Control architecture and simulation of the Borea quadrotor

Carlos Perez-Montenegro, Mauricio Lotufo, Enrico Canuto

Politecnico di Torino, Dipartimento di Automatica e Informatica
Corso Duca degli Abruzzi 24, 10129 Torino, Italy(e-mail: carlos.perez@polito.it)

Abstract: The paper presents modelling, control design and simulated results preliminary to test the Borea quadrotor**.** Guidance, navigation and control strategy are implemented following the Embedded Model Control (EMC) scheme.

Keywords: Quadrotor, simulation, embedded model control, guidance, navigation.

1. INTRODUCTION

The Borea quadrotor vehicle in Fig. 1 is under construction and test.

Fig. 1. The Borea quadrotor (not flying).

Preliminary to in-field tests a simulator has been built for testing and debugging the control code. Control design exploits the Embedded Model Control (EMC) framework (Canuto et al., 2012). The control unit is built around the vehicle dynamics embedded model, which is a set of discrete time state equations of centre-of-mass (CoM) and attitude dynamics. It has been shown by Canuto et al. (2013a) that the embedded model and the control strategies can be partitioned into horizontal, vertical and spin dynamics. Their cross-couplings are treated either as known or unknown disturbances to be estimated by state predictors (embedded model plus noise estimator) and rejected by the control law. As a non-conventional feature, CoM horizontal dynamics and attitude dynamics (vehicle tilt, pitch and roll) constitute a unique dynamics. Assuming moderate tilt (acrobatic flight is not an immediate goal), the nonlinear dynamics from angular jerk to CoM acceleration is feedback linearized (Slotine et al, 1991) to make available a fifth-order linear embedded model for each horizontal degree of freedom. Something similar has been done by Guerrero and Castillo (2011).

The EMC design implies the development of the noise estimators, this navigation allows the control unit to estimate the variables accurately. In this research, a simple method to make state predictors that guarantee stability with minimum variance is presented. The horizontal noise estimator implements a decoupled strategy that allows the attitude variables to be estimated with a classical state predictor, this prediction is connected with the motion noise estimator through a nonlinear gain also predicted. In addition the horizontal navigation is the same for the vertical and horizontal position.

The vertical guidance uses a standard minimum control strategy (Yanushevsky, 2011). The horizontal guidance splits in two identical models due to the feedback linearization. The strategy used is a polynomial strategy implemented with a two-phase state machine. The guidance avoids abrupt changes in the states, and thus the attitude variables are not forced, this feature is considered important to the stability of the quadrotor vehicles. The polynomial guidance also is adaptive to the measurements, and allows the reference states to be updated before a new maneuver.

The vertical control law is made with a classical state feedback. The horizontal case uses a state feedback with variable gains that are computed at each control step.

Navigation is implemented using the embedded model and noise estimator structure typical of Embedded Model Control. The details presented are just a summary. Sensor fusion is not treated, neither the performance design. Simulated results are presented.

2. VEHICLE DYNAMICS

1.1 Quadrotor

Dynamics and control of quadrotor vehicles is well known and widely studied (, Azzam 2010, Dikmen 2009, Hoffmann, 2010, Ping, 2012, Jaimes, 2008, Tayebi, 2004, Tewari, 2011). Here a summary is provided, in view of the derivation of the embedded model in Section 3.

Newton equation of the vehicle CoM is written with respect to the local vertical local horizontal frame $\mathcal{L} = \{O, \vec{l}_1, \vec{l}_2, \vec{l}_3\}$, assumed to be inertial, where $\vec{l}_3$ points to zenith and O is the target landing point. The propeller thrust components are expressed in the body frame,

$\mathcal{B}=\left\{C,\vec{b}_1,\vec{b}_2,\vec{b}_3\right\}$, where C is the vehicle CoM, $\vec{b}_3$ is the symmetry axis, nominally aligned with the thrust vectors. The body-to-local transformation R_b^l is selected to be function of the 123 Euler vector $\boldsymbol{\theta}$ for reasons to be explained below. It holds

$$R_b^l(\boldsymbol{\theta})=X(\varphi)Y(\theta)Z(\psi)=\begin{bmatrix} \mathrm{c}_\theta & 0 & \mathrm{s}_\theta \\ \mathrm{s}_\phi \mathrm{s}_\theta & \mathrm{c}_\phi & -\mathrm{s}_\phi \mathrm{c}_\theta \\ \mathrm{c}_\phi \mathrm{s}_\theta & -\mathrm{s}_\phi & \mathrm{c}_\phi \mathrm{c}_\theta \end{bmatrix} Z(\psi) \quad (1)$$

