Modelling and hardware co-simulation of a Quadrotor unmanned aerial vehicle

Soufiene Bouallègue*

Research Laboratory in Automatic Control (LARA),
National Engineering School of Tunis,
University of Tunis El Manar,
BP 37, Le Belvédère, 1002 Tunis, Tunisia
and
High Institute of Industrial Systems of Gabès,
University of Gabès,
6011 Gabès, Tunisia
Email: soufiene.bouallegue@issig.rnu.tn
*Corresponding author

Rabii Fessi

Research Laboratory in Automatic Control (LARA),
National Engineering School of Tunis,
University of Tunis El Manar,
BP 37, Le Belvédère, 1002 Tunis, Tunisia
and
National Engineering School of Gabès,
University of Gabès,
6000 Gabès, Tunisia
Email: rabii.fessi@gmail.com

Abstract: This paper deals with the modelling and hardware (HW) co-simulation of a Quadrotor vertical take-off and landing (VTOL) type of unmanned aerial vehicle (UAV). The developed HW co-simulation platform is based on a reconfigurable I/O (RIO) board of National Instruments (NI) Company, called sbRIO-9636, and a host PC with a real-time operating system (RTOS). The control design and simulation (CDSim) module of LabVIEW environment as well as an established network streams data communication protocol are used to emulate and co-simulate all flight dynamics’ within a processor-in-the-loop (PIL) framework. The flight motion principle of the Quadrotor, i.e. lift, rotation and translation, is firstly described as function of changes in the angular speed of the rotors. All aerodynamic forces and moments of such a vehicle are then described within an inertial earth frame and a nonlinear dynamical model is established thanks to the Newton-Euler formalism. The dynamics of the propellers’ brushless DC motors, accelerometer and gyroscope types of sensors are also modelled and co-simulated in order to complete the established model of the studied VTOL rotorcraft. HW simulations are carried out and compared to those obtained with software (SW) simulations in order to show the effectiveness of the proposed PIL co-simulation strategy.

Keywords: Quadrotor UAV; modelling; aerodynamic effects; rotors and sensors dynamics; Newton-Euler equations; PIL co-simulation; NI single-board RIO; LabVIEW.


Biographical notes: Soufiene Bouallègue graduated from the National Engineering School of Tunis in 2006 and received his PhD in Electrical Engineering in 2010. He is currently an Associate Professor of Electrical Engineering at the High Institute of Industrial Systems of Gabès. His research interests are in the area of metaheuristics optimisation, intelligent control, robotics, renewable energies, and digital control applications.

Rabii Fessi received his License Degree in Electronics, Electrotechnic and Automatic (EEA) and the Master Diploma in Automatic and Robotic Systems from the High Institute of Industrial Systems of Gabès in 2013 and 2015, respectively. He currently prepares his PhD thesis in Electrical Engineering in the National Engineering School of Gabès. His research interests include the modelling and nonlinear control of Quadrotor UAVs.
1 Introduction

The unmanned aerial vehicles (UAVs), particularly the Quadrotor-based architectures, are flying robots without pilot, which are able to conduct missions in autonomous or half-autonomous modes also in hostile and disturbed environments (Austin, 2010). Among the tasks to be conducted with these robots are found military acknowledgment, monitoring missions and civilian missions such as the inspection of dams and border monitoring, the prevention of forest fires and so on (Austin, 2010; Guerrero and Lozano, 2012; Marsh and Hill, 2008). These Quadrotors have seen a great evolution in terms of miniaturisation of their actuators and sensors, dynamics modelling and flight control design. This explains the interest shown by many researchers to study the flight dynamics modelling and simulation of these vehicles.

