Implementation and Testing of a Backstepping Controller Autopilot for Fixed-wing UAVs

Journal of Intelligent & Robotic Systems manuscript No.
(will be inserted by the editor)

Implementation and Testing of a Backstepping Controller Autopilot for Fixed-wing UAVs

Daniele Sartori · Fulvia Quagliotti · Matthew J. Rutherford · Kimon P. Valavanis

Received: date / Accepted: date

Abstract The ability of backstepping controllers to deal with nonlinearities make this technique a suitable candidate for the control of small fixed-wing Unmanned Aerial Vehicles (UAVs). The authors have already proposed a comprehensive approach combining backstepping with PID controllers for simultaneous longitudinal and latero-directional control of fixed-wing UAVs, achieving good performance even with considerable levels of signal noise [1]. In further detail, the ability of the mixed approach to control different size and configuration aircraft in the presence of parametric uncertainties or noise, and when implemented on a microcontroller board was demonstrated. The present paper illustrates integration and testing of the backstepping controller on a real unmanned aircraft. After a summarizing the adopted control design and strategy, initial software and hardware simulations validate the control action for the selected aircraft. The implementation of the microcontroller on the aircraft and the integration with other aircraft systems is also illustrated. Experimental results obtained for ground and flight tests are presented, validating the applicability of the backstepping controller.

Keywords Backstepping · Nonlinear control · UAV · Autopilot · Flight test

Daniele Sartori
Department of Mechanical and Aerospace Engineering, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy
E-mail: daniele.sartori@polito.it

Fulvia Quagliotti
Department of Mechanical and Aerospace Engineering, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy
E-mail: fulvia.quagliotti@polito.it

Matthew J. Rutherford
Department of Computer Science, University of Denver, John Greene Hall, 2360 S. Gaylord St., Denver, CO 80208, USA
E-mail: mjr@cs.du.edu

Kimon P. Valavanis
Department of Electrical & Computer Engineering, University of Denver, Knudson Hall, 2390 S. York St., Denver, CO 80208, USA
E-mail: kvalavan@du.edu

1 Introduction

Backstepping is one of the most promising advanced control laws for small fixed-wing unmanned aircraft. Small UAVs flight dynamics are characterized by highly nonlinear behavior, a severe cross-coupling between longitudinal and latero-directional dynamics, and a considerable sensitivity to external disturbances. The main benefit of the backstepping controller is its ability to deal with nonlinearities. Differently from traditional linear control techniques such as LQ or feedback linearization, the application of a nonlinear control law to a highly nonlinear UAV aircraft offers satisfying performance over a large flight envelope [2]. Useful nonlinearities are maintained and additional nonlinear damping terms can be introduced to increase robustness to model errors or to improve transient performance [3]. This approach, in fact, is different from feedback linearization where these forces are first modeled and then canceled. The nonlinear backstepping approach allows for less accurate knowledge of the aircraft dynamics. Furthermore, as backstepping belongs to the Lyapunov family of controllers it guarantees convergence of the tracking error and asymptotic stability [4].

The growing interest in applications of fixed-wing unmanned platforms to various civilian applications is strongly stimulating design and development of reliable and robust on-board controllers. The miniaturization and reduction in cost of microcontrollers, coupled with improved performance, see [5] and [6], enables researchers to actually fly unmanned aircraft driven by self-developed control laws. Recent surveys by Chao, Cao and Chen [7] and Ollero and Merino [8] illustrate the state of the art of autopilot systems and control laws. The use of PIDs is still popular as they guarantee simple implementation and low computational effort, and the designer has adequate control over the system response and clear understanding of the control action. Nevertheless, more advanced techniques such as $\mathcal{L}_1$ [9] or receding horizon [10] controllers are slowly being introduced into flight.

In this context, backstepping has been the object of many theoretical projects illustrating its application to fixed-wing aircraft flight control. For instance, [11] presents an adaptive backstepping control law for angle of attack tracking and [12] applies adaptive backstepping to path-following through roll angle control. These are examples where longitudinal and latero-directional controls are independent, the number of works describing combined longitudinal and latero-directional control is instead limited. In [13] incidence, sideslip and roll angles, are controlled by adaptive backstepping with neural networks through body-axes angular rates. In [14], constrained adaptive backstepping with neural adaptation laws is employed for tracking angle of attack, stability-axes roll rate and total velocity while sideslip is maintained at zero. The literature shows that combining backstepping with complex adaptive laws is a common approach that guarantees satisfying theoretical results. Nevertheless, the constraints imposed by real-time implementation limit the possibility to test these algorithms in flight on a real small scale UAV. To the authors' knowledge the only flying application of adaptive backstepping on fixed-wing unmanned aircraft is presented in [15], while the algorithm described in [16] is currently being implemented.

In already reported research [1], the authors presented a nonlinear backstepping controller capable of guaranteeing tracking control of longitudinal and latero-directional aircraft states. Inner-loop variables (angle of attack, sideslip angle and stability-axes roll rate) were controlled via the backstepping approach described in

[17] and, with more details, in [18]. Less dynamic outer-loop variables (velocity, altitude and heading) were controlled by PID gains. The main purpose of that work being the realization of a starting framework for the applied use of backstepping control on small UAVs, there was no discussion about adaptation with advanced outer-loop design. However, it was demonstrated that the proposed method was able to control aircraft different in size and configuration, in the presence of noise and parametric uncertainties and when implemented using a microcontroller. The present paper illustrates the following step in the controller development: the integration and the testing of the control strategy on a real aircraft. Software and hardware simulations with an accurate model of the adopted aircraft validate the approach and allow the tuning of the controller. The onboard installation of the microcontroller is described, the problems encountered and the proposed solutions are illustrated. Finally, preliminary ground and flight tests results are described.

The paper is organized as follows: Section II briefly summarizes the control design and strategy, Section III presents the aircraft and sensors model. Section IV illustrate the results of software and hardware simulations while Section V describes the procedure for integrating the controller on the aircraft. Section VI shows the results of preliminary ground and flight tests and, finally, Section VII concludes and describes the future work.

2 Backstepping controller design and strategy

The theoretical framework for the design procedure of the proposed backstepping approach has been presented in [1]. However, for clarification purposes, background information is provided next, which leads to the implementation details.

Backstepping is a form of recursive control stabilizing in cascade all the states of the considered system. The design of a backstepping controller requires that the equations describing the controlled system may assume the general triangular structure called pure-feedback form [3]:

$$\begin{cases} \dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})\xi_1 \\ \dot{\xi}_1 = f_1(\mathbf{x}, \xi_1, \xi_2) \\ \dot{\xi}_2 = f_2(\mathbf{x}, \xi_1, \xi_2, \xi_3) \\ \vdots \\ \dot{\xi}_{k-1} = f_{k-1}(\mathbf{x}, \xi_1, ..., \xi_k) \\ \dot{\xi}_k = f_k(\mathbf{x}, \xi_1, ..., \xi_k, u_b) \end{cases} \tag{1}$$

In (1) $\mathbf{x} \in \mathbb{R}^n$ is the state vector and $\xi_{1,...,k}$ are scalars denoting other states of the system. The functions f_i $(i = 1, ..., k)$ are nonlinear and depend only on $\mathbf{x}$ and on the states ξ_j $(j = 1, ..., i+1)$, i.e., they depend at most on the state variable of the upper order subsystem. The scalar u_b is the external controller of the global system; each subsystem represented by the state ξ_l $(l = 1, ..., k-1)$ is controlled by the virtual control input ξ_{l+1}.

In this work fixed-wing aircraft dynamics are considered, they are described by three vectorial differential equations: force equation, moment equation and attitude equation [19]. The manipulation of the equations of motion allows the building of the desired cascade form under some assumptions and for a limited

number of states: angle of attack α, sideslip angle β and stability-axes roll rate p_s, see Fig. 1. The aim is to design a controller so that $\alpha = \alpha^{ref}$, $p_s = p_s^{ref}$ and $\beta = 0$.