$$\boldsymbol{\theta}^T=\begin{bmatrix}\varphi & \theta & \psi\end{bmatrix}$$

Local position and velocity coordinates are denoted by $\mathbf{r}$ and $\mathbf{v}$. Thrust vectors and body coordinates are denoted with $\vec{F}_k$, $k=1,..,4$ and $\mathbf{F}_k$. They are expressed as follows (Martin and Salaun, 2010, Roskam and Lan, 2003, Seddon, 2011)

$$\begin{aligned} &\vec{F}_k=c_{\omega f}(r,n,p)\omega_k{}^2\vec{v}_k(\varphi_k,\theta_k)+c_{wf}(r,n,p)\omega_k\vec{w}_k+...= \\ &=f_k\left(\omega_k{}^2\right)\vec{v}_k+\vec{d}_k \\ &\mathbf{F}_k=f_k\left(\omega_k{}^2\right)\mathbf{v}_k+\mathbf{d}_k \end{aligned} \quad (2)$$

where $\vec{v}_k(\varphi_k,\theta_k)$ is the thrust direction depending on the misalignments (φ_k,θ_k), ω_k is the rotor angular speed and $\vec{w}_k$ the relative wind speed, which is approximated by the speed of the trust application point $\vec{a}_k$. The first term in (2) is the commanded term, whereas the second term $\vec{d}_k$ is the rotor drag. The coefficients $c_{\omega f}, c_{wf}$ depend on the blade radius r, the number n of blades and on the pitch p. CoM dynamics can be written as

$$\begin{aligned} &\dot{\mathbf{r}}(t)=\mathbf{v}(t),\ \mathbf{r}(0)=\mathbf{r}_0 \\ &\dot{\mathbf{v}}(t)=-\mathbf{g}+R_b^l(\boldsymbol{\theta})V\mathbf{f}(t)/m+\mathbf{d}/m, \mathbf{v}(0)=\mathbf{v}_0, \\ &\mathbf{g}=\begin{bmatrix}0\\0\\g\end{bmatrix},\ V=\begin{bmatrix}\mathbf{v}_1 & \mathbf{v}_2 & \mathbf{v}_3 & \mathbf{v}_4\end{bmatrix},\ \mathbf{f}=\begin{bmatrix}f_1\\f_2\\f_3\\f_4\end{bmatrix} \end{aligned} \quad (3)$$

where m is the vehicle mass, g is the local gravity acceleration, V is the thrust direction matrix and $\mathbf{d}$ summarizes rotor drag and high-frequency parasitic forces due to rotor vibrations. Sometimes CoM dynamics is written in body coordinates, but position and velocity reference trajectories are given in local coordinates, which suggests to compute command forces in local coordinates and then to convert them to body frame.

The propeller moment $\vec{M}_k$ and the body coordinate $\mathbf{M}_k$ is written as follows

$$\begin{aligned} &\vec{M}_k=s_k c_{\omega m}(r,n,p)\omega_k{}^2\vec{v}_k+c_{wm}(r,n,p)\omega_k\vec{w}_k\times\vec{v}_k+ \\ &+\vec{a}_k\times\vec{F}_k+...= \\ &=s_k m_k\left(\omega_k{}^2\right)\vec{v}_k+\vec{a}_k\times\vec{v}_k f_k\left(\omega_k{}^2\right)+\vec{d}_{mk} \\ &\mathbf{M}_k=s_k m_k\mathbf{v}_k+f_k\left(\omega_k{}^2\right)\mathbf{a}_k+\mathbf{d}_{mk} \end{aligned} \quad (4)$$

In (4) the coefficients $c_{\omega m}, c_{wm}$ depend as in (2) on propeller geometry, $\vec{a}_k$ is the thrust application point and $\mathbf{a}_k$ is the body vector of $\vec{a}_k\times\vec{v}_k$, $s_k=\pm1$ is the moments sign, $\vec{d}_{mk}$ is the rotor drag moment, and $\mathbf{d}_{mk}$ includes rotor drag and vibration torques.

