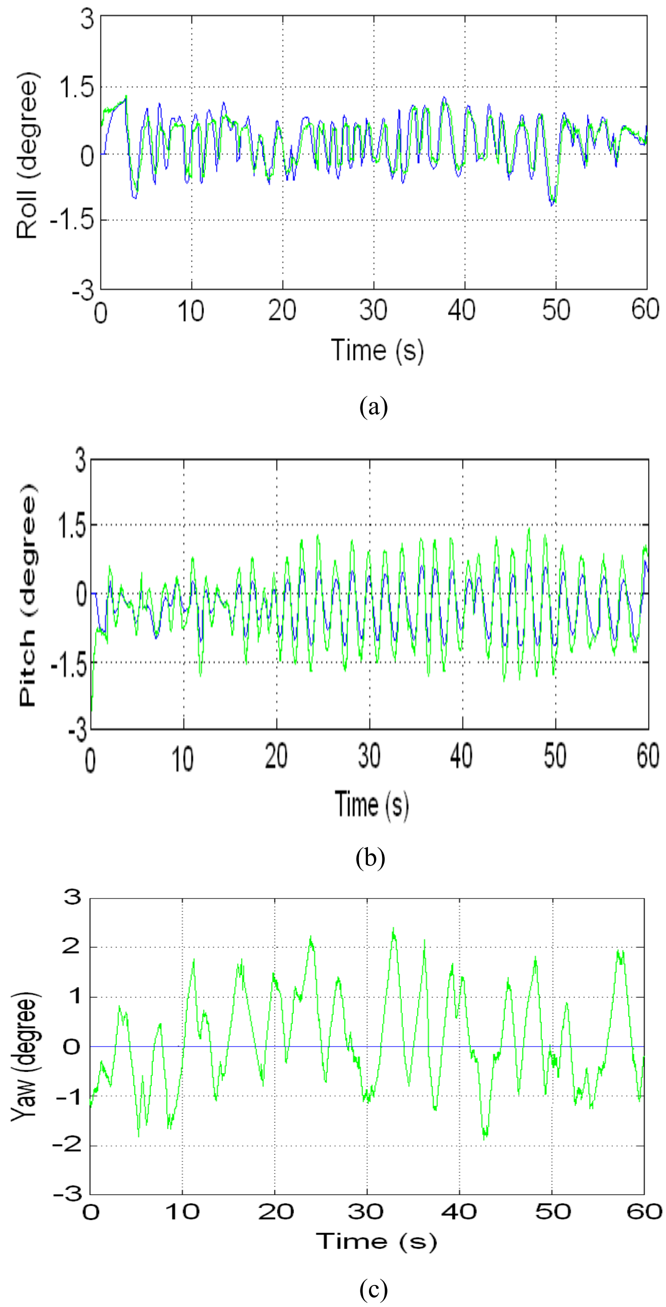

Figure 8. Attitude responses obtained using the experimental setup during a hovering flight. In (a) Roll response, (b) Pitch response and (c) Yaw response.

As one can see, the roll, pitch and yaw responses track the desired references for the hovering flight. The performance can be improved by means of refined tuning of the control parameters. The experiment results demonstrate the effectiveness of the developed Attitude and Heading Reference System applied to the control of rotation angles of the UAV.

Fig. 9 shows the quad-rotor robot during the execution of a hovering flight using the developed AHRS and also the implemented onboard control algorithm.

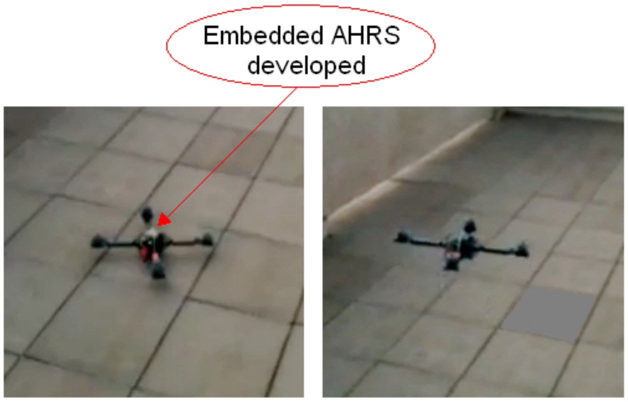

Figure 9. Hummingbird quad-rotor robot during the experimental test.