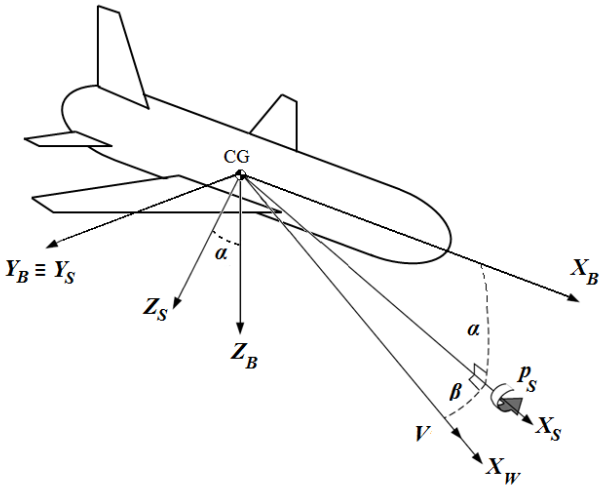

Fig. 1 Controlled variables and reference axes at t_0

The idea behind the proposed backstepping approach is to directly control the stability-axes angular velocities $\boldsymbol{\omega}_s = (p_s, q_s, r_s)^T$ so that, in cascade, α and β follow the reference signal. This direct control is achieved through the relationship $\dot{\boldsymbol{\omega}}_s = \mathbf{u}_c$, where $\mathbf{u}_c = (u_1, u_2, u_3)^T$ is the control vector. The dynamics of α and β are obtained from the force equation written in wind axes, the complete derivation is available in [18]. The combination of this formulation with the control definition results in the following set of equations:

$$\begin{cases} \dot{p}_s = u_1 \\ \dot{\alpha} = q_s - p_s \tan\beta + \dfrac{-Lift - T\sin\alpha + mg_2}{mV\cos\beta} \\ \dot{q}_s = u_2 \\ \dot{\beta} = -r_s + \dfrac{Y - T\cos\alpha\sin\beta + mg_3}{mV} \\ \dot{r}_s = u_3 \end{cases} \tag{2}$$

where $Lift$ is the aircraft lift, T is the engine thrust, Y is the aerodynamic force in body axes. V is the magnitude of the total velocity vector and m is the aircraft mass. The gravity acceleration components g_2 and g_3 are:

$$\begin{aligned} g_2 &= g(\cos\alpha\cos\theta\cos\phi + \sin\alpha\sin\theta) \\ g_3 &= g(\cos\beta\cos\theta\sin\phi + \sin\beta\cos\alpha\sin\theta - \sin\alpha\sin\beta\cos\theta\cos\phi) \end{aligned} \tag{3}$$

where $g = 9.81$ m/s^2 is the gravity acceleration. The angles ϕ and θ are called, respectively, roll and pitch. Together with the yaw angle ψ they represent the

attitude of the aircraft. The lift force depends on the angle of attack through the $C_L = C_L(\alpha)$ coefficient and the side force Y on the sideslip angle through the $C_Y = C_Y(\beta)$ coefficient. The thrust T is considered independent from aerodynamic angles

The equations of motion in (2) have been derived based on the following assumptions:

- **Assumption 1**: The deflection of control surfaces only generates a variation in moments, the variation in forces is small enough to be neglected.
- **Assumption 2**: Lift and side force coefficients, C_L and C_Y, only depend on aerodynamic angles and not on aerodynamic angle rates of change: $C_L = C_L(\alpha)$, $C_Y = C_Y(\beta)$.

The first assumption is reasonable for aircraft with traditional configuration, so that control surfaces are far from the aircraft center of gravity [19]. Assumption 2 is considered valid in steady flight or during smooth maneuvers when the delay in the pressure distribution to adjust to sudden attitude variation is large.

Two additional assumptions are introduced to simplify the controller design:

- **Assumption 3**: The time derivatives of speed V, altitude h and heading ψ can be neglected as they have a slower rate of change compared to the controlled variables α, β, and p_s.
- **Assumption 4**: Actuators have rapid enough dynamics, thus, they can be ignored in the design process.

Assumption 3 is mainly valid for progressive maneuvers where a controlled variation in the aircraft equilibrium has a primary effect on the faster attitude dynamics and a secondary effect on the slow-changing navigation variables. Assumption 4 is very common and generally reasonable, provided that Assumptions 2 and 3 are respected.

The application of a single backstepping controller stabilizing all the dynamics of (2) is not possible because the cascade form is not respected. However by separating the dynamics as:

$$\dot{p}_s = u_1 \tag{4}$$

$$\begin{cases} \dot{\alpha} = q_s - p_s \tan\beta + \dfrac{-Lift - T\sin\alpha + mg_2}{mV\cos\beta} \\ \dot{q}_s = u_2 \end{cases} \tag{5}$$

$$\begin{cases} \dot{\beta} = -r_s + \dfrac{Y - T\cos\alpha\sin\beta + mg_3}{mV} \\ \dot{r}_s = u_3 \end{cases} \tag{6}$$

three sub-controllers stabilizing the desired states α, β and p_s can be defined. These control laws are designed for a simultaneous action on these three variables, also taking into account cross-coupling effects. In fact, it is possible to observe the presence of p_s and β in the α dynamics and, at the same time, the presence of α in the β dynamics. Because of this coupling, the computation of a control action needs to consider, at each moment, the value of the state controlled by another control action. For instance, the control law defining u_2 is evaluated with the instantaneous value of p_s, controlled by u_1, and β controlled by u_3. This occurrence is beneficial when dealing with maneuvers where strong coupling between longitudinal and latero-directional planes exists.

A simple proportional controller is chosen for p_s, (4), while the cascade form of (5) and (6) is suitable for the application of a backstepping controller for α and β. Note that (5) and (6) have similar structure:

$$\begin{cases} \dot{\omega}_1 = f(\omega_1, y) + \omega_2 \\ \dot{\omega}_2 = u_s \end{cases} \tag{7}$$

The design of a single backstepping controller for (7) is applicable to (5) and (6). A change of variables is performed to have the origin as desired equilibrium point:

$$\begin{aligned} x_1 &= \omega_1 - H \\ x_2 &= \omega_2 + f(H, y) \\ \Omega(x_1) &= f(x_1 + H, y) - f(H, y) \end{aligned}$$

where H is the reference value for the controlled variable. The resulting dynamics are:

$$\begin{cases} \dot{x}_1 = \Omega(x_1) + x_2 \\ \dot{x}_2 = u_s \end{cases} \tag{8}$$

The external control input u_s controls x_2 that, in cascade, acts as virtual control to stabilize x_1. Table 1 summarizes the relationships between the variables used in the new and in the original systems. The functions $f_\alpha(\alpha, y_\alpha)$ and $f_\beta(\beta, y_\beta)$ are:

$$\begin{aligned} f_\alpha(\alpha, y_\alpha) &= -p_s \tan\beta + \frac{-Lift - T\sin\alpha + mg_2}{mV\cos\beta} \\ f_\beta(\beta, y_\beta) &= \frac{Y - T\cos\alpha\sin\beta + mg_3}{mV} \end{aligned}$$

Table 1: Change of variable relationships

General system	Longitudinal	Latero-directional
ω_1	α	β
ω_2	q_s	$-r_s$
u_s	u_2	$-u_3$
y	$p_s, \beta, V, h, \theta, \phi$	$\alpha, V, h, \theta, \phi$
$f(\omega_1, y)$	$f_\alpha(\alpha, y_\alpha)$	$f_\beta(\beta, y_\beta)$
H	α^{ref}	0
x_1	$\alpha - \alpha^{ref}$	β
x_2	$q_s + f_\alpha(\alpha^{ref}, y_\alpha)$	$-r_s + f_\beta(0, y_\beta)$
$\Omega(x_1)$	$f_\alpha(\alpha, y_\alpha) - f_\alpha(\alpha^{ref}, y_\alpha)$	$f_\beta(\beta, y_\beta) - f_\beta(0, y_\beta)$

As fully demonstrated in [18] through Lyapunov stability theory, a simple globally stabilizing control law for the system of (8) is the linear control:

$$u_s = -k_2(x_2 + k_1 x_1)$$

with $k_2 > k_1 > \max\{0, k_u\}$. Using the relationships of Table 1, the control laws for the systems of (5) and (6) are obtained:

$$\begin{aligned} u_2 &= -k_{\alpha,2}\left(q_s + k_{\alpha,1}\left(\alpha - \alpha^{ref}\right) + f_\alpha(\alpha^{ref}, y_\alpha)\right) \\ u_3 &= k_{\beta,2}\left(-r_s + k_{\beta,1}\beta + f_\beta(0, y_\beta)\right) \end{aligned} \tag{9}$$

with:

$$\begin{aligned} k_{\alpha,2} &> 2k_{\alpha,1}, \quad k_{\alpha,1} > \max\{0, k_\alpha\} \\ k_{\beta,2} &> 2k_{\beta,1}, \quad k_{\beta,1} > \max\{0, k_\beta\} \end{aligned} \tag{10}$$

where:

$$\begin{aligned} k_\alpha &= \max_{\alpha, y_\alpha} \frac{\partial f_\alpha(\alpha, y_\alpha)}{\partial \alpha} \\ k_\beta &= \max_{\beta, y_\beta} \frac{f_\beta(\beta, y_\beta) - f_\beta(0, y_\beta)}{\beta} \end{aligned}$$

Finally, a proportional control is adopted for p_s:

$$u_1 = k_{p_s}\left(p_s^{ref} - p_s\right), \quad k_{ps} > 0 \tag{11}$$

The relation between control inputs and stability-axes angular accelerations is defined by $\mathbf{u}_c = (u_1, u_2, u_3)^T = \dot{\boldsymbol{\omega}}_s$. Angular accelerations are the result of the variation in moments originated primarily by the deflection of aircraft control surfaces. The vector of deflections $\boldsymbol{\delta}$ is obtained from the moment equation:

$$\mathbf{M}(\boldsymbol{\delta}) = I\left(R_{sb}^T \mathbf{u_c} + \dot{R}_{sb}^T \boldsymbol{\omega}_s\right) + \boldsymbol{\omega} \times I\boldsymbol{\omega} \tag{12}$$

To calculate $\boldsymbol{\delta}$, a control strategy matching the controlled variables with the aircraft control surfaces must be defined.