The 123 attitude kinematics is the following

$$\begin{aligned} &\dot{\boldsymbol{\theta}}(t)=\begin{bmatrix}1/\cos\theta & 0 & 0\\ 0 & 1 & 0\\ -\tan\theta & 0 & 1\end{bmatrix}Z(\psi)\boldsymbol{\omega}(t)=A(\theta)\boldsymbol{\omega}_s(t) \\ &\boldsymbol{\theta}(0)=\boldsymbol{\theta}_0 \end{aligned} \quad (5)$$

which shows that $Z(\psi)$ can be absorbed by the de-spun angular rate $\boldsymbol{\omega}_s=Z(\psi)\boldsymbol{\omega}$, with the advantage that the first two angles (pitch θ and roll φ) become decoupled, and spin kinematics can be treated separately, though coupled with roll. Indeed it can be used to orient body axes and not to orient the thrust vector to achieve horizontal motion.

The propeller rotor dynamics assumes that the brushless motor is regulated such as to provide an angular rate ω_k proportional to the command voltage V_k through a first-order dynamics

$$\dot{\omega}_k(t)=-q\omega_k(t)+qbV_k(t),\ \omega_k(0)=\omega_{k0} \quad (6)$$

where q, b are common motor parameters.

Euler dynamics can be written in terms of the de-spun angular rate $\boldsymbol{\omega}_s$ as follows

$$\begin{aligned} &\dot{\boldsymbol{\omega}}_s(t)=-J^{-1}\boldsymbol{\omega}\times\left(J\boldsymbol{\omega}+J_r\sum_{k=1}^4\omega_k\mathbf{v}_k\right)-\dot{Z}(\psi)\boldsymbol{\omega}+ \\ &J^{-1}\left(SV\mathbf{m}+A\mathbf{f}\right)+J^{-1}\mathbf{d}_m,\ \boldsymbol{\omega}_s(0)=\boldsymbol{\omega}_{s0} \end{aligned} \quad (7)$$

where S is the diagonal matrix of the torque sign s_k, J_r is the rotor inertia and J is the vehicle inertia matrix free of rotor inertia.

1.2 Nominal dispatching and simplified dynamics

Using (3) and (7) the command force vector $\mathbf{F}_u$ and the torque command vector $\mathbf{M}_u$ are related to propeller thrust and torque vectors through

$$\begin{bmatrix}\mathbf{F}_u\\ \mathbf{M}_u\end{bmatrix}=\begin{bmatrix}V & 0\\ A & SV\end{bmatrix}\begin{bmatrix}\mathbf{f}\\ \mathbf{m}\end{bmatrix}=(B+\Delta B)\begin{bmatrix}\mathbf{f}\\ \mathbf{m}\end{bmatrix}, \quad (8)$$

where B is the nominal dispatching matrix and ΔB the deviation. The nominal propeller layout is such to obtain

$$\begin{aligned} &B=\begin{bmatrix}B_z & 0\\ B_t & B_m\end{bmatrix} \\ &B_z=\begin{bmatrix}0&0&0&0\\0&0&0&0\\1&1&1&1\end{bmatrix},\ B_m=\begin{bmatrix}0&0&0&0\\0&0&0&0\\-1&1&-1&1\end{bmatrix}=\begin{bmatrix}0\\0\\B_{mz}\end{bmatrix}. \\ &B_t=\begin{bmatrix}0&d/2&0&-d/2\\-d/2&0&d/2&0\\0&0&0&0\end{bmatrix}=\begin{bmatrix}B_{mxy}\\0\end{bmatrix} \end{aligned} \quad (9)$$

Including dispatching deviation among unknown disturbances, CoM and attitude dynamics can be simplified in a form suitable to Embedded model. It can be shown (Canuto, 2013a) that the CoM dynamics (3) splits into horizontal and vertical dynamics as follows. The horizontal

dynamics holds

$$\begin{bmatrix}\dot{x}\\ \dot{y}\end{bmatrix}=\begin{bmatrix}v_x\\ v_y\end{bmatrix}$$
$$\begin{bmatrix}\dot{v}_x\\ \dot{v}_y\end{bmatrix}(t)=a(t)\begin{bmatrix}q_x\\ q_y\end{bmatrix}(t)+\begin{bmatrix}d_x\\ d_z\end{bmatrix}(t),\begin{bmatrix}v_x\\ v_y\end{bmatrix}(0)=\begin{bmatrix}v_{x0}\\ v_{y0}\end{bmatrix}, \quad (10)$$
$$\begin{bmatrix}q_x\\ q_y\end{bmatrix}=\begin{bmatrix}\sin\varphi\cos\theta\\ \sin\theta\end{bmatrix}$$

where the following local coordinate change has been done

$$\begin{bmatrix}x\\ y\end{bmatrix}=\begin{bmatrix}-r_y\\ r_x\end{bmatrix}, \quad (11)$$

and a is the vertical acceleration to be defined below. The vertical dynamics holds

$$\dot{v}_z(t)=-g+\frac{a}{\cos\theta\cos\varphi}+d_z(t)=-g+u_z+d_z(t)$$
$$a(t)=\frac{1}{m}\sum_{k=1}^{4}f_k(\omega_k) \quad (12)$$

