Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

CO-DESIGN OF A SAFE NETWORK CONTROL QUADROTOR

C. Berbra, S. Lesecq, S. Gentil, J.-M. Thiriet

GIPSA-lab Control Systems Dpt, CNRS INPG UJF, BP 46, 38402 Saint Martin d'Hères Cedex France (Tel: +33 476 82 62 44; e-mail: {cedric.berbra, suzanne.lesecq, sylviane.gentil, jean-marc.thiriet}@gipsa-lab.inpg.fr).

Abstract: This paper deals with the co-design of a Network Control System (NCS) and its diagnosis. Residuals are shown to be affected by network packet losses. A new indicator, sensitive to packet losses is proposed and thus changes in the residuals due to faults or to packet losses can be differentiated. Results are exemplified through the simulation of a 4-rotor helicopter using the Matlab/Simulink standard. The network is simulated thanks to the TrueTime toolbox.

Keywords – Fault Detection and Isolation, Fault Tolerance, Network Control System, Co-design, quadrotor, UAV, 4-rotor helicopter.

1. INTRODUCTION

Miniature rotorcraft-based Unmanned Aerial Vehicles (UAVs) have received a growing interest in industrial and academic research. Thanks to their hover capability, they are prone to be useful for many civilian missions such as video supervision of road traffic, surveillance of urban districts, forest fire detection or building inspection. Various industrial and research areas such as electronics, aeronautics, computer science and high technology meet in order to build these complex and sophisticated systems. UAVs need to fulfill classical properties (stability, precision, maneuverability) but also reliability and safety. Actually, the use of UAVs is clearly safety critical. The occurrence of faults can be extremely detrimental to the equipment and surrounding. Thus, early Fault Detection and Isolation (FDI) techniques must be implemented, as well as Fault Tolerant Control (FTC) schemes. Moreover, hardware redundancy is inconceivable because of load limitation and power autonomy.

A key challenge for civilian applications of UAVs is to develop a cheap and robust system, able to achieve its missions with the required safety level. While fixed wing vehicles have had extensive applications for military and meteorological purposes due to their range, speed and flight duration, rotorcraft vehicles are considered to be more preferable for civilian applications. The quadrotor is a small vehicle controlled by the rotational speed of four rotors (Cowling et al., 2007). It benefits from having very few constraints on motion and an ability to carry a "high" payload compared to its own weight. Usually, low-cost, lightweight components are assembled to build the quadrotor, each of the components likely to be affected by faults (sensor faults, actuator faults, malfunction of the communication network, etc.). Due to its low-cost and simplicity, the quadrotor provides an excellent test bench for application of advanced control techniques (Tayebi et al., 2004; Sanchez et al., 2007; Guerrero-Castellanos et al., 2007) and embedded diagnostic strategies (Tanwani et al., 2007b). The quadrotor is a partially redundant structure, excellent for applying FDI methods, as well as designing FTC schemes. FDI techniques have been applied to autonomous vehicles such as cars, aircrafts (Napolitano et al., 1998, Simani et al., 2006). To our knowledge, few FDI applications to autonomous helicopters (Heredia et al., 2005) and (Planar) Vertical Take-Off and Landing ((P)VTOL) aircraft systems (Sharma and Aldeen, 2007) have been reported in the literature.

UAV architectures and especially the quadrotor one may be embedded and distributed systems, controlled via a network. By means of the network, the wire weight can be decreased and maintenance procedures become easier. However, the network induces some drawbacks. Actually, some network parameters (for instance the limited bandwidth or a traffic increase) can result in additional unknown delays or packet losses (Cervin et al., 2003). The effects of the delay influence on control systems have been studied for some years. There is now a great interest in the study of the delay influence on FDI algorithms (Kambhampati, Patton and Uppal, 2006). Another crucial problem with NCS is due to packet losses: important information may be missed. Control and diagnosis algorithms have thus to be adapted to this possible data loss. This paper focuses on this problem rather than on the delay influence. The network load and the resulting packet loss have an influence on the residuals and thus on the diagnostic results. With respect to our previous work, (Tanwani et al., 2007b), the diagnostic algorithm is now adapted to packet losses thanks to a specific residual.

The simulation of the network is done with the TrueTime toolbox (Anderson, Henriksson, Cervin, 2005). This library provides specific blocks for the network interface modeling in the Simulink environment. It is developed in C++ language and the files are compiled in Matlab by using an external C++ compiler. TrueTime simulates various networks that can be implemented in Simulink for the simulation of Network Control Systems (NCS). The CAN network has been chosen for the quadrotor (Tanwani et al., 2007a).