The proposed backstepping approach is designed to control the attitude of the aircraft through three variables α, β and p_s. In order to achieve an autopilot configuration capable of navigating the aircraft on a defined path the control of speed V, altitude h and heading ψ is required. Consistent with Assumption 3, these outer loop variables are characterized by slower dynamics compared to inner loop attitude states and they can be effectively handled with traditional PID technique. PID gains are tuned manually following a trial and error approach. The goal is optimizing the response in terms of overshoot, rise time, settling time and ringing compatible with aircraft behavior.

The PID controlling the speed feeds the backstepping controller with the desired angle of attack, while the PID controlling the heading defines the desired roll rate. The control surfaces employed are the elevator δ_e, the aileron δ_a and the rudder δ_r. According to Assumption 1 these only generate a variation in moments and not in forces. The deflection vector $\boldsymbol{\delta} = (\delta_e, \delta_a, \delta_r)^T$ is obtained substituting in (12) the most general expressions of the moments:

$$\begin{aligned} L &= L(\delta_a, \delta_e, \delta_r) \\ M &= M(\delta_a, \delta_e, \delta_r) \\ N &= N(\delta_a, \delta_e, \delta_r) \end{aligned} \tag{13}$$

and solving the resulting linear system with three equations and three unknowns. The expressions for the moments are linear combinations of the unknown surface deflections and of the known states of the aircraft, the coefficients are the known aerodynamic and control derivatives. Note that commonly the contribution of δ_a and δ_r on M is very small or zero, and similarly the effect of δ_e on L and N. In this case the calculation of the commands is more simple: δ_e is found from the $M(\delta_e)$ equation, while δ_a and δ_r are found solving the linear system with $L(\delta_a, \delta_r)$ and $N(\delta_a, \delta_r)$. The engine thrust vector is considered aligned with the aircraft X_B axis, it does not generate moments.

The third PID defines the throttle δ_{th} with a feedback from the altitude, independently from backstepping. As explained in [20], the idea of controlling airspeed with the deflection of a surface, and not with the motor, is a common autopilot approach which guarantees accurate tracking of this sensible parameter. Table 2 summarizes the controlled variables, their command, and the control method.

Table 2: Relationship between variables and commands

Outer-loop	Inner-loop	Command	Control method
V	α	δ_a, $\boldsymbol{\delta_e}$, δ_r	Backstepping + PID
h	–	δ_{th}	PID
ψ	p, β	$\boldsymbol{\delta_a}$, δ_e, $\boldsymbol{\delta_r}$	Backstepping + PID

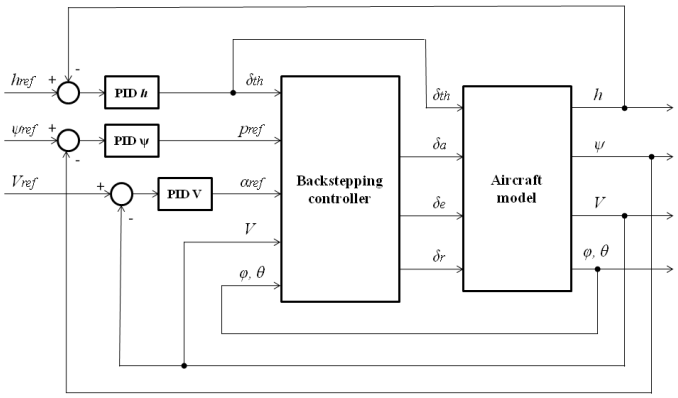

Fig. 2 Backstepping control strategy for fixed-wing aircraft

The control scheme is shown in Fig. 2. The measures of the controlled states are the feedback variables and their differences with the corresponding reference values define the error inputs for the PIDs. The resulting elevator, aileron and throttle control inputs act on the aircraft model. The throttle command and the

measured speed are given as input to the backstepping controller as required by the control law definition and for the estimation of the inner loop states. In fact, α, β and p_s, used for the definition of the inner loop error, are estimated within the backstepping controller with satisfying accuracy integrating Equations (4)-(6) [1]. A support to the accurate estimation of α and β is provided by the feedback of θ and ψ.

3 Aircraft and Sensors Model

The main obstacle in testing a controller is the difference between the theoretical aircraft model and the real-life model. Unmodeled dynamics, high order nonlinear terms, parametric uncertainties and external disturbances can introduce unexpected behaviors affecting the controller performance. The ability of the backstepping controller to deal with nonlinearities is an advantage, but having a reliable and accurate mathematical model of the aircraft is still a necessary requirement. In fact, the possibility to perform an accurate tuning of the controller during simulations allows to foresee the aircraft controlled behavior. This considerably reduces the risk and time of the flight tests.

3.1 Aircraft Model

The aircraft chosen for testing the backstepping controller into flight is the Ultrastick 25e, Fig. 3. The aircraft has traditional configuration, electrical propulsion and control is achieved through elevator, ailerons and rudder. Its characteristics are summarized in Table 3. A traditional mathematical model of the Ultrastick based just on equations and theoretical assumptions, even if very detailed, can hardly reach the level of accuracy required. A model based on experimental tests is preferred. The work described in [21] and [22] deals with this issue, the authors describe the procedure to experimentally identify the model of the Ultrastick 25e aircraft.

Fig. 3 Ultrastick 25e aircraft

Table 3: Ultrastick 25e aircraft characteristics

Parameter	Value
Length	1.05 m
Wingspan	1.27 m
Wing surface	1.27 m^2
Max. weight	1.8 kg
Max. power	500 W
Cruise speed	17 m/s

The procedure consists in a preliminary modeling based on wind tunnel tests, the obtained baseline model is used to define flight tests where system identification is used to build a definitive model. The result of this work is easily accessible [23]. Among other tools on this website, one of the most relevant is a Simulink model of the aircraft. This is available in linear and nonlinear form, it embodies an accurate model of the aircraft dynamics, including trim conditions, actuators transfer functions and electric motor behavior.

3.2 Sensors Model

The backstepping controller acts on three feedback variables, velocity V, altitude h and heading ψ, three sensors able to measure these parameters are thus necessary. Note, in fact, that the inner loop variables α, β and p_s are estimated inside the backstepping controller with good degree of accuracy, as demonstrated in [1]. This unconventional but effective approach is adopted to maintain simple the physical integration of the controller on the aircraft and because of the lack of affordable, reliable and small aerodynamic angles sensors. The feedback of θ and ψ must be provided as well to improve the inner loop states estimation. The noise of the adopted sensors is analyzed and added to the aircraft Simulink model as Gaussian white noise in order to guarantee a high level of reality.

3.2.1 Velocity Sensor

The total velocity V is a key parameter. It strongly influences the aircraft states through is quadratic presence in the forces and moments, it is a navigation variable and, as visible from (2), it directly affects the attitude of the aircraft. Achieving a good measurement of the velocity is essential to achieve a good performance of the controller. The pitot tube method is chosen for its good precision and for its high sampling rate, compatible with the controller frequency action. GPS is dismissed as not able to provide accurate data with high frequency.

The pitot functioning is represented in Fig. 4. A pitot probe collects the total and static pressures, through silicon tubes these are carried to the Freescale Semiconductors MPXV7002 transducer [24]. The transducer outputs an analog voltage proportional to the difference between total and static pressure, this voltage is digitalized through a Linear Technology LTC 1865 16-bit Analog to Digital Converter (ADC) [25]. The ADC is able to sample data with a frequency up to 125 kHz when one channel is used. In this case the sampling frequency is limited to 20 Hz to match the slowest sensor, the barometric pressure sensor. The digital

voltage is read by the XMOS board which first converts the voltage value into pressure difference, as the pressure difference is equal to the dynamic pressure the airspeed is easily retrieved.