The attitude dynamics is obtained through feedback linearization facilitated by 123 Tayt-Brian sequence (Canuto, 2013a) and holds

$$\begin{bmatrix}\dot{q}_x\\ \dot{q}_y\end{bmatrix}(t)=\begin{bmatrix}\Omega_x\\ \Omega_y\end{bmatrix}(t)$$
$$\begin{bmatrix}\dot{\Omega}_x\\ \dot{\Omega}_y\end{bmatrix}(t)=P(\boldsymbol{\theta})B_{mxy}\mathbf{f}(t)+\mathbf{d}_m(t), \quad (13)$$
$$P(\boldsymbol{\theta})=\begin{bmatrix}\frac{c_\psi}{c_\varphi} & \frac{s_\psi}{c_\theta}+c_\psi t_\varphi t_\theta\\ -\frac{s_\psi}{c_\varphi} & \frac{c_\psi}{c_\theta}-s_\psi t_\varphi t_\theta\end{bmatrix}^{-1}$$

where the new angular rates are defined as

$$\begin{bmatrix}\Omega_x\\ \Omega_y\end{bmatrix}=\begin{bmatrix}\cos\varphi & -\sin\varphi\sin\theta\\ 0 & \cos\theta\end{bmatrix}Z(\psi)\begin{bmatrix}\omega_x\\ \omega_y\end{bmatrix} \quad (14)$$

Spin equation is straightforward. Cross couplings are treated as known/unknown disturbances.

3. EMBEDDED MODEL

The embedded model is a discrete-time version of equations (10), (12) and (13). The embedded models are the core of the control units and they are stabilized by the noise estimator in charge of estimating the noise vectors.

3.1 Vertical dynamics

A key feature of the EMC is to include a specific disturbance dynamics in order to estimate the unknown disturbances and then to reject them. In this case disturbances only affect acceleration. A first-order disturbance dynamics is used here for simplicity's sake. Vertical equations is written by adding two noise components w_{1z}, w_{2z} that drive the disturbance and controllable dynamics and must be estimated by the noise estimator:

$$\left[\begin{array}{c}z\\ v_z\\ \hline z_d\end{array}\right](i+1)=\left[\begin{array}{cc:c}1 & 1 & 0\\ 0 & 1 & 1\\ \hline 0 & 0 & 1\end{array}\right]\left[\begin{array}{c}z\\ v_z\\ \hline z_d\end{array}\right](i)+\left[\begin{array}{c}0\\ 1\\ \hline 0\end{array}\right]u_z(i)+$$
$$+\left[\begin{array}{cc}0 & 0\\ 1 & 0\\ \hline 0 & 1\end{array}\right]\begin{bmatrix}w_{1z}\\ w_{2z}\end{bmatrix} \quad (15)$$

1.3 Horizontal dynamics

Horizontal dynamics combine (10) and (13) in two decouple equations. Only the first component is reported.

$$\begin{bmatrix}x\\ v_x\\ q_x\\ \Omega_x\\ x_d\\ x_q\end{bmatrix}(i+1)=\begin{bmatrix}1 & 1 & 0 & 0 & 0 & 0\\ 0 & 1 & \alpha(i) & 0 & 1 & 0\\ 0 & 0 & 1 & 1 & 0 & 0\\ 0 & 0 & 0 & 1 & 0 & 1\\ 0 & 0 & 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix}x\\ v_x\\ q_x\\ \Omega_x\\ x_d\\ x_q\end{bmatrix}(i)$$
$$+\begin{bmatrix}0\\ 0\\ 0\\ 1\\ 0\\ 0\end{bmatrix}u_x(k)+\begin{bmatrix}0 & 0 & 0 & 0\\ 1 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix}w_{x1}\\ w_{x2}\\ w_{x3}\\ w_{x4}\end{bmatrix} \quad (16)$$

The parameter $\alpha(i)$ corresponds to a in (12) and holds

$$\alpha(i)=a(i)T^2 \quad (17)$$

where T is the control time unit.