The paper is organized as follows: Section 2 briefly presents the quadrotor equipped with an Inertial Measurement Unit (IMU). The attitude observer and the control law are shortly summarized due to paper length limitation. Section 3 describes the diagnostic algorithm that has been developed for this application. Section 4 aims at comparing results obtained by the diagnostic algorithm without network and when a CAN network is embedded. In order to avoid false alarms, a new fault indicator is proposed: it takes into account packet losses that may appear in the network. Section 5 concludes this paper.

2. PRESENTATION OF THE SYSTEM

2.1 Description of the quadrotor structure

Two frames are considered (Fig. 1): the inertial frame R(e_x, e_y, e_z) and the body frame B(e_1, e_2, e_3) attached to the aircraft with its origin at the centre of mass of the quadrotor. The quadrotor is mechanically simpler than classical helicopters: it does not have swashplate and it has constant pitch blades (Escareno, Salazar-Cruz and Lozano, 2006). By design, the quadrotor is controlled by independently varying the rotational speed ω_{mi} of each electric motor. Force f_i produced by motor i is proportional to the square of the rotational speed. The total thrust is given by:

$$T = \sum_{i=1}^{4} f_i = L \sum_{i=1}^{4} \omega_{mi}^2 \tag{1}$$

The three torques applied to the structure are given by:

$$\tau_a^1 = db(\omega_{m2}^2 - \omega_{m4}^2) \quad \tau_a^2 = db(\omega_{m1}^2 - \omega_{m3}^2) \quad \tau_a^3 = k(\omega_{m1}^2 + \omega_{m3}^2 - \omega_{m2}^2 - \omega_{m4}^2) \tag{2}$$

where d is the distance from the rotors to the centre of mass of the quadrotor. The motors are supposed identical. L>0 and k>0 depend of the air density, the radius, the shape, the pitch angle of the blades and others factors. The complete mechanical model can be found in (Tanwani et al., 2007b).

In this study, the attitude is not represented by the classical angles yaw-pitch-roll (ϕ, θ, ψ) but it is modelled with a unit quaternion $q = [q_0 \vec{q}^{\,T}]^T \in R^4$ (Chou 1992; Hamel et al., 2002). Thus, the dynamical equations related to the attitude are given by:

$$I_f \dot{\omega} = -\omega \times I_f \omega - G_a + \tau_a \tag{3}$$

$$\dot{q} = (1/2)\,\Omega(\omega) q = (1/2)\,\Xi(q)\,\omega \tag{4}$$

$$\Xi(q) = \begin{bmatrix} -q^T \\ q_0 I + [\vec{q}^{\times}] \end{bmatrix}, \quad \Omega(\omega) = \begin{bmatrix} 0 & -\omega^T \\ \omega & -[\omega^{\times}] \end{bmatrix}, \quad [\omega^{\times}] = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \tag{5}$$

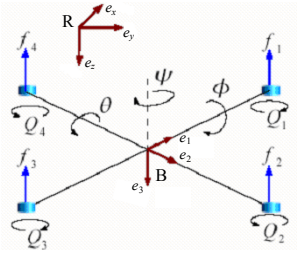

Fig. 1. Coordinate frames R and B

$I_f \in R^{3\times3}$ is the inertia matrix (symmetric definite positive) and ω are the angular velocities of the quadrotor measured by three rate gyros in frame B. × is the cross product. The gyroscopic torques G_a due to the combination of the rotation of the quadrotor and the four rotors, are modeled as:

$$G_a = \sum_{i=1}^{4} I_r (\omega \times e_z)(-1)^{i+1} \omega_{mi} \tag{6}$$

I_r is the moment of inertia of each motor (supposed equal). Note that if $\vec{r}$ is expressed in R, its coordinates in B are:

$$\vec{b} = C(q)\vec{r} \quad \text{or} \quad b = q^{-1} \otimes r \otimes q \tag{7}$$

where ⊗ is the quaternion product, $r = (0, \vec{r}^{\,T})^T$, $b = (0, \vec{b}^{\,T})^T$, $q^{-1} = [q_0 \; -\vec{q}^{\,T}]^T$ and $C(q)$ is the Rodrigues matrix defined as:

$$C(q) = (q_0^2 - \vec{q}^{\,T}\vec{q})I + 2(\vec{q}\vec{q}^{\,T} - q_0 [q^{\times}]) \tag{8}$$