In Bouallègue and Fessi (2016), the authors used the Newton-Euler formalism to establish a nonlinear model of an UAV aircraft with fours rotors. Such a 12 degree of freedom dynamic model is implemented in a LabVIEW-based simulator and controlled. Becker et al. (2012) studied and modelled a mini-VTOL helicopter, called OS4 Quadrotor, to avoid flight collision. The authors used the Euler-Lagrange formalism, model identification, DC motor equations and blade element and momentum theories to model the OS4 Quadrotor. The well known Tait-Bryan angles were used for the parameterisation. In Huang (2016), the authors proposed the characteristic model formalism of Hongxin Wu to model a Quadrotor. They give an all-coefficient adaptive design method for the attitude control. In Li et al. (2016), an oblique cross Quadrotor is modelled and the active disturbance rejection control (ADRC) approach is designed to deal with its nonlinear and underactuated characteristics. In Wang et al. (2016), the proposed Quadrotor’s hybrid model, i.e. prior and fuzzy TS-based non-parametric, is used to design a fault tolerant controller to compensate the system uncertainties caused by damages failures.

To obtain a suitable mathematical description for the control and observation steps, the Quadrotor dynamical model must be verified, simulated and well prototyped before its use in the control design and definitive implementation stages. This verification stage, which can be efficiently achieved with adequate co-simulation platforms, is of growing interest in the real-world control application. Sophisticated SW/HW solutions for this systems design stage are usually needed and a powerful digital platform for achieving both the rapid prototyping and final real-world implementation is very required. This problem can be efficiently handled thanks to the computer-aided design (CAD) and processor-in-the-loop (PIL) concepts as shown in Ananthan and Vaidyan (2013), Datar et al. (2012), Hau and Khalil-Hani (2009), Zhang et al. (2013) Derouiche et al. (2016), Wang and Pham (2008), Wu et al. (2010), Dias et al. (2014) and Xie et al. (2014).

Recently, the National Instruments (NI) Company proposed advanced reconfigurable inputs and outputs (RIO) platforms, with increased processing power, like the embedded sbRIO-9636 devices (National Instruments, 2012). Such a digital RIO target remains well suited to perform the advanced processing tasks required by complex and hard applications such as the PIL cosimulation and rapid prototyping of nonlinear dynamical models. The final real-world implementation of various flight controllers becomes easy with work time-consuming reduction and practical tradeoffs handling for the Quadrotor UAV types of complex systems. Based on its suitable architecture and powerful onboard devices, the sbRIO-9636 target is a promised embedded solution for both prototyping and HW co-simulation of complex models and controllers for aerial robots. This digital board is associated to LabVIEW real-time (RT) and FPGA software tools for the deterministic execution of emulated models. So, in this paper we present the use of a developed PIL co-design platform for rapid prototyping and HW co-simulation of the established dynamical model of a Quadrotor UAV. Such a MBD methodology is built and successfully implemented around the embedded NI sbRIO platform and a host PC. A detailed study for the modelling of the VTOL rotorcraft is firstly presented while writing the Newton-Euler physical equations of such a vehicle. A dynamical nonlinear model is then obtained and HW co-simulated thanks to the developed NI sbRIO-9636 board-based PIL platform.

The remainder of this paper is organised as follows. Section 2 presents the description of the Quadrotor’s flight motion as well as the mathematical equations of all aerodynamic forces and torques in VTOL mode. Dynamics of the propellers’ brushless DC motors and sensors are presented. A dynamical nonlinear model is then established while using the Newton-Euler equations. Such a model is given in a canonical state-space form to be used in the control stage. In Section 3, the proposed PIL cosimulation platform is presented and the used NI sbRIO-9636 target is described. In Section 4, all numerical simulation results for the modelling and HW co-simulation stages are shown and discussed. Finally, conclusions are drawn in Section 5.

2 Quadrotor UAV modelling

2.1 System description

A Quadrotor is an UAV with four rotors that are controlled independently. It is a six degree-of-freedom (DOF) body as
shown in Figure 1. The movement of the Quadrotor results from changes in the speed of the rotors. The structure of the Quadrotor in this paper is assumed to be rigid and symmetrical. The centre of gravity and the body fixed frame origin coincide. The propellers are rigid. The thrust and drag forces are proportional to the square of propellers speed. The rotation speed of the propeller rotors relative to the ground is not taken into account.