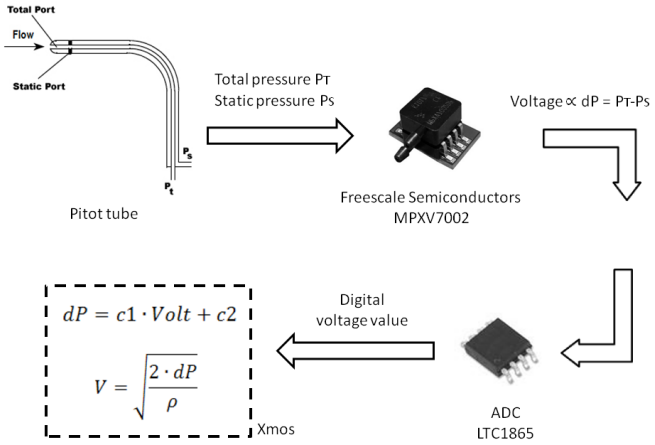

Fig. 4 Pitot airspeed measurement scheme

The law relating the voltage $Volt$ with the pressure difference dP is assumed linear with very limited error and the coefficients $c1$ and $c2$ of Fig. 4 are obtained through wind tunnel calibration, see Fig. 5. As a consequence the relationship between pressure difference and freestream velocity is quadratic. As each recorded time series is characterized by noise, the adoption of a filter is necessary. Further details about the filtering operation is introduced in section 3.2.4.

3.2.2 Altitude Sensor

Altitude measurement is performed through barometric pressure sensor [26] mounted on a breakout board [27]. Considering the data frequency, range and precision required and the limited weight and space budget, other options such as GPS or laser and ultrasonic sensors were excluded. The estimation of altitude h relies on the measure of the local air pressure p through the formula

$$h = 44330 \cdot \left(1 - \left(\frac{p}{p_0}\right)^{0.1903}\right) \tag{14}$$

where p_0 is the equivalent pressure at sea level for the moment and the location considered. This parameter is generally available through weather station websites or can be calculated inverting the formula of (14) in a location with known altitude. In order to be independent from this value that sometimes can be unavailable or changes during the day, the parameter given to the controller will be the height above the airfield (AGL, above ground level altitude) and not the absolute altitude.

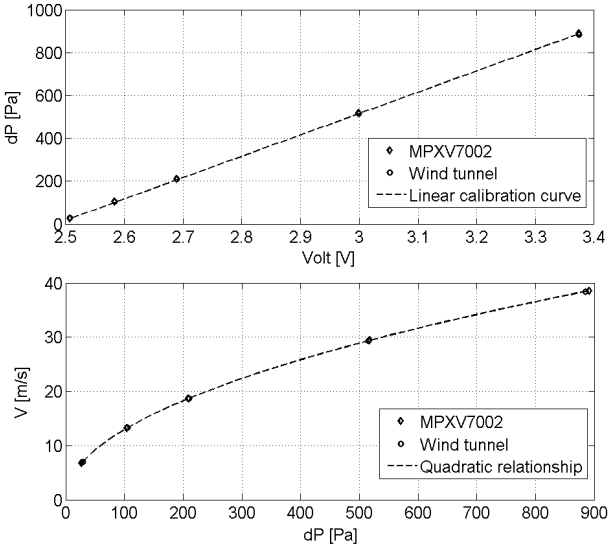

Fig. 5 Pitot calibration curve comparing data from pitot and wind tunnel

A first measurement is performed and stored when the control board is powered on the ground, the AGL is obtained subtracting this value to future measurements.

The barometric pressure sensor has 4 operating modes, each characterized by a combination of sampling frequency and precision: the higher the precision, the lower the sampling frequency. The operating mode characterized by a 0.4 m root mean square noise error and 20 Hz sampling frequency offers the best compromise. Like the pitot, this sensor is affected by noise and its output need to be filtered, more details will be added in Section 3.2.4.

3.2.3 Attitude Sensor

The measurement of the attitude angles ϕ, θ and ψ is performed with the Vectornav VN-100 IMU [28]. This sensor incorporates a 3-axis magnetometer, a 3-axis accelerometer and 3-axis gyroscope with extended Kalman filter. Its outputs include aircraft attitude as Euler angles or quaternions, linear accelerations, angular rates or magnetic local field. In the considered case the angles of interest are obtained from the string defining the aircraft Euler angles. The string is an output of the VN-100 sensor, it is received and analyzed by the XMOS board. For the measurement of ψ, this IMU is preferred to a simple compass because of its high quality of the measurement, the existence of an extended Kalman filter provides a smooth and noise-free output. Similarly to the other sensors the IMU is set to output data with a 20 Hz frequency.

3.2.4 Kalman Filter

The presence of noise in the velocity and altitude measurements forces the inclusion of a filtering operation before feeding these values to the controller. This helps to

prevent an oscillatory response which could be amplified by the presence of a derivative term in the outer loop control. A Kalman filter is thus implemented for the data coming from pitot and barometric pressure sensors. As stated, the IMU already includes an extended Kalman filter. The Kalman filter sequence for each sampling operation i is summarized in (15).

$$\begin{aligned} p_i &= p_{i-1} + Q \\ K &= \frac{p_i}{p_i + R} \\ x_i &= x_{i-1} + K \cdot (\widetilde{x_i} - x_{i-1}) \\ p_i &= (1 - K) \cdot p_i \end{aligned} \tag{15}$$

Here x_i is the filtered variable, $\widetilde{x_i}$ is the measured variable and p_i is the estimated error. K is the Kalman gain, Q is the process noise variance and R is the sensor noise variance. The value of R can be found by experimental tests, the value of Q is more difficult to estimate and it can be used as a parameter for tuning the intensity of the filter action.

Table 4 illustrates the standard deviations σ, with and without Kalman filter, from a mean value indicating the trim flight condition. The improvement achieved with the filter is remarkable. Fig. 6 illustrate an example of filtering action on a randomly varying altitude measurement. It is possible to see how the filter was tuned in order to reduce the oscillations without introducing a considerable delay in the filtered time series.

Table 4: Kalman filter parameters

Measurement	σ	σ_{Kalman}
V	0.17	0.05
h	0.44	0.16

4 Simulations

In [1] it was demonstrated that the proposed backstepping approach is able to control aircraft different in size and configuration also when running in real time on a microcontroller board. In this section preliminary software and hardware simulations are performed in order to validate and calibrate the controller on the Simulink model of the Ultrastick 25e aircraft.

4.1 Software-in-the-loop Simulations

Simulations are obtained merging a backstepping controller Simulink model with the aircraft Simulink model available from the University of Minnesota UAV research group. This approach is defined as software-in-the-loop simulation (SIL). The resultant system is integrated in discrete time with a time step of 0.01 seconds, equal to the controller action frequency, Heun method is used. Sensors noise and

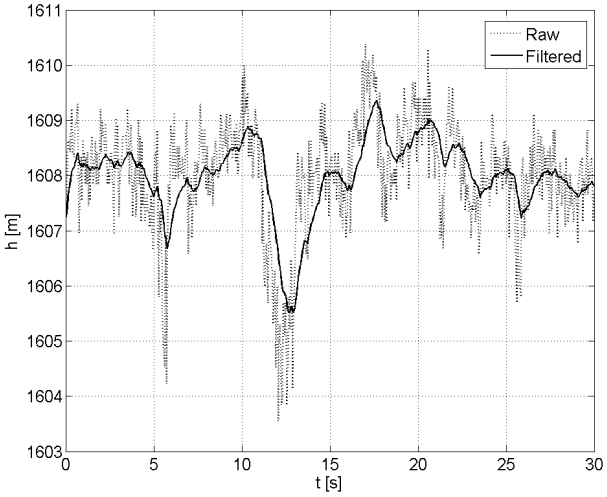

Fig. 6 Kalman filtering action on a noisy altitude measurement

a Kalman filter are modeled as described in the previous sections. The Simulink model is built so that the trim commands are already an input for the aircraft. Because of that, the backstepping controller, instead of calculating the absolute commands, calculates the commands variation to be added to the trim commands.