4. GUIDANCE

4.1 Vertical guidance

The vertical guidance implements a standard minimum time control to reach a planned altitude. Maximum acceleration and velocity are accounted for. Only typical profiles are shown in Fig. 2. The desired altitude is 30 m and then a descent down to 15 m is requested.

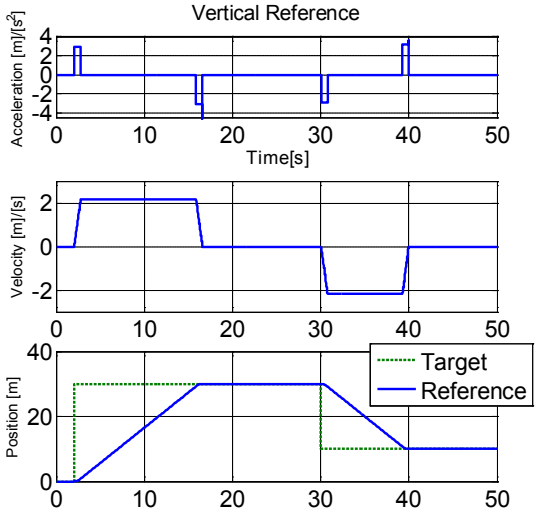

Fig. 2. Vertical reference profiles.

1.4 Horizontal guidance

To obtain smooth angular rates and accelerations, a polynomial guidance is implemented, which is typical of landing space vehicles (Canuto et al., 2013b). In order to avoid abrupt changes in the attitude model a polynomial strategy is used in the horizontal reference generator.

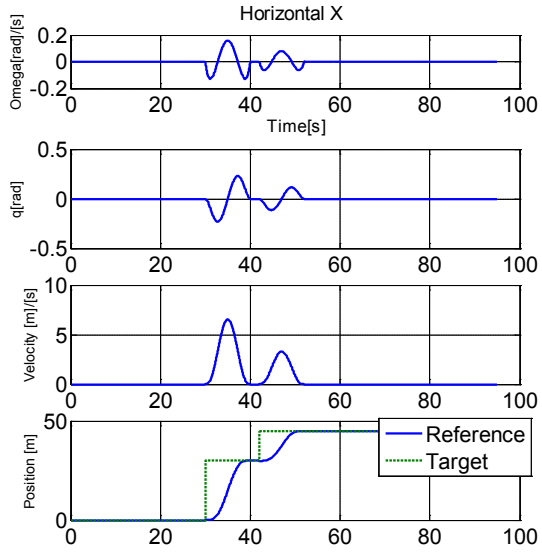

Fig. 3. Horizontal reference profiles.

The following details refer to a single axis. Assuming $\alpha(i)$ to be known at each time from vertical guidance, the model can be considered a chain of discrete time integrators. To simplify analysis the reference generator is detailed in continues time and then discretized. Since the states must follow a polynomial profile the command must be

$$u_x(t) = \begin{bmatrix} 1 & t & t^2 & t^3 \end{bmatrix} \mathbf{a}, \tag{18}$$

where $\mathbf{a}$ is a coefficient vector that depends on target and initial conditions. The state trajectory results as follows

$$\mathbf{x}(t) = \begin{bmatrix} \Omega \\ q \\ v \\ x \end{bmatrix}(t) = \begin{bmatrix} \Omega(0) \\ q(0) + \Omega(0)t \\ v(0) + q(0)t + \Omega(0)t^2/2 \\ x(0) + v(0)t + \frac{q(0)t^2}{2} + \frac{\Omega(0)t^3}{6} \end{bmatrix} + \begin{bmatrix} t & t^2/2 & t^3/3 & t^4/4 \\ t^2/2 & t^3/6 & t^4/12 & t^5/20 \\ t^3/6 & t^4/24 & t^5/60 & t^6/120 \\ t^4/24 & t^5/120 & t^6/360 & t^7/840 \end{bmatrix} \mathbf{a} \tag{19}$$