Each motor $M_i$ ($i = 1, 2, 3$ and $4$) of the Quadrotor produces the force $F_i$ which is proportional to the square of the angular speed $\omega_i$. As the motors are turning only in a fixed direction, the produced force $F$ is always positive. The front and rear motors (M1 and M3) rotate counterclockwise, while the left and right motors (M2 and M4) rotate clockwise. As given in Austin (2010), the gyroscopic effects and aerodynamic torques tend to cancel in trimmed flight thanks to the mechanical design of the Quadrotor. The total thrust force $F$ is the sum of individual thrust of each motor.

Figure 1  Mechanical structure of the Quadrotor (see online version for colours)

The lift motion of the Quadricopter is obtained by the balanced distribution of the lift force provided by each motor. When varying the forces $F_1$, $F_2$, $F_3$ and $F_4$, the Quadrotor tilts lift to the low side which results a translation in trimmed flight thanks to the mechanical design of the Quadrotor. The studied Quadrotor is detailed with their body-frame $R_B$ ($O, x_b, y_b, z_b$) and inertial earth-frame $R_E$ ($O, e_x, e_y, e_z$) as shown in Figure 1. Let us denote by $m$ the total mass of the Quadrotor, $g$ the acceleration of the gravity and $l$ the distance from the centre of each rotor to the centre of gravity (COG) of the vehicle.

In accordance with the realistic Euler angles of the Quadrotor, we define the following rotational matrices associated to the roll, pitch and yaw angles, respectively:

$$\text{rot}_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi) & \cos(\phi) \end{bmatrix}$$

(1)

$$\text{rot}_y(\theta) = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix}$$

(2)

$$\text{rot}_z(\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

(3)

The orientation of the Quadrotor is defined by the rotation matrix $\mathcal{R} : R_E \rightarrow R_B$ which depends on the three rotational matrices of equations (1) to (3). Such an orientation matrix is given as follows:

$$\mathcal{R}(\phi, \theta, \psi) = \text{rot}_z(\psi) \times \text{rot}_y(\theta) \times \text{rot}_x(\phi)$$

$$\begin{bmatrix} 
ccss \sin(\theta) \cos(\theta) - \sin(\phi) \cos(\psi) \\
ccss \sin(\phi) \cos(\psi) + \cos(\phi) \sin(\theta) \\
\cos(\phi) \cos(\theta) 
\end{bmatrix}$$

(4)

where $c = \cos(\cdot)$ and $s = \sin(\cdot)$.

Let us consider the following model partitions of the Quadrotor naturally into translational and rotational coordinates:

$$\xi = (X, Y, Z)^T \in \mathbb{R}^3, \eta = (\phi, \theta, \psi)^T \in \mathbb{R}^3$$

(5)

where $\xi = (X, Y, Z)^T$ denotes the position vector of the Quadrotor COG relative to its fixed earth-frame and $\eta = (\phi, \theta, \psi)^T$ presents the attitude of the Quadrotor given by the Euler angles for rolling $\phi \in [-\pi/2, \pi/2]$, pitching $\theta \in [-\pi/2, \pi/2]$ and yawing $\psi \in [-\pi, \pi]$ motions.

Let a vector $\mathbf{v}_e = (v_x, v_y, v_z)^T$ denotes the linear velocity of the UAV in the earth-frame $R_E$ and it is defined as follows:

$$\mathbf{v}_e = \mathcal{R}(\phi, \theta, \psi) \mathbf{v}_b$$

(6)

where $\mathbf{v}_b$ is the linear velocity of the mass centre expressed in the body-frame. Consider the vector $\vartheta = (p, q, r)^T$ that represents the angular velocity in the frame $R_B$. The transformation of this vector from the body frame $R_B$ into the inertial one $R_E$ is derived as follows:
So, we can deduce the following velocities’ transformation relationship:

\[
\varphi = \begin{pmatrix} 0 \\ c\phi \theta \\ -s\phi \theta \end{pmatrix} + \begin{pmatrix} -s\theta \psi \\ c\phi \theta \psi \\ -c\phi \psi \end{pmatrix} = \begin{pmatrix} \phi - s\theta \psi \\ c\phi \theta + s\phi \theta \psi \\ c\phi \psi - s\phi \theta \end{pmatrix}
\]

(7)

where \( T \) is the well known velocities’ transformation matrix which is invertible.