2.2 Inertial Measurement Unit

The attitude estimation is a prerequisite for flight control. As a consequence, a central navigation unit (MEMS technology) is embedded in the quadrotor. It consists of a tri-axis accelerometer (a_1, a_2, a_3), three magnetometers (m_1, m_2, m_3) and three rate gyros (g_1, g_2, g_3) mounted at right angle. The sensor measurements (expressed in frame B) are modelled as:

$$\text{rate gyro: } \omega_g = \omega + \beta + \eta_1, \quad \dot{\beta} = -T^{-1}\beta + \eta_2 \tag{9}$$

$$\text{accelerometers: } b_{acc} = C(q)(a - g e_z) + \eta_{acc} \tag{10}$$

$$\text{magnetometers: } b_{mag} = C(q) h_m + \eta_{mag} \tag{11}$$

β is the bias inherent in rate gyro measurements, $T = \tau I_3$ and τ = 100s. η_i, i = {1, 2, *acc*, *mag*} are assumed to be Gaussian white noises of appropriate dimension. The motion is supposed quasi-static so that acceleration a is neglected. $g = 9.81 \text{ms}^{-2}$ is the constant gravity acceleration. h_m = (h_{mx}, 0, h_{mz}) represents the magnetic field measured in frame R. Note that (10), (11) are static non linear equations.

2.3 Attitude state observer and attitude control

Measurements (b_{mag}, b_{acc}, ω_g) are used to feed a non linear observer (Guerrero-Castellanos et al., 2005; Guerrero-Castellanos et al., 2006), whose block diagram is given in Fig. 2. From b_{mag} and b_{acc}, a pseudo measured quaternion q_{ps} is computed:

$$q_{ps} = \arg\min_{\|q\|^2=1} \left\{ (1/2) \left\| \begin{bmatrix} b_{acc}^T & b_{mag}^T \end{bmatrix}^T - h(q) \right\|^2 \right\} \tag{12}$$

where $h(q)$ is derived from (10) and (11). $\hat{q}$ is obtained by propagating the kinematics equation (4) using ω_g, $\hat{\beta}$ and the discrepancy between $\hat{q}$ and q_{ps}:

$$q_e = \hat{q} \otimes q_{ps}^{-1} = [q_{e0} \; \vec{q}_e^{\,T}]^T \qquad \|q_e\| = 1 \tag{13}$$

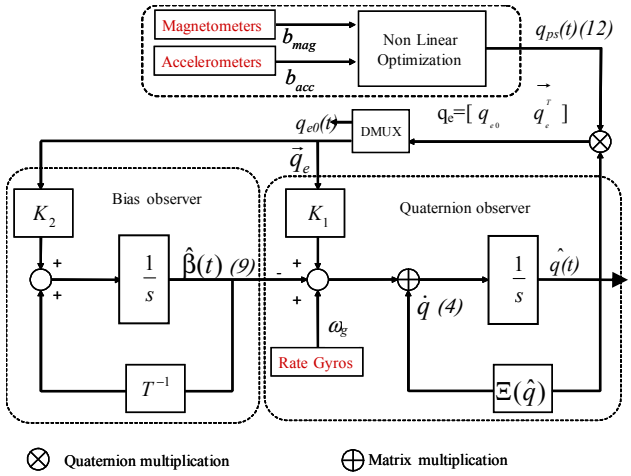

Fig. 2. Non linear observer for attitude control(K_1>0, K_2>0)

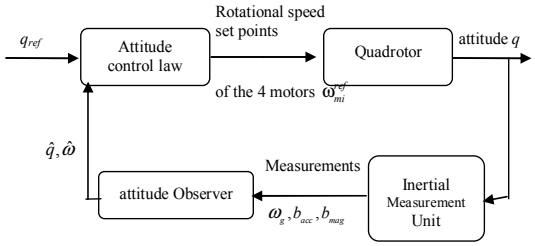

Fig. 3. The quadrotor control loop

The rotational speed and quaternion estimation $\hat{\omega}$ and $\hat{q}$ are used in a feedback loop. The attitude reference is given by a quaternion reference (Fig. 3). The controller implemented for the attitude stabilization is detailed in (Guerrero-Castellanos et al., 2007).

2.4 Embedded CAN Network

A distributed architecture of the quadrotor based on a Controller Area Network (CAN) is implemented (Fig. 4). The control and the diagnostic modules are fed through the network that is considered as a component of the system and not only as a simple communication media.