and can be rewritten as

$$\mathbf{x}(t) = \mathbf{x}_0(t) + \begin{bmatrix} t & 0 & 0 & 0 \\ 0 & t^2/2 & 0 & 0 \\ 0 & 0 & t^3/6 & 0 \\ 0 & 0 & 0 & t^4/24 \end{bmatrix} \times \begin{bmatrix} 1 & 1/2 & 1/3 & 1/4 \\ 1 & 1/3 & 1/6 & 1/10 \\ 1 & 1/4 & 1/10 & 1/20 \\ 1 & 1/5 & 1/15 & 1/35 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & t & 0 & 0 \\ 0 & 0 & t^2 & 0 \\ 0 & 0 & 0 & t^3 \end{bmatrix} \mathbf{a}, \tag{20}$$

and

$$\mathbf{x}(t) = \mathbf{x}_0(t) + T_1 A T_2 \mathbf{a} \tag{21}$$

Equation (21) is solved to obtain the unknown vector $\mathbf{a}$

$$\mathbf{a} = T_2^{-1} A^{-1} T_1^{-1} \left(\mathbf{x}(t) - \mathbf{x}_0(t) \right) \tag{22}$$

The above solution can be easily converted to discrete time.

Fig. 3 shows a solution of the above guidance algorithm. The operator firstly demands a horizontal displacement of 30 m, and then another displacement of 15 m. In order to avoid command saturation, an appropriate maneuver time is selected, this allows relaxing thrust constraints. Fig. 3 clearly shows the sequence of derivatives from horizontal displacement (bottom) back to angular acceleration (top)

5. NAVIGATION

Since all the state variables are not available (for instance the disturbance terms to be rejected) state estimation is mandatory and facilitated by the embedded model. We are using the following sensors, thinking to outdoor flight: IMU, GPS, ultrasound sensor and magnetometer (Cook, 2011). For indoor flight we are not using GPS. The navigation is organized in to different state predictors, consisting of the embedded model that is fed back by noise estimator, driven by the model error (measurement minus model output). The noise estimators are analyzed in a discrete time. All state units are meters.

The model error is the sole accessible measure of the uncertainty. Its current value summarizes the past discrepancies that have not been saved in the embedded model. The model error can be elaborated and accumulated in disturbance states. The residual discrepancies are used to reduce the model error which is must brought to be bounded (internal stability).

1.5 Vertical navigation

For the vertical navigation the position model error is defined as

$$e_z = y_z - \hat{z}, \tag{23}$$

where y_z provided either by GPS and ultrasound sensor. The embedded model variables are marked with a hat to signify that they are instantiated by the measurements. Their fusion is not treated here. The model error is used to close the loop through the gains l_{z0}, l_{z1}. To guarantee stability a dynamic filter with state $p(i)$ which is driven by the model error is added. The following equation shows the navigation discrete dynamic.

$$\begin{bmatrix} p \\ \hat{z} \\ \hat{v}_z \\ \hat{z}_d \end{bmatrix}(i+1) = \begin{bmatrix} 1-\beta_z & 0 & -1 & 0 \\ 0 & 1 & 1 & 0 \\ m_{z0} & -l_{z0} & 1 & 1 \\ m_{z0} & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p \\ \hat{z} \\ \hat{v}_z \\ \hat{z}_d \end{bmatrix}(i) + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} u_z(i) + \begin{bmatrix} 1 \\ 0 \\ l_{z0} \\ 0 \end{bmatrix} y_z(i) \quad (24)$$

The gains $l_{z0}, \beta_z, m_{z0}, m_{z1}$ are computed by fixing the closed-loop eigenvalues. The above state predictor is actually very simplified, since it does not use the accelerometer measurements. Their fusion is not treated here.

5.1 Horizontal navigation

The horizontal navigation uses the embedded model (16) with a fourth-order controllable dynamics. A decoupled strategy is implemented to estimate the horizontal noise vector. The gyro measurements allow to estimate the attitude noise components. The GPS (fused with accelerometer measurements) allow to estimate the horizontal noise vector. When hovering the attitude (pitch and roll) is obtained using lateral accelerometers. When moving attitude is improved using magnetometers.