Each motor \( M_i \) (\( i \in \{1, 2, 3, 4\} \)) of the Quadrotor produces the thrust force \( F_i \) which is proportional to the square of the angular speed, denoted as \( \omega_i \) as depicted in Figure 2. The thrust force generated by the \( i^{th} \) rotor of the Quadrotor is given by:

\[
F_i = \frac{1}{2} \rho C_T r^2 \omega_i^2 = b \omega_i^2
\]

(9)

where \( \rho \) is the air density, \( r \) and \( \Lambda \) are the radius and the section of the propeller, respectively. The term \( C_T \) is the aerodynamic thrust coefficient.

The aerodynamic drag torque, caused by the drag force at the propeller of the \( i^{th} \) rotor and opposed the motor torque, is defined as follows:

\[
\delta_i = \frac{1}{2} \rho C_D r^2 \omega_i^2 = d \omega_i^2
\]

(10)

where \( C_D \) is the aerodynamic drag coefficient.

The pitch torque is a function of the difference \( (F_3 - F_1) \) and the roll torque is proportional to the term \( (F_2 - F_4) \). The yaw one is the sum of all reactions torques generated by the four rotors and due to the shaft acceleration and propeller drag. All these pitching, rolling and yawing torques are defined respectively as follows (Guerrero and Lozano, 2012; Austin, 2010):

\[
\Gamma_{\phi} = l(F_3 - F_1)
\]

(11)

\[
\Gamma_{\theta} = l(F_2 - F_4)
\]

(12)

\[
\Gamma_{\psi} = \lambda l(F_1 - F_2 + F_3 - F_4)
\]

(13)

where \( \lambda \) is a proportional coefficient.

Two gyroscopic effects torques, due to the motion of the propellers and the Quadrotor body, are additively provided. These moments are given as:

\[
M_{gp} = \sum_{i=1}^{4} \Omega \wedge \left( 0, 0, J_r (-1)^{i+1} \omega_i \right)^T
\]

(14)

\[
M_{gb} = \Omega \wedge J \Omega
\]

(15)

where \( \Omega = \dot{\eta} \) is the vector of angular velocities in the fixed earth-frame, \( J_r \) is the Z-axis inertia of the propellers’ rotors and \( J \) denotes the constant inertia matrix of the Quadrotor which is defined as follows:

\[
J = \begin{pmatrix}
J_{xx} & -J_{xy} & -J_{xz} \\
-J_{yx} & J_{yy} & -J_{yz} \\
-J_{zx} & -J_{zy} & J_{zz}
\end{pmatrix}
\]

(16)

where

\[
J_{xx} = \int (Y^2 + Z^2) \, dm, \quad J_{xy} = -\int XY \, dm,
\]

\[
J_{xz} = -\int XZ \, dm, \quad J_{yy} = \int (X^2 + Z^2) \, dm,
\]

\[
J_{yz} = -\int YZ \, dm, \quad J_{zz} = \int (X^2 + Y^2) \, dm.
\]

We assume that the Quadrotor has a symmetric structure in terms of the distance between the motors and its centre where the well known electronic control unit (ECU) will be placed. Therefore, the above equation of the inertia is re-expressed as:
To calculate the inertial moment terms $J_{xx}$, $J_{yy}$ and $J_{zz}$, the centre of the studied Quadrotor is assumed to have a parallelepipedal shape of length ‘$L$’, width ‘$W$’ and height ‘$H$’, as shown in Figure 2. Therefore, these inertial moments are given as follows:

$$
\begin{align*}
J_{xx} &= \frac{1}{12} m (H^2 + L^2) + 2l^2 \\
J_{yy} &= \frac{1}{12} m (W^2 + L^2) + 2l^2 \\
J_{zz} &= \frac{1}{12} m (W^2 + H^2) + 4l^2
\end{align*}
$$

(18)

2.3 Rotors dynamics

The generated thrust force and aerodynamical torques, given in equation (9) and equations (11) to (13), respectively, are provided by four brushless DC motors which have a high torque and little friction (Austin, 2010). In the steady-state regime, a brushless motor has the same dynamics of a conventional DC motor. Each motor is coupled to a propeller which is characterised by the non-flapping effect. Hence, the armature voltage of the $i^{th}$ Brushless DC motor is defined as follows:

$$
v_i = \frac{R_{mot}}{k_{mot}} J_i \omega_i + k_{mot} \omega_i + d R_{mot} \omega_i^2
$$

(19)

where $R_{mot}$ and $k_{mot}$ denote the internal resistance and torque coefficient of the Brushless motors, respectively, $d$ is the drag propellers’ coefficient.