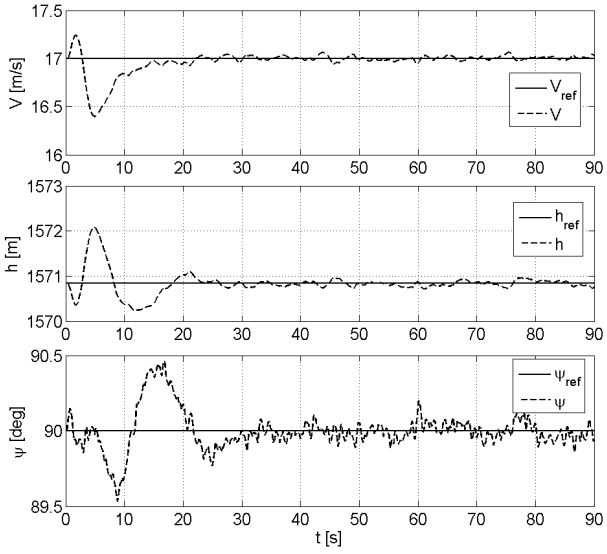

Fig. 7 SIL outer loop response for Ultrastick 25e, trim conditions hold

Cruise conditions are chosen as starting values: V=17 m/s, h=1571 m and ψ=0 deg. The value for h corresponds to the altitude of the Denver research facility where initial tests are performed. For this value of h the air density is around 15% lower than at sea level. Controlled results appear in Fig. 7 and 8. The first simulation asks the aircraft to maintain the cruise conditions. This represents one of the first flight tests that will be performed. As the aircraft is already trimmed for the cruise conditions, the controller only needs to adjust the commands so that they quickly return to trim values after the usual initial oscillations. Fig. 7 shows that the aircraft is successfully brought back to trim conditions. An average time of 20 seconds is necessary, some minor noise-induced oscillations persist.

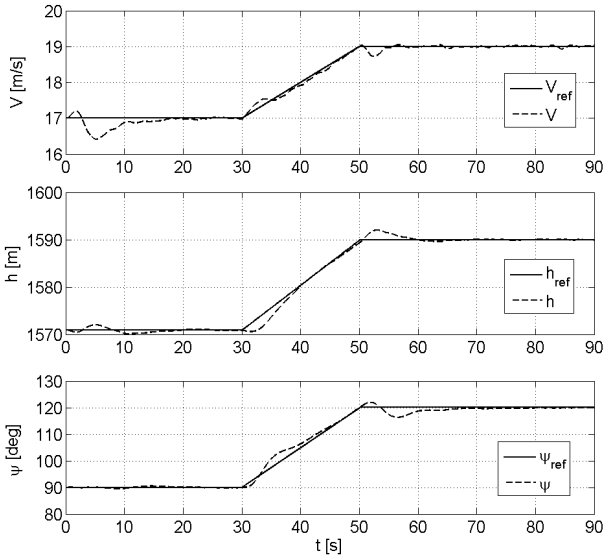

Fig. 8 SIL outer loop response for Ultrastick 25e, maneuver

Fig. 8 shows the results when a coupled maneuver is requested. Small oscillations appear in the velocity, mostly because the derivative gain of the velocity is disturbed by the noise. The altitude response has a minor lag and overshoot when following the reference value during the slope segment, steady state value is successfully achieved. All states, in particular the heading angle ψ, suffer from some mild oscillations at the beginning and at the end of the ramp generated by cross-coupling between longitudinal and latero-directional planes. The corresponding commands are illustrated in Fig. 9. In spite of the filtering action, the noise still has a strong influence on the elevator command because of the derivative gain in the outer loop PID on V. A reduction of this gain would reduce the damping and so generate an oscillatory behavior, the proposed solution is a best compromise between the two conflicting requirements. The throttle is less affected by noise, the derivative gain of the PID on h can be drastically reduced without the onset of oscillations.

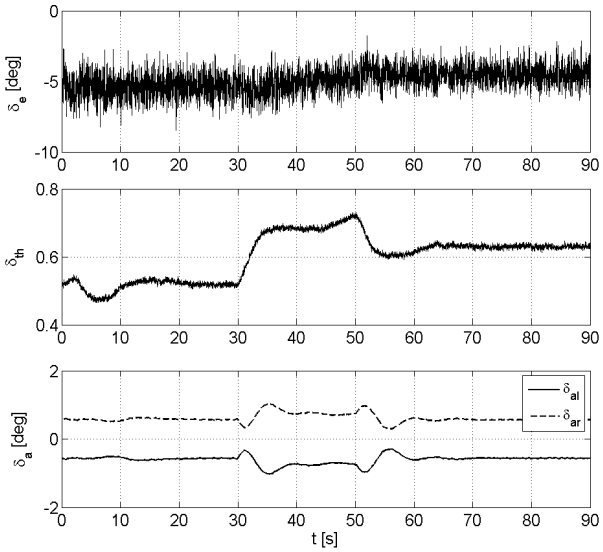

Fig. 9 SIL commands for Ultrastick 25e, maneuver

4.2 Hardware-in-the-loop simulations

This section presents the results obtained applying the backstepping controller running on a microcontroller board to the Simulink Ultrastick 25e aircraft model. These simulations imitate the closest the behavior of the real aircraft. The controller is implemented on the XMOS XK-1A board, see Fig. 10. The XMOS board is a technology produced by XMOS Ltd [29]. This board is characterized by a multi-core multi-thread processor able to perform eight real time tasks in parallel. Its parallel computing ability is essential for unmanned applications where high level tasks, for instance the control logic, have to be combined with low level assignments, such as I/O [30]. The board low cost, limited weight (19 g) and dimensions (50 x 50 mm) make it suitable for small UAV applications.

Fig. 10 XMOS XK-1A board

HIL is obtained following the scheme of Fig. 11. The states and the reference values generated by the Simulink model are sent through serial connection to the XMOS board, the resulting commands are sent back to Simulink. The Simulink model and the controller integrate with a 0.01 seconds time step. The data frequency to and from the XMOS board are sent with a 0.05 seconds interval in order to simulate the sensors and servos update rate. As a consequence, the controller is repeated five times for each set of input data.

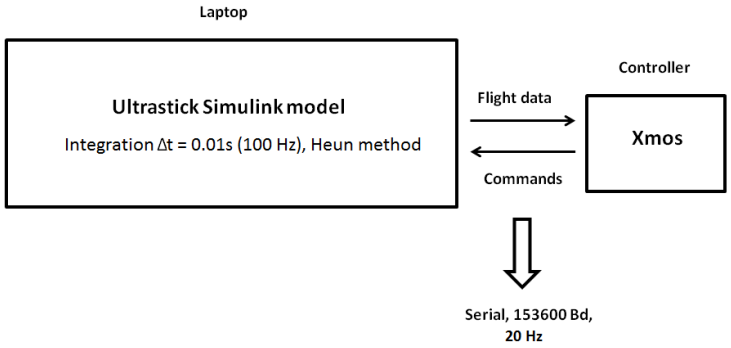

Fig. 11 HIL layout

The same reference inputs of the SIL cases are proposed. The results for the trim condition hold are represented in Fig. 12 while Fig. 13 and Fig. 14 show, respectively, the states and the commands for the coupled maneuver. Comparing with the SIL simulations images it is possible to affirm that results are basically equivalent. The slightly higher oscillatory behavior at initial time and during transition might be caused by the slower control rate of the HIL case. Note that also PID gains need to be adjusted in particular for the speed control.

5 Aircraft - Controller integration

The general autopilot configuration and the relationship of the microcontroller board with the chosen I/O devices is illustrated in Fig. 15. The board chosen for the in-flight control remains the XMOS XK-1A, the radio link frequency is 2.4 GHz. The sensors were illustrated in previous sections. The controller board outputs go to the aircraft motor and to the four analog mini servos controlling elevator, rudder and two ailerons. Power is supplied by a 3 cell LyPo battery with 4200 mAh. The servos and the propulsion system are the same used by the Minnesota research group in order to guarantee similar results.

The aircraft needs to be piloted during non-controlled maneuvers (take off and landing for instance) or as a safety feature to recover from unwanted behaviors. Therefore, the radio receiver needs to be integrated with the XMOS board in order to allow manual operations and to guarantee a switch between Computer In Command (CIC) and Pilot In Command (PIC) modes. The transition is guaranteed by