The network is characterized by the traffic of 17 periodic flows whose order of priority is: the four flows from the main control unit (ω_{mi}^{ref} , i = 1:4, for each local motor control), the nine flows from the IMU (ω_{gi} ,i = 1:3, b_{acci}, i = 1:3, b_{magi}, i = 1:3) and the four flows from the motor speed measurements (ω_{mi}, i = 1:4).

The sensor task is time-triggered; the sampling time is T_s = 10ms. Note that the IMU data acquisition is synchronised. The data rate is 1Mbits/seconds. The data length is 59 bits for all the periodic flows. Each flow is transmitted in 59µs. Thus all the periodic flows are transmitted in 1ms (17×59µs), corresponding to 10% of the sampling period, which is negligible. The observer and control algorithms have been discretised. The controller task and the diagnostic task are event triggered and they must wait for all the sensor values before any computation is performed (Berbra, et al., 2007). ω_{mi}^{ref} is sent through the network to the local motor control as soon as the controller has finished its computation.

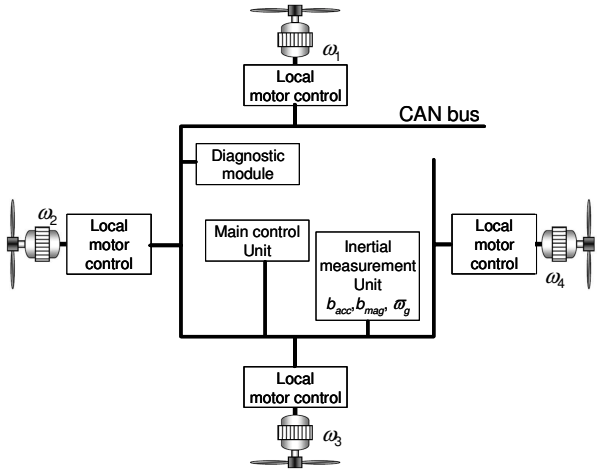

Fig. 4. Quadrotor architecture with the embedded network

3. FDI ALGORITHMS

The use of a bank of observer to design structured residuals is well known (Isermann, 2006). As a state observer has been designed for the control synthesis, it seems quite natural to investigate if this observer can be adapted to the diagnostic purpose.

3.1 Fault diagnosis of the inertial measurement unit

The accelerometer and magnetometer measurements satisfy (10), (11). Thus, a fault in one of these sensors is diagnosed thanks to the resolution of a non linear optimization problem similar to the one in (12) that takes now 5 over the 6 measurements (Fig. 5). 6 such estimators are designed. Under the hypothesis that actuators are fault-free, a bank of 3 observers that take 2 out of 3 rate gyro measurements is implemented to diagnose rate gyro faults (Fig. 5). The non linear observer is similar to the one in Fig. 2 except that the measurement ω_{gi} that is discarded is replaced by ω_i computed according to (3). The estimated quaternions $\hat{q}_{res}(i)$, i = 1:9 are compared to the quaternion q_{model} obtained from the model to provide the error quaternions (14). $\phi_e(i)$ is chosen as residual and plotted in degrees. Actually, it has some physical meaning: it represents the angle to pass from $\hat{\vec{q}}_{res}$ to $\vec{q}_{model}$, thus representing the error in the rotation computed with the model or estimated with part of the measurements. This residual is sensitive to errors in the drone inertia I_f used in the model (Tanwani et al., 2007b).

$$\begin{aligned} q_e(i) &= \hat{q}_{res}(i) \otimes q^{-1} \left[q_{e0}(i) \;\; \vec{q}_e^{\,T}(i) \right]^T \qquad i = 1:9 \\ &= \left[\cos(\phi_e(i)/2) \;\; \sin(\phi_e(i)/2) \vec{u}_e^{\,T}(i) \right]^T \end{aligned} \tag{14}$$

The fault signature table obtained in case of rate gyro faults is given in Table 1 while Table 2 presents the fault signature table for accelerometer and magnetometer faults. In these tables, the first column represents the residual when the measurement issued from the corresponding sensor is discarded. As can be seen, both tables are strongly isolable. For instance, the estimator $gyro_1$ is sensitive to faults in all the sensors (f_{ai}, f_{mi}, f_{gi}, i = 1, 2, 3) except f_{g1}, which indicates a fault in the rate gyro 1 (along the x-axis). The same reasoning applies to all other symbols.