Since that the drag coefficient $d$ is very small, this dynamic can be approximated as a first order lag transfer function where the characteristic parameters can be identified by experimental trials as shown in (Becker et al., 2012). From a practical point of view, the Brushless DC motors are controlled by Pulse Width Modulated (PWM) signals becoming from the ECU hardware. The supplied voltage is directly proportional to the RPM of their rotations. So, the Quadrotor is controlled by independently varying the speed of these four rotors. The effective control inputs of the Quadrotor are thus defined as follows:

$$
\begin{pmatrix}
u_1 \\
u_2 \\
u_3 \\
u_4
\end{pmatrix} = \begin{pmatrix} F \\
\Gamma_x \\
\Gamma_y \\
\Gamma_z
\end{pmatrix} \begin{pmatrix}
\frac{b}{b} \frac{b}{b} \frac{b}{b} \\
\frac{0}{-b} \frac{b}{0} \frac{b}{b} \\
\frac{0}{-b} \frac{b}{0} \frac{b}{b} \\
\frac{d}{d} \frac{d}{d} \frac{d}{d}
\end{pmatrix} \begin{pmatrix} \omega_1 \\
\omega_2 \\
\omega_3 \\
\omega_4
\end{pmatrix}
$$

(20)

where $\omega_{1,2,3,4}$ are the angular speeds of the four rotors, respectively.

From this relationship, it can be observed that the input $u_1$ denotes the total thrust force on the Quadrotor body in the $Z$-axis. The inputs $u_2$ and $u_3$ represent the roll and the pitch torques, respectively. The input $u_4$ represents the yawing torque.

2.4 Sensors dynamics

The feedback states of the Quadrotor can be measured thanks to an embedded inertial measurement unit (IMU) that usually contains accelerometer and gyroscope types of sensors (Guerrero and Lozano, 2012; Becker et al., 2012; Austin, 2010). The accelerometer and the gyroscope use the translational and rotational flight movements to provide the linear and angular velocities, respectively. So, the generated accelerometer and gyroscope outputs are given along $X$, $Y$, and $Z$ axes by equations (21) and (22) respectively:

$$
y^{acc} = \alpha^{acc} v_\theta + \beta^{acc} + \gamma^{acc}
$$

(21)

where $y^{acc} = (y^\alpha_{X}, y^\alpha_{Y}, y^\alpha_{Z})^T$ are the sensor outputs’ voltage, $\alpha^{acc}$ are the accelerometer gains, $v_\theta = R^{-1} (\phi, \theta, \psi) \psi$ denotes the linear velocities in the body-frame, $\beta^{acc} = (\beta^\alpha_{X}, \beta^\alpha_{Y}, \beta^\alpha_{Z})^T$ are the sensor bias and $\gamma^{acc} = (\gamma^\alpha_{X}, \gamma^\alpha_{Y}, \gamma^\alpha_{Z})^T$ present the zero mean white noises.

$$
y^{gyro} = \alpha^{gyro} \phi + \beta^{gyro} + \gamma^{gyro}
$$

(22)

where $y^{gyro} = (y^\phi_{X}, y^\phi_{Y}, y^\phi_{Z})^T$ are the sensor outputs’ voltage, $\alpha^{gyro}$ are the gyroscopic torques and $\beta^{gyro} = (\beta^\phi_{X}, \beta^\phi_{Y}, \beta^\phi_{Z})^T$ are the gyroscopic gains, $\phi$ denotes the angular velocities in the body-frame, $\gamma^{gyro} = (\gamma^\phi_{X}, \gamma^\phi_{Y}, \gamma^\phi_{Z})^T$ present the zero mean white noises.