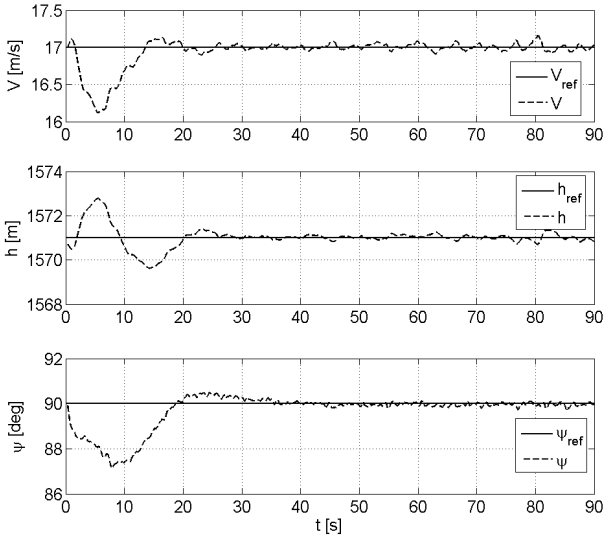

Fig. 12 HIL outer loop response for Ultrastick 25e, trim conditions hold

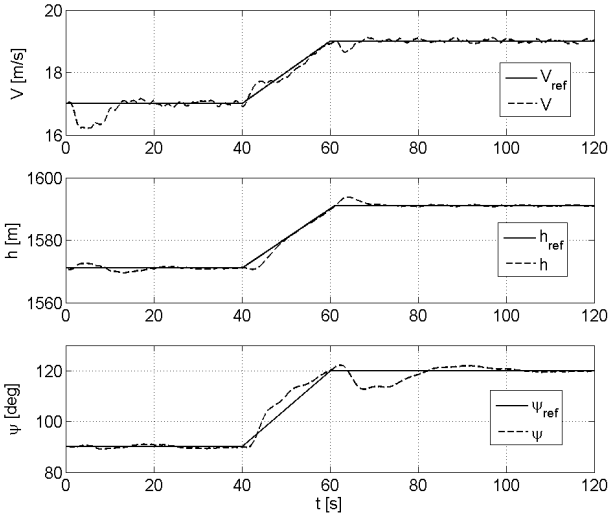

Fig. 13 HIL outer loop response for Ultrastick 25e, maneuver

a switch button on the radio transmitter acting on the GEAR channel. When the PIC mode is active the XMOS software monitors the THROTTLE, ELEVATOR, AILERON and RUDDER radio channels and forwards the received inputs to the motor and the servos. When CIC mode is active, the channels are ignored and the desired signals are directly generated by the controller. When in autonomous mode

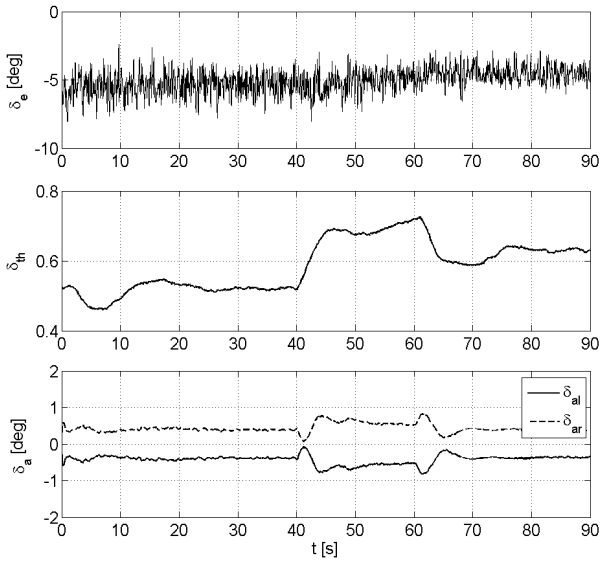

Fig. 14 HIL commands for Ultrastick 25e, maneuver

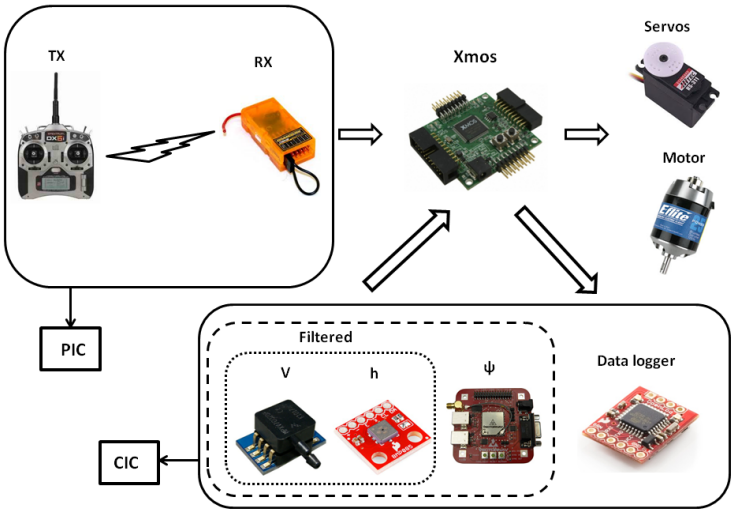

Fig. 15 Ultrastick 25e logical controller integration scheme

all sensors data and Pulse Width Modulation (PWM) commands are recorded for post-flight analysis on a micro-SD card.

The only source of power on the aircraft is the 11.1 V battery that supplies power to the electric motor and to all onboard systems. As the servos, the RX radio and the microcontroller require a 5 V supply, a voltage regulation is required. An

Table 5: Power budget for onboard electronics

Component	Maximum current
Servo	4×1 A
XMOS	0.5 A (just CPU)
Data logger	6 mA
Barometric sensor	0.012 mA
Airspeed sensor	10 mA
Heading sensor	65 mA
ADC	0.85 mA
Radio receiver	50 mA
Total	> 4.6 A

initial setup where the regulated output is provided to all utilities by a single motor controller via BEC was tested. Unfortunately, when all servos where overloaded this resulted in a drop of voltage and the loss of the radio connection. In fact, the considered ESC has maximum continuous voltage equal to 2.5 A. By observing the power budget illustrated in Table 5 it is clear that when servos are employed at their maximum this value is not sufficient. This problem was overcome with the adoption of two motor controllers, both connected to the main battery. One is responsible for powering the servos via BEC, the other simply acts as a DC regulator for the radio and in cascade the XMOS board and all sensors.

The pitot tube is installed on the right wing so that the pressure ports are ahead of the wing leading edge. The installation of the pitot sensor is represented in Fig. 16, the analog transducer is placed directly inside the wing as it is preferable to run into the wing an electric cable instead of a silicon tube. This might introduce some noise but considerably reduces pressure losses. The pitot tube is aligned with the aircraft fuselage. As stated by [31] generally pitot tubes can handle flows with an angle up to $\pm$ 12 degrees with respect to their axis, well beyond the values of α and β here expected.

Fig. 17 illustrates the disposition of all electronic devices on the Ultrastick 25e bay. The IMU is screwed on a wooden support tightly blocked on the fuselage. All other devices are just positioned in the free space, when flying they are wrapped in foam to reduce vibrations and mutual friction.

In order to match the theoretical control surface deflection defined by the backstepping controller with the real surface deflection, a calibration has to be performed. The aim is to identify the mathematical correlation between the PWM given to the servos and the corresponding surface deflection. In this way, when the backstepping controller calculates a desired deflection the conversion to PWM for the servos is straightforward.

Twenty PWM steps from minimum (1000 microseconds) to maximum (2000 microseconds) are sent to the servos and the corresponding rotations are recorded. Data are processed with Matlab to generate the PWM-angular displacement curve. Data distribution shows that a linear relationship is not assumable and an asymmetry between positive and negative deflections is sometimes observable. Two second order polynomials obtained with least squares method are used to interpolate the rotation of the surfaces, one for each direction of deflection. Aileron curves are displayed in Fig. 18, the upper part illustrates the results for negative deflections while the lower for positive deflections. Their mathematical formulation is represented in (16). Similar curves are obtained for elevator and rudder. For

Fig. 16 Pitot installation on the Ultrastick 25e right wing

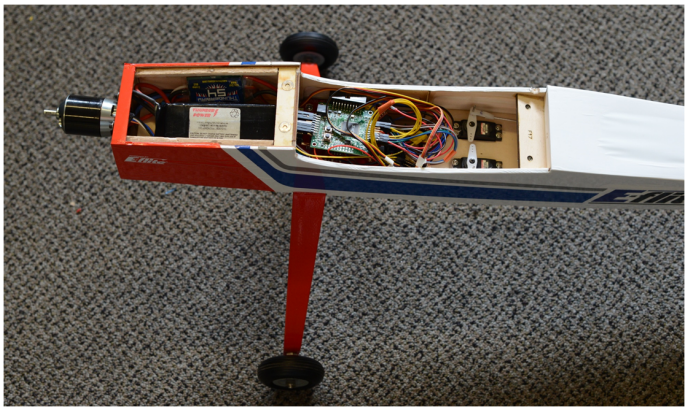

Fig. 17 Ultrastick 25e aircraft with sensors and controller

the motor a similar test is performed, for each PWM value a rotational regime is measured through an optical tachometer. Tests confirm that a linear relationship can be assumed with good confidence. No throttle corresponds to the minimum PWM and full throttle to maximum PWM.

$$\begin{cases} -991.8 \cdot {\delta_a}^2 + 888.3 \cdot \delta_a + 1504.6 = 0 & \delta_a > 0 \\ 165.5 \cdot {\delta_a}^2 + 980.5 \cdot \delta_a + 1506.6 = 0 & \delta_a < 0 \end{cases} \tag{16}$$