The state predictor equation assumes that the attitude (pitch and roll) and the angular rate (gyro) are measured. Actually a multi-rate estimator is implemented to account for the lower-rate magnetometer measurements.

$$\begin{bmatrix} \hat{q}_x \\ \hat{\Omega}_x \\ \hat{d}_q \end{bmatrix}(i+1) = \begin{bmatrix} 1 & 1 & 0 \\ -m_0 & 1-l_0 & 1 \\ -m_1 & 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{q}_x \\ \hat{\Omega}_x \\ \hat{d}_q \end{bmatrix}(i) + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} u_x(i) + \begin{bmatrix} 0 & 0 \\ m_0 & l_0 \\ m_1 & 0 \end{bmatrix} \begin{bmatrix} y_g \\ y_q \end{bmatrix}(i) \quad (25)$$

The horizontal position state predictor is very similar to (24) since it requires a dynamic feedback.

$$\begin{bmatrix} p_x \\ \hat{x} \\ \hat{v}_x \\ \hat{x}_d \end{bmatrix}(i+1) = \begin{bmatrix} 1-\beta_x & 0 & -1 & 0 \\ 0 & 1 & 1 & 0 \\ m_{x0} & -l_{x0} & 1 & 1 \\ m_{x0} & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_x \\ \hat{x} \\ \hat{v}_x \\ \hat{x}_d \end{bmatrix}(i) + \begin{bmatrix} 0 \\ 0 \\ \alpha(i) \\ 0 \end{bmatrix} \hat{q}_x(i) + \begin{bmatrix} 1 \\ 0 \\ l_{x0} \\ 0 \end{bmatrix} y_x(i) \quad (26)$$

Fig. 4 shows the horizontal position model error of the state predictor (26) during a test where the operator demands a displacement of 30m (see Fig. 3).

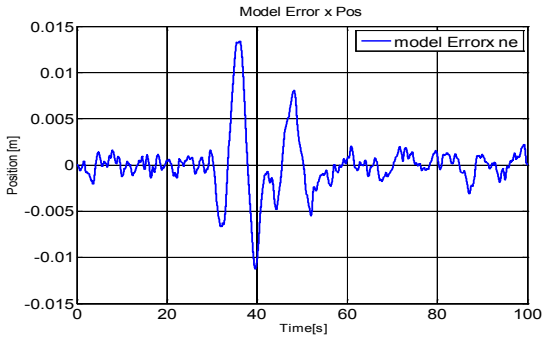

Fig. 4. Position model error

6. CONTROL LAW

1.6 Vertical control law

The CoM control law is computed in local frame coordinates. The command $u_z(i)$ is divided in three components, the reference command and the feedback state command.

$$u_z(i) = \underline{u}_z(i) + k_z(\underline{z} - \hat{z}) + k_v(\underline{v}_z - \hat{v}_z) - \hat{z}_d(i) \quad (27)$$

The feedback gains k_z, k_v are computed to guarantee internal stability of the embedded model. We assume that the overall stability of the vehicle is guaranteed by the state predictor eigenvalues, in charge of cancelling from the control law the effect of neglected dynamics (thruster dynamics, sensor dynamics) when they are not modeled (for instance through a delay).

1.7 Horizontal control law

The horizontal control law follows the same strategy of vertical control law, but is more complex since it requires to allocate the estimate disturbance to some tracking error. It is not reported here (see Canuto et al. 2012a)

7. SIMULATED RESULTS

1.8 Simulation parameters

The main parameters of the quadrotor in Fig. 1 are summarized in Table 1.

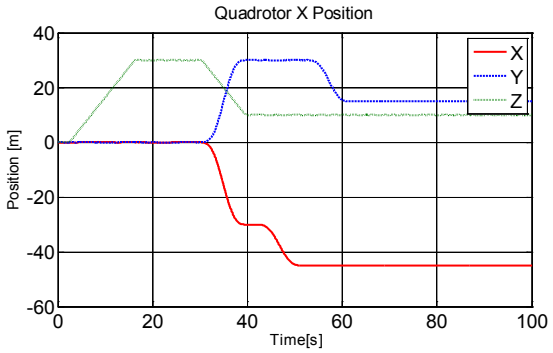

Fig. 5. Simulated quadrotor vertical and horizontal positions

Table 1. Main parameters

No.	Parameter	Unit	Value	Comments
0	Mass	kg	1.49	
1	Inertia	kgm^2	0.017	
2	Inertia	kgm^2	0.018	
3	Inertia	kgm^2	0.027	
4	Diameter	m	0.5	
5	Propeller diameter	m	0.25	
6	Propeller pitch	m	0.18	

The main simulator parameters are listed in Table 2.

Table 2. Main simulation parameters

No.	Parameter	Unit	Value	Comments
0	Time unit	ms	5	
1	Duration	s	100	
2	Control time unit	ms	20	

Fig. 5 shows the vehicle displacement. The reference trajectories are successfully achieved. The true tracking error can be proved to be of the same order of the model error in Fig. 4. The simulated results were obtained by just accelerometer integration (no fusion with GPS). Accelerometer errors were simulated according to data sheet values. Actual errors are worse (a factor of two). Also the attitude was obtained by gyro integration.