2.5 Modelling with Newton-Euler formalism

Using the Newton-Euler formalism for modelling, the Newton’s laws lead to the following motion equations of the Quadrotor:

$$
\begin{pmatrix}
\dot{\xi} \\
\dot{\eta} \\
\dot{\zeta}
\end{pmatrix} = \begin{pmatrix}
F_x + F_y + F_z \\
F_y + \Gamma_x + F_z \\
\Gamma_x + \Gamma_y + F_z
\end{pmatrix}
$$

(23)

where $F_x = (mg \cos \theta \cos \phi, mg \cos \theta \sin \phi, mg \sin \theta)^T$ is the air drag force which resists to the Quadrotor motion, $F_y = (0, 0, mg)^T$ is the gravity force, $M = (\Gamma_x \Gamma_y \Gamma_z)^T$ represents the total rolling, pitching and yawing torques. The terms $M_{gy}$ and $M_{gb}$ are the gyroscopic torques and $M_{a} = diag(k_{a}, k_{b}, k_{c}) \phi$ is the torque resulting from aerodynamic friction.

By substituting the position vector and the forces with their expressions into equation (23), we have the following translational dynamics of the Quadrotor:
\[
\begin{align*}
\dot{X} &= \frac{1}{m} (c\phi\psi s\theta + s\phi\psi) u_i - \frac{K_1}{m} X \\
\dot{Y} &= \frac{1}{m} (c\phi\psi s\theta - s\phi\psi) u_i - \frac{K_2}{m} Y \\
\dot{Z} &= \frac{1}{m} c\phi\psi u_i - g - \frac{K_3}{m} Z
\end{align*}
\]

From the second part of equations (23), and while substituting each moment by its expression, we deduce the following rotational dynamics of the rotorcraft:

\[
\begin{align*}
p &= qr \frac{J_{yy} - J_{zz}}{J_{xx}} - \frac{J_y}{J_{xx}} \phi \dot{q} - \frac{K_2}{J_{xx}} p + \frac{1}{J_{xx}} u_z \\
q &= pr \frac{J_{zz} - J_{xx}}{J_{yy}} + \frac{J_z}{J_{yy}} \phi \dot{r} - \frac{K_3}{J_{yy}} q + \frac{1}{J_{yy}} u_y \\
r &= pq \frac{J_{xx} - J_{yy}}{J_{zz}} \frac{K_0}{J_{xx}} r + \frac{1}{J_{zz}} u_x
\end{align*}
\]

According to the established equations (24) and (25), \( x = (\phi, \dot{\phi}, \theta, \phi, \psi, \psi, X, \dot{X}, Y, \dot{Y}, Z, \dot{Z})^T \) is retained as the state-space vector of the nonlinear model of the Quadrotor rewritten as the following form:

\[
\dot{x} = f(x, u)
\]

where

\[
\begin{align*}
u_i &= c\phi\psi s\theta + s\phi\psi, \quad u_i = c\phi\theta\psi - s\phi\psi, \\
a_1 &= \left( J_{yy} - J_{zz} \right) / J_{xx}, \quad a_2 = \kappa_2 / J_{xx}, \quad a_3 = -J_r, \quad a_4 = \left( J_{zz} - J_{xx} \right) / J_{yy}, \\
a_5 &= \left( J_{yy} - J_{xx} \right) / J_{zz}, \quad a_6 = \kappa_3 / J_{yy}, \quad a_7 = -J_r, \quad a_8 = \kappa_4 / J_{zz}, \quad a_9 = -J_r, \quad a_{10} = -\kappa_5 / J_{zz} \\
b_1 &= -\kappa_2 / m, \quad b_2 = -\kappa_3 / m, \quad b_3 = -\kappa_4 / m
\end{align*}
\]

Note that \( \kappa_{2,3,4,5,6} \) are the aerodynamic friction and translational drag coefficients, \( \phi_\theta = \phi_1 - \phi_2 + \phi_3 - \phi_4 \) is the overall residual rotor angular speed and \( x = (x_1, x_2, x_3, x_4, \\
x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11}, x_{12})^T \) is the above Quadrotor state-vector.