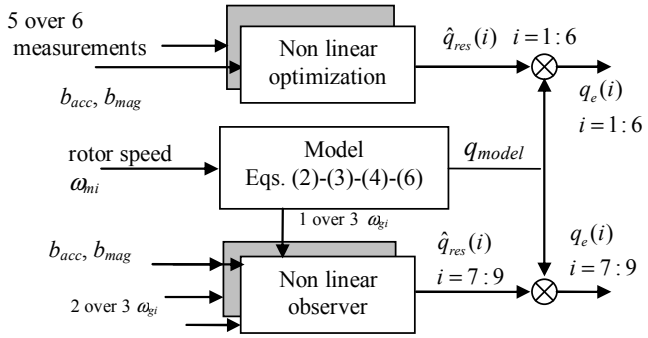

Fig. 5. Residuals generation for sensor faults

Table 1. Signature table for diagnosis of rate gyro faults

	f_{a1}	f_{a2}	f_{a3}	f_{m1}	f_{m2}	f_{m3}	f_{g1}	f_{g2}	f_{g3}
$gyro_1$	1	1	1	1	1	1	0	1	1
$gyro_2$	1	1	1	1	1	1	1	0	1
$gyro_3$	1	1	1	1	1	1	1	1	0

Table 2. Signature table for diagnosis of accelerometer and magnetometer faults.

	f_{a1}	f_{a2}	f_{a3}	f_{m1}	f_{m2}	f_{m3}	f_{g1}	f_{g2}	f_{g3}
acc_1	0	1	1	1	1	1	0	0	0
acc_2	1	0	1	1	1	1	0	0	0
acc_3	1	1	0	1	1	1	0	0	0
mag_1	1	1	1	0	1	1	0	0	0
mag_2	1	1	1	1	0	1	0	0	0
mag_3	1	1	1	1	1	0	0	0	0

3.2 Diagnosis of Actuators

The 4 actuators are independent. The model of each electrical motor has been used for their diagnosis, (Fig. 6, Table 3). A fault in the rotational speed sensor or in the corresponding motor cannot be isolated. This limitation could be removed by additional sensors to measure the current or the voltage. However, this solution is contrary to the weight constraint.

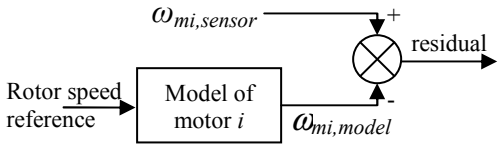

Fig. 6. Residual generation for actuators

Table 3. Signature table for diagnosis of actuators

	$f_{rotor\ 1}$	$f_{rotor\ 2}$	$f_{rotor\ 3}$	$f_{rotor\ 4}$
$rotor_1$	1	0	0	0
$rotor_2$	0	1	0	0
$rotor_3$	0	0	1	0
$rotor_4$	0	0	0	1

4. NETWORK INFLUENCE

Simulation results are now presented for unfaulty and faulty situations, without or with the embedded CAN network described in section 2.4. The attitude is given in the yaw-pitch-roll formulation for the sake of clarity. In all the experiments, the quadrotor is in an initial attitude and the reference attitude is given by $q = (1\ 0\ 0\ 0)^T$ leading to (0 0 0) for (ϕ, θ, ψ). Fig. 7 shows the small differences on the attitudes estimated by the non linear observer in the fault-free case, without and with the network (no packet losses). Fig. 8 and Fig. 9 show the residuals obtained in the fault-free case from (14), respectively without and with the network. The residuals are similar. Thus, the network without packet losses has no influence on the FDI results. The network induced delay is small compared to the sampling period, which explains this result. The residual acc2 and mag2 variations are explained by the great sensitivity of the optimization procedure to the lack of the respective measurements.

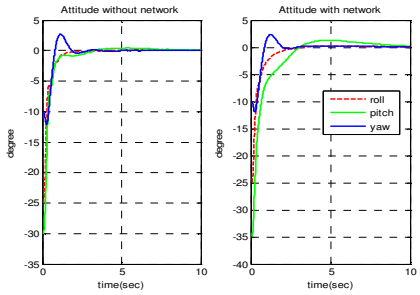

Fig. 7. Attitude estimated by the non linear observer (left: without the network, right: with the network)