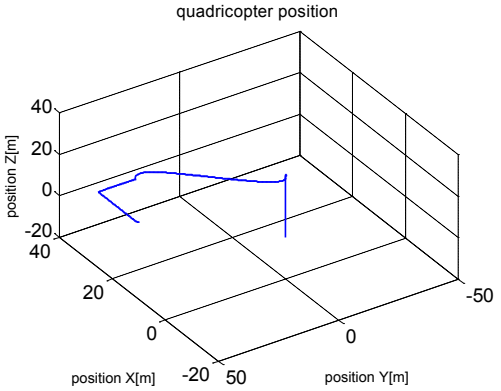

Fig. 6. 3D vehicle motion.

8. CONCLUSIONS

The paper presents the control architecture (Embedded Model Control framework) and some simulated results obtained from a Matlab-Simulink simulator. Control architecture and algorithms are under coding on the target processor.

REFERENCES

Azzam, A. and Wang X. Quad rotor arial robot dynamic modeling and configuration stabilization. *2nd International Asia Conference on Informatics in Control, Automation and Robotics* (CAR). Vol. 1, pp. 438 - 444.

Canuto E., Acuna-Bravo W., Molano A. & Perez C. (2012) Embedded model control calls for disturbance modelling and rejection. *ISA Trans.,* 51 (5), 584-595.

Canuto E and Perez-Montenegro C. (2012) Modelling and control of a small auadrotor for testing propulsive planetary landing guidance, navigation and control. *63th International Astronautica Congress (IAC 2012),* Naples (Italy), 1-5 October 2012.

Canuto E. et al. (2013a) Planetary landing: modelling and control of the propulsion descent, *Zhonguo Kexue Jishu Daxue Xuebao*, 43 (1), 1-14.

Canuto E., Molano Jimenez A, and Perez Montenegro C. (2013b), Guidance and control for the propulsion phase of planetary landing, *Preprints of the 19th IFAC Symposium on Automatic Control in Aerospace*, Wuerzburg (Germany), 2-6 September 2013, pp. 360-365

Cook G. (2011), *Mobile robots. Navigation, Control and Remote Sensing*. John Wiley & Sons.

Dikmen İ, Arīsoy A and Temeltaş H Attitude control of a quadrotor *Int. Conf on Recent Advances in Space Technologies*, RAST '09. 4th . pp. 722 – 727.

Guerrero J.A. and Castillo P. (2011), Trajectory tracking for a group of mini rotorcraft flying in formation, in *Preprints 18th IFAC Word Congress,* 2011, Milan (Italy), pp. 6331-6336.

Hoffmann F., Goddemeier N. and Bertram T. (2010), Attitude estimation and control of a quadrocopter. *IEEE Int. Conf on intelligent Robots and Systems (IROS),* pp. 1072 - 1077.

Jaimes A., Kota S. and Gomez J. (2008) An approach to surveillance an area using swarm of fixed wing and quad-rotor unmanned aerial vehicles. *IEEE Int. Conf on System of Systems Engineering*, pp. 1 - 6 .

Martin P. and Salaun E (2010),The true role of accelerometer feedback in quadrotor control.

Ping, J.T.K (2012) Design of a easy to fly, low-cost generic unmanned aerial vehicle (UAV) for civilian aerial-imaging application. *IEEE Int. Conf on Sustainable Utilization and Development in Engineering and Technology* (STUDENT), pp. 283 - 288.

Roskam J. and Lan C. E. (2003) *Airplane Aerodynamics and Performance*, Design, Analysis and Research Corporation (DARcorporation), ISBN 1-884885-44-6.

Seddon J and Newman, S. (2011) *Basic Helicopter Aerodynamics* – 3rd ed, John Wiley & Sons, Ltd, 2011.

Slotine J-J.E. & Li W. (1991) *Applied nonlinear control*. Prentice-Hall Int. Ed.

Tayebi A and McGilvray S. Attitude stabilization of a four-rotor aerial robot. *43rd IEEE Conference on Decision and Control*, 2004. CDC Vol. 2, pp. 1216 – 1221

Tewari A. (2011) *advanced control of aircraft, spacecraft and rockets*. John Wiley & Sons, Ltd,.

Yanushevsky R. (2011) *Guidance of unmanned aerial vehicles*. CRC Press, Taylor & Francis Group.