### 3 PIL co-simulation platform

The proposed hardware setup of the NI single-board RIO-based PIL co-simulation is depicted in Figure 3. A similar platform, which is based on a NI myRIO-1900 target, has been used in our previous works (Bouallègue and Fessi, 2016) to implement and prototype various control algorithms for the Quadrotor. Here, we propose another solution for HW co-simulation of the established dynamical model of the Quadrotor and not of its control algorithm. The latter can be deployed now on the host PC. In this paper, the built PIL solution is based mainly on the embedded NI Single-Board RIO platform. Well-suited for complex processing and real-time computing, this target is associated to a host PC with LabVIEW/CDSim and LabVIEW/Robotics environments. Thanks to its powerful tools, LabVIEW software environment simplifies construction and prototyping of dynamic systems and provides the ability to co-simulate in real-time framework the behaviour of modelled systems. After prototyping phase, this NI device operates autonomously to execute a LabVIEW project which is deployed on its RT dual core processor and/or FPGA circuit.

The used RIO platform, as depicted in Figure 4, includes standard hardware architecture, known as sbRIO-9636, that includes a floating point processor running a RTOS, an Xilinx FPGA target, and modular I/O which can be programmed using a single development toolchain. From hardware characteristics point of view, the used NI sbRIO-9636 board is equipped with a 400 MHz PowerPC processor and a non-volatile memory with 512 MB and a DRAM with 256 MB for deterministic control and analysis. Such a RIO target has a reconfigurable Xilinx Spartan-6 LX45 FPGA for custom timing, inline processing and control. In the sbRIO-9636 board there are 16 analogue inputs and four analogue outputs with resolution of 16-bit as well as 28 bidirectional 3.3 V DIO lines. Several IO ports such as 10/100BASE-T Ethernet, RS232 and RS485 serial, USB, CAN, and SDHC are available on this board supplied with 9 VDC to 30 VDC voltages. The maximum voltage range are about –10 V to +10 V (National Instruments, 2012). All these onboard devices make this target very suitable for the emulation and rapid prototyping of complex dynamical models, especially those of UAV aircrafts. On the other hand, the measurement data communication between different VIs, which implement the different sub-models of the Quadrotor, is achieved thanks to the implemented LabVIEW server/client architecture (Bouallègue and Fessi, 2016). In this PIL scheme, the NI sbRIO-9636 board acts as a ‘Client’ and the host PC as the ‘Server’. Such a built data communication protocol is based on the network streams tool of the LabVIEW software.
Modelling and hardware co-simulation of a Quadrotor unmanned aerial vehicle

4 Co-simulation results and discussions

The physical parameters of the Quadrotor are taken from Becker et al. (2012) and summarised in Table 1. For the SW/HW simulations, an implementation of all established models and aerodynamic equations is carried out under MATLAB/Simulink environment for SW simulations and after that under LabVIEW/sbRIO-9636 platform for HW co-simulations.

Figure 5 shows the built S-Function block for the nonlinear model (26) of the VTOL rotorcraft. In Figure 6, Figure 7 and Figure 8, are shown the simulation models of the rotors’ and sensors’ dynamics, respectively. These blocks will be cascaded with the under-actuated model of the VTOL drone of Figure 5. Recall that both accelerometer and gyroscope sensors give the measured translational and rotational velocities in the body-frame.

From initial conditions equal to zero, we co-simulate the open-loop Quadrotor’s dynamics to show the angular motors’ speeds in Figure 9, the position, attitude, linear and angular velocities in Figure 10, Figure 11, Figure 12 and Figure 13, respectively. From these results, we show that the SW and HW co-simulated responses are relatively close with some deviations due to the hardware constraints of implementation that we will have to take into account when implementing control algorithms. Different control and observation algorithms can now be developed based on the sbRIO-based emulated model of the drone that is closest to physical reality. However, the proposed PIL emulation platform will be used for the implementation and co-simulation of such algorithms. Compared to other prototyping and HW co-simulation methods of the literature, especially those using low cost microcontrollers and textual programming, the proposed NI sbRIO-9636-based technique is more efficient and faster since it is based on graphical programming concepts. Its deterministic execution of the implemented controllers leads to fast and real-time tests and prototyping.