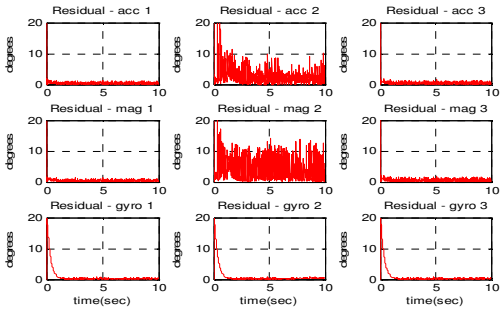

Fig. 8. Sensor residuals (fault-free case, without the network)

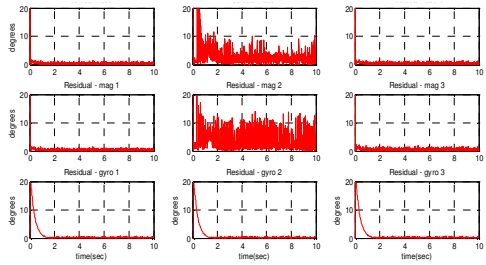

Fig. 9. Sensor residuals (fault-free case, with the network)

4.1 No packet losses

Fig. 10 and 11 show residuals acc_i and mag_i, i = 1:3 without and with the network when a breakdown in m_1 (magnetometer along the x-axis) is introduced at t = 4s. As expected, mag_1 is insensitive to f_{m1} while the other residuals exhibit "high values" after the fault injection, which is consistent with Table 2. Note that the residuals obtained with or without the network are quite similar.

A failure of actuator 1 f_{rotor1} is now introduced at t = 4s. Fig. 12 shows the four related residuals (Table 3), with the network, that are again quite similar to the case without the network. As expected, the residual $rotor_1$ is clearly sensitive to f_{rotor1} while the other three residuals exhibit "low values" after the fault injection.

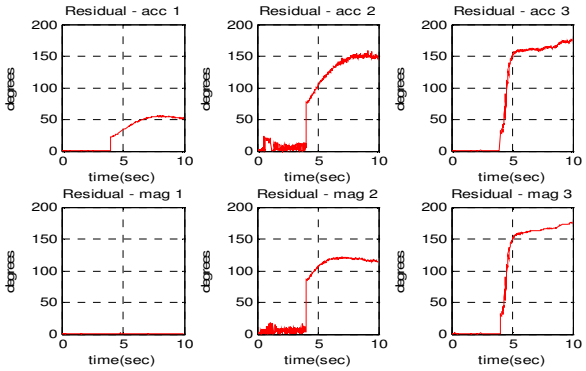

Fig. 10. acc_i and mag_i, i = 1:3 (f_{m1}, without the network)

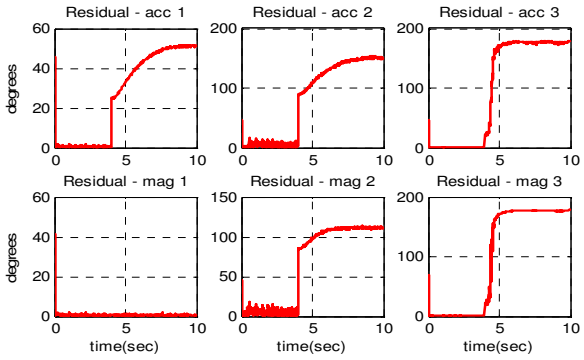

Fig. 11. acc_i and mag_i, i = 1:3 (f_{m1}, with the network)

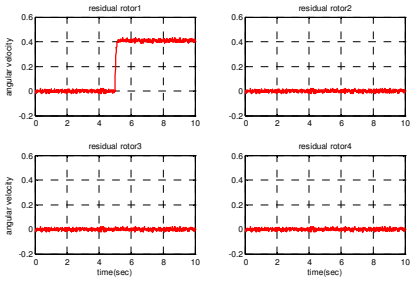

Fig. 12. Actuator residuals (f_{rotor1}, with the network)

4.3 Network malfunction: Packet losses

Thanks to the TrueTime capabilities, packet losses can be easily introduced in the network. 10% of packet losses are considered in the next experiment. Berbra et al. (2007) have shown that packet losses may induce false alarms when the diagnostic task is triggered every time a new data is received leading to residual computation with desynchronized data. Consequently, they have proposed a new fault indicator for each data $r^i_{network}$, equal to "1" when the data is not received on time by the control and diagnostic modules (Fig. 13). When a data has not been received in the current sampling period kT_e *(in practice received after* kT_e+0.66T_e*)*, its corresponding indicator equals "1". When a packet is lost at $t = kT_e$, the quaternion $\hat{q}(kT_e)$ is not computed and the control algorithm maintains the references ω^{ref}_{mi} computed at time $t = (k - 1)T_e$. At time $t = (k+1)T_e$ the quaternion is computed by the observer taking into account $\hat{q}((k-1)T_e)$ (Fig. 14). Small differences can be noted with respect to Fig. 7 but it can be seen that the control law is robust to 10% of packet losses. Fig. 15 presents residual acc_1, valid only when $r^i_{network}$ equals "0". Note that different experiments with other faults have been done in the case of 10% of packet losses on various data and the conclusions are identical to the ones given in the fault-free case.