VI. CONCLUSIONS

In this paper, a real-time implementation and the results of the Attitude and Heading Reference System using low cost sensors are presented. This system has been integrated to the attitude control loop implemented in an embedded microcontroller. Using the accurately computed attitude, the UAV can be stabilized during flight. To evaluate the performance of the integrated system, flight tests were conducted with a quad-rotor robot. This platform can be easily adapted for the implementation of other control methods and also used in different types of unmanned aerial vehicles in several applications.

The use of AHRS proved to be very valuable for the stabilization control of a quad-rotor. It enables the developer to evaluate many aspects of the aircraft autopilot hardware, to refine the control algorithm, to test and tune control parameters, and many others. The developed system uses a wireless network to send the computed information to a Ground Control Station. It permits the monitoring of the obtained states during flight experiments. Also, the experimental results show that the method used to obtain the attitude is effective for the stabilization control of a quad-rotor during hovering flight, even while executing high speed maneuvers.

ACKNOWLEDGEMENT

The authors would like to thank Canadian Directorate of Technical Airworthiness Engineering and Support 6 and CAPES Foundation (process n. 9855114), Ministry of Education of Brazil for the financial support.

REFERENCES

[1] C. Liu, Z. Zhou, and X. Fu, “Attitude Determination for MAVs Using a Kalman Filter,” In Proceedings of the IEEE Tsinghua Science and Technology, 2008.

[2] D. Long, H. Chao, and Y. Chen, “A Two-stage Calibration Method for Low-cost UAV Attitude Estimation using Infrared Sensor,” In Proceedings of the IEEE/ASME International Conference on Mechatronics, Embedded Systems and Applications, 2010.

[3] W. Adiprawita, A. S. Ahmad, and J. Sembiring1, “Development of AHRS (Attitude and Heading Reference System) for Autonomous UAV (Unmanned Aerial Vehicle),” In Proceedings of the International Conference on Electrical Engineering and Informatics, 2007.

[4] J. F. Guerrero-Castellanos, H. Madrigal-Sastre, S. Durand, N. Marchand, W. F. Guerrero-Sanchez, and B. B. Salmeron, “ Design and implementation of an Attitude and Heading Reference System (AHRS),” In Proceedings of the IEEE International Conference on Electrical Engineering Computing Science and Automatic Control, 2011.

[5] D. Gurdan, J. Stumpf, M. Achtelik, K.-M. Doth, G. Hirzinger., and D. Rus., “Energy-efficient Autonomous Four-rotor Flying Robot Controlled at 1 kHz,” In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2007.

[6] S. R. B. Santos, C. L. Nascimento, and S. N. Givigi, “Design of Attitude and Path Tracking Controllers for Quad-Rotor Robots using Reinforcement Learning,” In Proceedings of the IEEE International Aerospace Conference, 2012.

[7] Y. Jiong, Z. Lei, J. Deng, R. Shu, and J. Wang, “GPS/SINS/BARO Integrated Navigation System for UAV,” In Proceedings of the IEEE International Forum on Information Technology and Applications, 2010.

[8] K. Nonami, F. Kendoul, S. Suzuki, W. Wang, and D. Nakazawa, “Design and Implementation of Low-Cost Attitude Quaternion Sensor,” Autonomous Flying Robots: Unmanned Aerial Vehicles and Micro Aerial Vehicles, Chapter 11, DOI 10.1007/978-4-431-53856-1 11, Springer, 2010.

[9] J. Farrell, and M. Barth., “The Global Positioning System & Inertial Navigation,” ISBN: 0071369449, 9780071369442, McGraw Hill Professional, 1998.

[10] M. S. Grewal, L. R. Weill, and A. P. Andrews, “Global Positioning System Inertial Navigation and Integration,” ISBN: 0470099712, 9780470099711, John Wiley & Sons, 2007.

[11] Honeywell, “Electronic Compass Design Guide using the HMC5843 Digital compass IC”, Honeywell Magnetic Sensor Products, 2002.