Table 1 Model parameters of the Quadrotor vehicle

<table>
<thead>
<tr>
<th>Param.</th>
<th>Description</th>
<th>Values/unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>m</td>
<td>Mass of the Quadrotor</td>
<td>0.650 kg</td>
</tr>
<tr>
<td>l</td>
<td>Rotor distance to COG</td>
<td>0.23 m</td>
</tr>
<tr>
<td>Jxx</td>
<td>Body inertia along X-axis</td>
<td>0.0075 kg.m^2</td>
</tr>
<tr>
<td>Jyy</td>
<td>Body inertia along Y-axis</td>
<td>0.0075 kg.m^2</td>
</tr>
<tr>
<td>Jzz</td>
<td>Body inertia along Z-axis</td>
<td>0.0130 kg.m^2</td>
</tr>
<tr>
<td>Jr</td>
<td>Propellers inertia along Z-axis</td>
<td>6.0e-5 kg.m^2</td>
</tr>
<tr>
<td>d</td>
<td>Drag propellers coeff.</td>
<td>3.23e^-7</td>
</tr>
<tr>
<td>b</td>
<td>Lift body coeff.</td>
<td>2.92e^-5</td>
</tr>
<tr>
<td>g</td>
<td>Acceleration of the gravity</td>
<td>9.81 m.s^-2</td>
</tr>
<tr>
<td>κ_x</td>
<td>X-axis translational drag coeff.</td>
<td>5.57e^-4</td>
</tr>
<tr>
<td>κ_y</td>
<td>Y-axis translational drag coeff.</td>
<td>5.57e^-4</td>
</tr>
<tr>
<td>κ_z</td>
<td>Z-axis translational drag coeff.</td>
<td>6.35e^-4</td>
</tr>
<tr>
<td>κ_x</td>
<td>X-axis aero friction coeff.</td>
<td>5.57e^-4</td>
</tr>
<tr>
<td>κ_y</td>
<td>Y-axis aero friction coeff.</td>
<td>5.57e^-4</td>
</tr>
<tr>
<td>κ_z</td>
<td>Z-axis aero friction coeff.</td>
<td>6.35e^-4</td>
</tr>
</tbody>
</table>
Figure 5  Simulation model of the Quadrotor UAV (see online version for colours)

Figure 6  Simulation model of the rotors’ dynamics (see online version for colours)

Figure 7  Simulation model of the accelerometer sensor
Figure 8  Simulation model of the gyroscope sensor

Figure 9  Angular speeds of Brushless DC motors (see online version for colours)
Figure 10  Position dynamics of the Quadrotor (see online version for colours)

Figure 11  Attitude dynamics of the Quadrotor (see online version for colours)
Figure 12  Linear velocities dynamics of the Quadrotor (see online version for colours)

Figure 13  Angular velocities dynamics of the Quadrotor (see online version for colours)
5 Conclusions

In this paper, a nonlinear dynamical model of a Quadrotor UAV is established using the Newton-Euler formalism and emulated on a hardware NI sbRIO-9636 board. All aerodynamic forces and moments of such a vehicle are described within an inertial frame. The propellers’ brushless DC motors as well as the embedded flight sensors’ dynamics are studied and modelled. The used sbRIO-9636-based PIL platform for the emulation and HW co-simulation of the dynamical model is described. Demonstrative co-simulation results are carried out and show the effectiveness of the proposed dynamic modelling and HW co-simulation of the studied VTOL Quadrotor. Indeed, the obtained SW and HW co-simulation results are generally close for the different dynamics of the Quadrotor. They show the effectiveness of the proposed rapid prototyping and HW emulation methodology. The closed-loop HW co-simulation of the VTOL rotorcraft becomes easy and the practical real-world implementation of flight controllers is made possible with the same used hardware target and software tools.

References