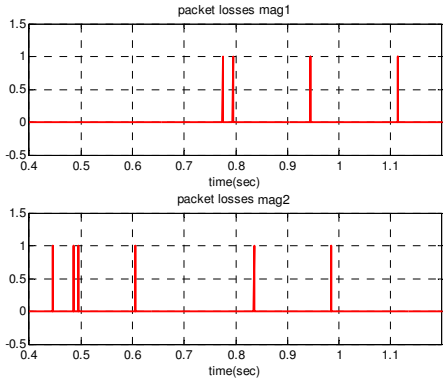

Fig. 13. Indicator of packet losses for mag_1 and mag_2

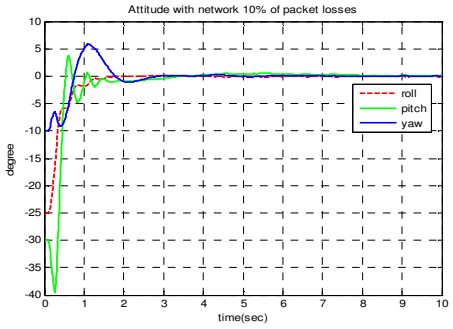

Fig. 14. Attitude estimated by the observer (10% of packet losses)

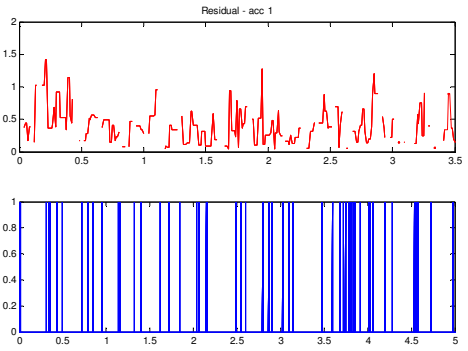

Fig. 15. Sensor residual acc_1 (no fault, 10% of packet losses)

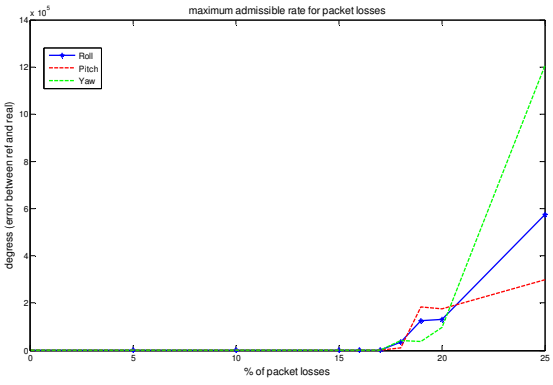

Fig. 16. Packet loss rate.

Figure 16 shows the influence of the packet loss rate on the difference between the reference and the real quaternion expressed in degrees as in (14). As can be seen, until 17%, no error is observed.

5. CONCLUSION AND PROSPECTS.

In this paper, the influence of an embedded network on the functioning of a quadrotor has been studied. The drone attitude is controlled thanks to a non linear state observer. A CAN network has been chosen for data transmission. It is well known that this network is suitable for control applications. In the present case, the network induced delays are negligible.

The comparison of FDI results without and with the network has been done with particular attention to packet losses. The implementation of the network FDI algorithm takes into account the necessity to process synchronized data. Thus, the residuals do not present false alarms due to packet losses. A packet loss indicator has been proposed. A residual is valid only when the data involved in its computation have their packet loss indicators equal to "0". In that way, the diagnosis is insensitive to the network packet losses.

When all the data necessary for the observer computation are not received, the control is simply maintained to its previous value. It has been observed that the results are insensitive to 17% of data lost. However, a reconfiguration of the observer could be done, leading to a bank of observers/estimators, the choice depending on the data available, exactly as it is done for residual generation. The system would be in this way more tolerant to packet losses. A prototype is actually under construction and real experiments will be done soon.