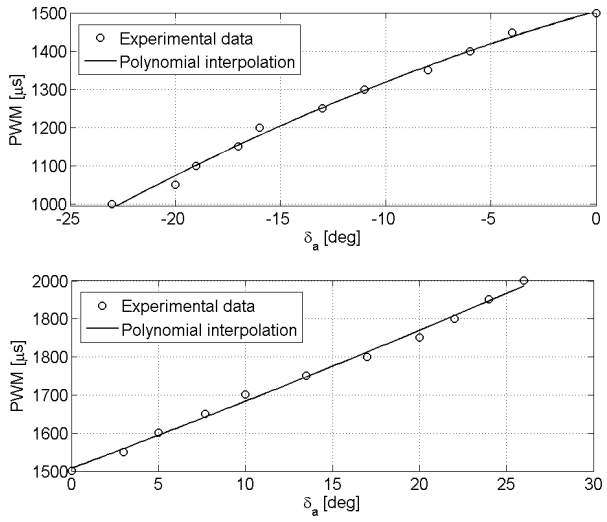

Fig. 18 Ultrastick 25e aileron servo calibration curves

6 Experimental Tests

This section illustrates the preliminary tests performed with the Ultrastick 25e aircraft controlled with backstepping. The results of a set of ground tests are initially illustrated. Unfortunately, some logistic problems limited the number of flight tests to a handful. The most significant results and considerations are here proposed

6.1 Preliminary Ground Test

The configuration for the ground test is slightly different from the one adopted for the flight. First, the propeller is not mounted for safety reasons. Second, as the aircraft is still and so the pitot would indicate zero velocity, the code is corrected to assign a ficticious input velocity equal to the cruise speed. The aircraft is rotated and lifted to simulate heading and minimal altitude changes. Aileron and rudder commands are visible, minor throttle changes are audible. All data are recorded on micro-SD card and analyzed. In these preliminary tests the controller is asked to maintain the conditions measured when CIC mode is engaged.

Results of the ground test are shown in the following figures: Fig. 19 for the elevator response to speed variations, Fig. 20 for the altitude to throttle loop and Fig. 21 for heading control with latero-directional commands. The maneuver consists in turning the aircraft of 360 degrees while lifting it from the floor to maximum arm extension, the velocity is constant. All commands act to counteract perturbations. Note that the elevator peak at 15 seconds is caused by the coupling effect generated by the heading angle measuring 180 degrees.

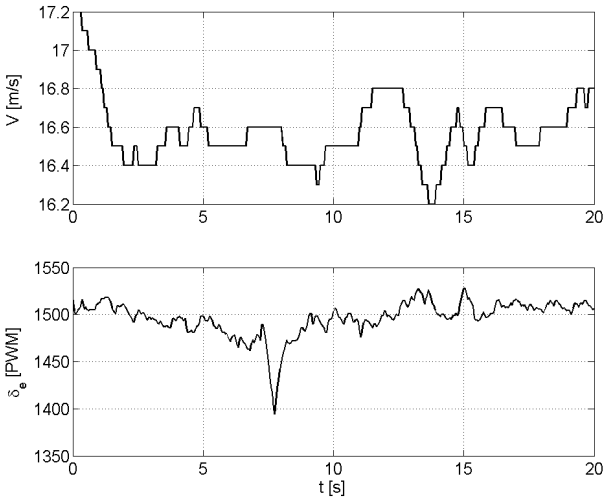

Fig. 19 Ultrastick 25e ground test on V control

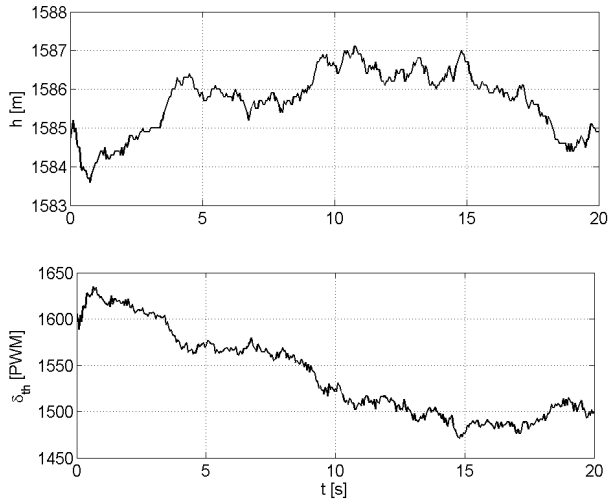

Fig. 20 Ultrastick 25e ground test on h control

6.2 Preliminary Flight Tests

Globally five flights were performed, of these four were dedicated to latero-directional control and one to longitudinal control. The code was modified so that when backstepping is controlling the latero-directional plane with aileron and rudder, the pilot still has throttle and elevator control. The contrary happens when backstep-

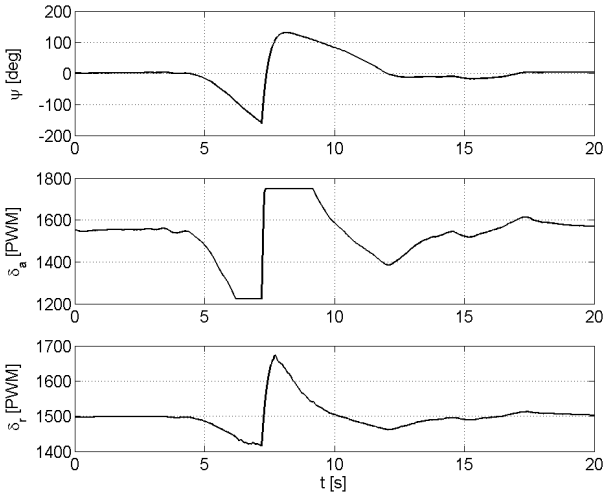

Fig. 21 Ultrastick 25e ground test on ψ control

ping controls the longitudinal plane. This approach allows testing the controller behavior without controller-induced cross-coupling effects and so facilitates the initial PID tuning procedure. During the autonomous flight the pilot is asked to avoid touching the stick unless minimal corrections are required or dangerous situations occur. The microcontroller code is designed so that the aircraft maintains the initial conditions that are measured when switching from PIC to CIC.

The most promising results were obtained in the flight controlling altitude and velocity. Figures 22, 23 and 24 show the sensors data and the corresponding commands recorded. From Fig. 22 and Fig. 23 it is possible to observe that the aircraft is able to control both velocity and altitude. Initially the aircraft loses some altitude and thus accelerates, the controller reacts increasing the throttle and reducing (in sign) the elevator deflection. The aircraft starts to climb again and slows down closer to the target velocity. It is possible to observe from Fig. 23 how the throttle quickly saturates without much affecting the altitude response, a more powerful motor would have resulted in better performance. Fig. 24 shows that the initial heading is not maintained, despite some pilot corrections on the aileron. These generate a coupling effect on the velocity visible from second 6, the elevator command tries to cancel it. An adjustment in the PID tuning, in particular for the velocity control loop, is expected to guarantee considerable improvements.

Tests on the latero-directional plane did not guarantee satisfactory results. In all four tests where slightly different gains configurations were tried an excessive roll motion was recorded. Further tests need to be carried out to investigate the cause of the problem and to address it effectively. A likely cause is an imprecise tuning of the gains or a inaccuracy in the PWM-δ_a correlation. In fact, during the simulations the aircraft has showed to be extremely sensible to minimal variations in the aileron command.