ACKNOWLEDGEMENTS

This work is partially supported by the SafeNeCS project funded by the ANR (Agence Nationale de la Recherche, France) under grant ANR-05-SSIA-0015-03.

REFERENCES

Anderson M., Henriksson D. and Cervin A., 2005, "*Truetime 1.4 – reference manual*", Department of Automatic Control, Lund Institute of Technology, Sweden.

Berbra C., Gentil S., Lesecq S. and Thiriet J.-M., 2007, "Co-design for a safe networked control DC motor", *3rd IFAC Workshop NECST,* Nancy, France.

Cervin A., Henriksson D., Lincoln B., Eker J. and Arzen K.E., 2003, "How Does Control Timing Affect Performance? Analysis and Simulation of Timing Using Jitterbug and TrueTime," *IEEE Control Systems Magazine,* Vol. 23, No. 3, pp. 16–30.

Chou J. C. K, 1992, "Quaternion kinematics and dynamic differential equations", *IEEE Transactions on Robotics and Automation*, Vol. 8, pp. 53-64.

Cowling I., Yakimenko O., Whidborne J. and Cooke A., 2007, "A prototype of an autonomous controller for a quadrotor", *ECC 2007*, Kos, Greece, 2-5 July.

Escareno J., Salazer-Cruz S. and Lozano R., 2006, "Embedded control of a four-rotor UVA, *American Control Conference*, Minneapolis, Minnesota, USA.

Guerrero-Castellanos J., Lesecq S. Marchand N. and Delamare J., 2005, "Attitude observer and control of a four-rotor helicopter", Report of the Laboratoire d'Automatique de Grenoble, France (in French).

Guerrero-Castellanos J., Lesecq S. Marchand N. and Delamare J., 2006, "Estimación de la orientación: aplicación a un mini-helicóptero con cuatro rotores", AMCA 2006, Congreso Nacional de Control Automático, México city, México.

Guerrero-Castellanos J., Hably A., Marchand N. and Lesecq S., 2007, "Bounded attitude stabilization: Application on four-rotor helicopter", *IEEE International Conference on Robotics and Automation*, 10-14 April.

Hamel T., Mahony R., Lozano R. and Ostrowski J., 2002, "Dynamic modelling and configuration stabilisation for an X4-flyer", *2002 IFAC World Congress*, Barcelona, Spain.

Herredia G., Ollero A., Mahtani R., Bejar M., Remuß V. and Musial M., 2005, "Detection of sensor faults in autonomous helicopters", *IEEE Int. Conf. on robotics and Automation*, Barcelona, Spain, April 2005.

Isermann R, 2006, "*Fault Diagnosis Systems: An introduction from fault detection to fault tolerance*", Springer-Verlag Berlin-Heidelberg.

Kambhampati C., Patton R. and Uppal F., 2006, "Reconfiguration in networked control systems: fault tolerant control and plug and play," *IFAC Symposium Safeprocess 2006*, Beijing, PRC.

Napolitano M., Windon D., Casanova J., Innocenti M. and Silvestri G., 1998, "Kalman filters and neural-network schemes for sensor validation in flight control systems", *IEEE Trans. on Control Systems Techn.*, Vol. 6, pp. 596-611.

Sanchez A., Castillo P., Escareno J., Romero H. and Lozano R., "Simple real-time control strategy to stabilize the PVTOL aircraft using bounded inputs", *ECC 2007*, Kos, Greece, 2-5 July.

Sharma R. and Aldeen M., 2007, "Fault and unknown input reconstruction in VTOL aircraft systems using sliding mode observer", *ECC 2007*, Kos, Greece, 2-5 July.

Simani S., Bonfè M., Castaldi P. and Geri W., 2006, "Application of fault diagnosis methodologies to a general aviation aircraft", *IFAC Symposium Safeprocess*, Beijing, PRC.

Tanwani A., Galdun J., Thiriet J-M., Lesecq S. and Gentil S., 2007a, "Experimental networked embedded minidrone - Part I. Consideration of Faults", *ECC 2007*, Kos, Greece, 2-5 July.

Tanwani A., Gentil S., Lesecq S. and Thiriet J-M., 2007b, "Experimental networked embedded minidrone - Part II. Distributed FDI", *ECC 2007*, Kos, Greece, 2-5 July.

Tayebi P. and Mc Gilvray, S., 2004, "Attitude, stabilization of a four rotor aerial robot," *43rd CDC*, Atlantis (Bah), USA.