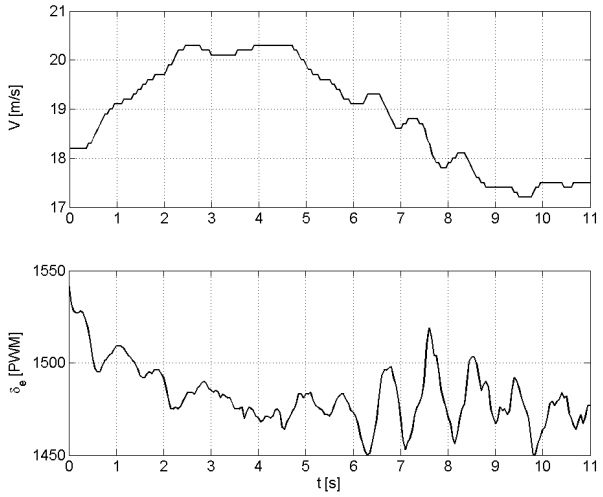

Fig. 22 Ultrastick 25e longitudinal flight test, V control

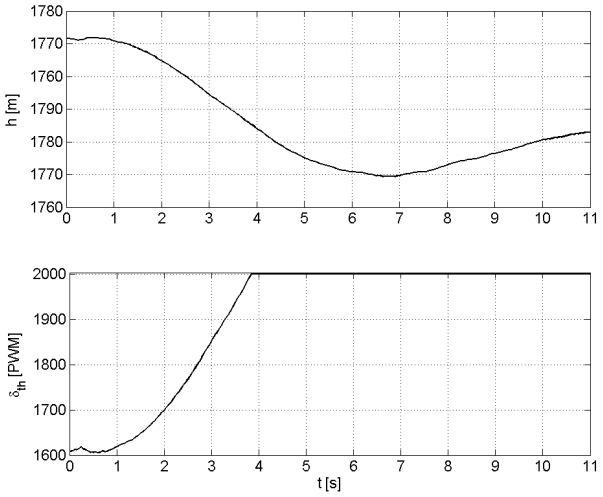

Fig. 23 Ultrastick 25e longitudinal flight test, h control

7 Conclusions and Future Work

The present paper illustrates the procedure adopted for implementing and testing a nonlinear backstepping controller on a fixed-wing UAV. The main theoretical basis of the control structure are summarized according to the full description already proposed by the same authors in [1]. In this paper SIL and HIL simulations

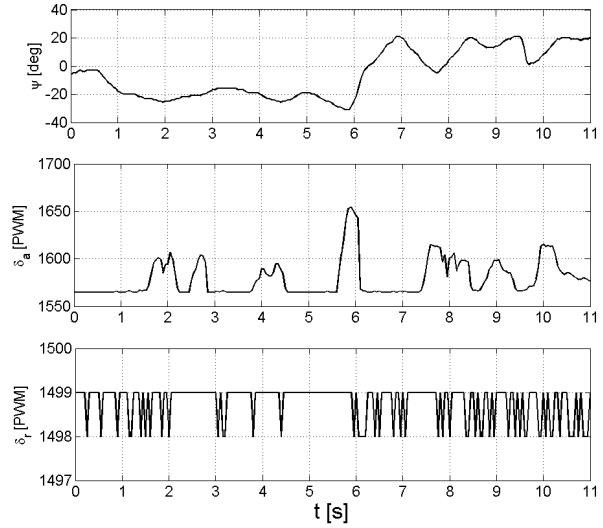

Fig. 24 Ultrastick 25e longitudinal flight test, latero-directional variables

demonstrate that this backstepping approach is able to control with satisfactory performance the aircraft adopted for the flight tests. A high-fidelity model of the aircraft inclusive of real sensors noise is employed. The paper also describes the installation and the integration of the backstepping-based autopilot on the aircraft. The problems encountered and the solutions adopted are explained. Finally, the results of some preliminary ground and flight tests are illustrated. Ground tests demonstrate that this autopilot configuration is suitable for the control of the aircraft. The limited number of performed flight tests indicate that a good longitudinal control can be easily achieved, while improvements on the latero-directional plane control are still necessary.

In order to identify these improvements a larger number of flight tests is required. A systematic analysis of the loop on ψ need to be performed to correlate the gains adjustments with the aircraft roll behavior. In early stage a comparison with simulated results might be helpful. Finally, a study to improve the precision of the correlation between theoretical and physical commands is suggested. In fact, the high responsiveness of the aircraft is affected by minimal disparities.

References

1. Sartori, D., Quagliotti, F., Rutherford, M. J., Valavanis, K. P.: Design and Development of a Backstepping Controller Autopilot for Fixed-wing UAVs. Denver University Unmanned Systems Research Institute, CO, USA, Tech. Rep. DU2SRI-2013-12-001 (2013)

2. Härkegård, O.: Backstepping Designs for Aircraft Control - What is there to gain?. Division of Automatic Control, Department of Electrical Engineering Linköpings Universitet, Sweden, Tech. Rep. LiTH-ISY-R-2339 (2001)

3. Krstić, M., Kanellakopoulos I., Kokotović, P.: Nonlinear and Adaptive Control Design. Wiley, New York (1995)

4. Kim, K-S., Kim, Y.: Robust Backstepping Control for Slew Maneuver Using Nonlinear Tracking Function. IEEE Trans. on Control Syst. Technol. 11, 822-829 (2003)

5. Tu, H., Du, X.: The Design of Small UAV Autopilot Hardware System Based on DSP. Proc. of the 2010 Int. Conf. on Intell. Comput. Technol. and Autom. 3, 780-783, Changsha, China (2010)

6. Guanglin, H., Rujun, G., Shi, Y.: Application of FPGA in Small UAV Autopilot Based on Embedded Linux System. Proc. of the 33rd Annu. Conf. of the IEEE Ind. Electron. Soc. 731-734, Taipei, Taiwan (2007)

7. Chao, H. Y., Cao, Y. C., Chen, Y. Q.: Autopilots for Small Unmanned Aerial Vehicles: A Survey. Int. J. of Control Autom. and Syst. 8, 36-44 (2010)

8. Ollero, A., Merino, L.: Control and Perception Techniques for Aerial Robotics. Annu. Rev. in Control 28, 167-178, (2004)

9. Gregory, I. M., Xargay, E., Cao, C., Hovakimyan, N.: Flight Test of $\mathcal{L}_1$ Adaptive Control on the NASA AirSTAR Flight Test Vehicle. AIAA Guid., Navig. and Control Conf., Toronto, Canada (2010)

10. Keviczky, T., Balas, G. J.: Flight Test of a Receding Horizon Controller for Autonomous UAV Guidance. 2005 Am. Control Conf. 3518-3523, Portland, OR, USA (2005)

11. Ju, H-S., Tsai, C-C.: Longitudinal Axis Flight Control Law Design by Adaptive Backstepping. IEEE Trans. on Aerosp. and Electron. Syst. 43, 311-329 (2007)

12. Jung, D., Tsiotras, P.: Bank-to-Turn Control for a Small UAV Using Backstepping and Parameter Adaptation. Proc. of the 17th IFAC World Congr. 17, 4406-4411, Seoul, South Korea (2008)

13. Lee, T., Kim, Y.: Nonlinear Adaptive Flight Control Using Backstepping and Neural Networks Controller. J. of Guid., Control, and Dyn. 24, 675-682 (2001)

14. Sonneveldt, L., Chu, Q. P., Mulder, J. A.: Nonlinear Flight Control Design Using Constrained Adaptive Backstepping. J. of Guid., Control, and Dyn. 30, 322-336 (2007)

15. Matthews, J. S., Knoebel, N. B., Osborne, S. R., Beard, R. W., Eldredge, A.: Adaptive Backstepping Control for Miniature Air Vehicles. Proc. of the Am. Control Conf., Minneapolis, MN, USA (2006)

16. Brezoescu, A., Espinoza, T., Castillo, P., Lozano, R.: Adaptive Trajectory Following for a Fixed-Wing UAV in Presence of Crosswind. J. of Int. & Robot. Syst. 69, 257-271 (2013)

17. Härkegård, O., Torkel Glad, S.: Flight Control Design Using Backstepping. Proc. of the 5^th IFAC Symp., St. Petersburg, Russia, (2001)

18. O. Härkegård: Backstepping and Control Allocation with Applications to Flight Control. Ph.D. diss., Dep. of Electr. Eng., Linköping Univ., Linköping, Sweden (2001)

19. Etkin, B., Reid, L. D.: Dynamics of Flight: Stability and Control. Wiley, New York (1996)

20. Guglieri, G.: Effect of Autopilot Modes on Flight Performances of Electric Mini-UAVs. The Aeronaut. J. 117, 57-69 (2013)

21. Dorobantu, A., Murch, A. M., Mettler, B., Balas, G. J.: Frequency Domain System Identifcation for a Small, Low-Cost, Fixed-Wing UAV. AIAA Guid., Navig., and Control Conf., Portland, OR, USA (2011)

22. Paw, Y. C.: Synthesis and Validation of Flight Control for UAV. PhD Thesis, University of Minnesota, MN, USA (2009)

23. http://www.uav.aem.umn.edu/

24. http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=MPXV7002

25. http://www.linear.com/product/LTC1865

26. http://www.bosch-sensortec.com

27. https://www.sparkfun.com/products/11282

28. http://www.vectornav.com/

29. http://www.xmos.com/

30. Martins, G., Moses, A., Rutherford, M. J., Valavanis, K. P: Enabling intelligent unmanned vehicles through XMOS Technology. J. of Def. Model. and Simul. Appl., Methodol., Technol. 9, 71-82 (2012).

31. Selig, M. S., Deters, R. W., Williamson, G. A.: Wind Tunnel Testing Airfoils at Low Reynolds Numbers. 49^{th} AIAA Aerosp. Sci. Meet., Orlando, FL, USA (2011)