Stabilization and Control of a UAV Flight Attitude Angles using the Backstepping Method

Mihai Lungu

Abstract—The paper presents the design of a mini-UAV attitude controller using the backstepping method. Starting from the nonlinear dynamic equations of the mini-UAV, by using the backstepping method, the author of this paper obtained the expressions of the elevator, rudder and aileron deflections, which stabilize the UAV, at each moment, to the desired values of the attitude angles. The attitude controller controls the attitude angles, the angular rates, the angular accelerations and other variables that describe the UAV longitudinal and lateral motions. To design the nonlinear controller, by using the backstepping technique, the nonlinear equations and the Lyapunov analysis have been directly used. The designed controller has been implemented in Matlab/Simulink environment and its effectiveness has been tested with a campaign of numerical simulations using data from the UAV flight tests. The obtained results are very good and they are better than the ones found in previous works.

Keywords—Attitude angles, Backstepping, Controller, UAV.

I. INTRODUCTION

All over the world, Unmanned Aerial Vehicles have a broad range of civil and military applications. A lot of scientists design UAVs that have to meet some expectations regarding their reliability, low cost, small size and so on and so forth. In the domain of small-size UAVs, the most systems are still deployed as prototypes because of their lack of reliability. Their reliability may be increased by improving their modeling and their flight control systems [1,2]. Modeling, simulation, analysis and flight testing of full-size aircrafts or large UAVs are very well presented in lots of scientific works over the past few decades [3].

In this paper, the equations of the flying object motion are nonlinear [2], the backstepping method being used here for the control of the UAV attitude angles. There are a lot of techniques for the control of flying objects when their flight is described by nonlinear equations: dynamic inversion, nonlinear predictive control or techniques that use neuronal networks [4]. The system nonlinearities are cancelled and a closed loop linear system is obtained by using the first method. This method disadvantage is the existence of nonlinearities and their derivatives. This is a serious problem because the aerodynamic forces and moments, which describe the system, can not be modeled at a high precision degree [5]. An alternative to the dynamic inversion is the backstepping method; it is a simpler method and its most important advantage is that the nonlinearities must not be necessary cancelled in the control law.

In this paper the attitude controller is projected for the Sekwa UAV (Fig. 1) [6] by means of the backstepping method. This UAV is characterized by a mobile centre mass in order to minimize the drag force. The design of the attitude angles controller (autopilot) means the determination of the three deflections (elevator, ailerons and rudder deflections) so that the angles characterizing the UAV attitude tend to their imposed values. The Defense, Peace, Safety and Security (DPSS) branch of the South African Council for Scientific and Industrial Research (CSIR) launched, some years ago, the Sekwa program; the main objective of this research program was to demonstrate the advantages of UAVs with variable centre mass [6]. The main characteristics of this UAV are the lack of the vertical empennage and the blended wing; these thinks lead to small drag forces and superior aerodynamic performances. The only disadvantage may be the stability of the UAV; as a consequence, the designed autopilot must be very efficient. The purpose of the flight control system was to increase the natural UAV stability so that it is characterized by nominal static stability and all its motion variables are stabilized toward their desired values.

The main sensors that are used on UAVs are: three accelerometers (for the measurement of the linear accelerations \( \dot{u}, \dot{v}, \dot{w} \), which, by integrating, lead to velocities \( u, v, w \)) three gyroimeters (for the measurement of the angular velocities \( p, q, r \)), sensors for static and dynamic pressure (the first for the determination of the barometric altitude and the both sensors for the determination of the flight velocity), a radio-altimeter or other system for the measurement of the UAV’s height with respect to the ground and so on [7,8,9].

In contrast with [6], in this paper, the UAV flight control is made by using the backstepping method and the efficiency of this method will be demonstrated by a campaign of numerical simulations using data from the UAV flight tests. Most UAV autopilots use classical proportional-integral-derivative (PID) controllers and ad-hoc methods to tune the controller gains during the flight. This methodology is not the best one because it has high risks and because there are a lot of limitations in the UAVs performances and robustness. It is important to obtain an integrated framework that enables it to rapidly synthesize, implement, analyze and validate a controller configuration by using iterative development cycles [10-14].

Fig. 1 The model of the Sekwa UAV

M. Lungu is with the University of Craiova, Faculty of Electrical Engineering, Avionics Division, 107 Decembal Street, 200440 Craiova, Dolj, Romania (email: Lma1312@yahoo.com, mlungu@elth.ucv.ro).

International Scholarly and Scientific Research & Innovation 6(1) 2012 53 scholar.waset.org/1999.8/6207
The mathematical procedure of using the backstepping method to the stabilization and control of a mini UAV, the determination of the control laws (deflections of the elevator, rudder and ailerons), which assure the control of mini UAV, and the new controller software implementation represent the originality elements of this paper.

The paper is organized as follows: the dynamic equations of the Sekwa UAV is given in section II, the controller design for the pitch angle channel is presented in section III; in section IV and V the author determines the control laws for the stabilization and control of the roll and yaw angles. In section VI, the designed controller is implemented in Matlab/Simulink environment and its effectiveness is tested with a campaign of numerical simulations using data from the UAV flight tests; finally, some conclusions are shared in section VII.

II. DYNAMIC EQUATIONS OF THE SEKWA MINI-UAV

There are a lot of methods to define the model of the UAVs [15-17]. As one can see in [18], the control system architecture can be greatly simplified by judiciously expressing the aircraft dynamics and carefully selecting the variables that need to be controlled. Thus, the complexity of the automatic pilot is greatly reduced and most design techniques become very efficient [6].

In order to obtain the mathematical model of the Sekwa UAV, one needs to define three axis systems. The first one is the inertial reference system and the Newton’s laws can be successfully applied with respect to this frame. To use this frame, it is assumed that the surface of the Earth is flat and non-rotating. This frame [19] defines a plane that is tangential to the Earth surface (Fig. 2); the system centre is conveniently chosen as a point from the runway. The second frame (OXYZ) is the body axis system with its origin in the centre mass of the flying object. The OX axis is the longitudinal axis of the UAV, the OY is the lateral axis, pointing out the aircraft right starboard wing, while OZ axis is the vertical axis and it is defined to reside within the UAV’s plane of symmetry. The third frame is the well known aerodynamic frame, its origin being the same with the origin of the inertial frame; the OX axis has, this time, the direction of the UAV total velocity. This frame is very important because the determination of the stability derivates is made with respect to this axis system.

\[
\begin{align*}
\dot{\phi} &= p + \tan \theta (q \sin \varphi + r \cos \varphi), \\
\dot{\theta} &= q \cos \varphi - r \sin \varphi, \\
\dot{\psi} &= \frac{q \sin \varphi + r \cos \varphi}{\cos \theta}.
\end{align*}
\]

To obtain the UAV dynamics, the UAV is considered to be a rigid body with six degree of freedom [6,18]. To simplify the derivation of the UAV model, a static centre of mass position is assumed. The dynamics of the UAV is considered to be the resultant between the kinematics and the kinetics of the UAV. Because the purpose of this paper is not to deduce the dynamic equations of the Sekwa UAV, I will concentrate on the UAV control design using the backstepping method; the Sekwa UAV dynamics is presented in detail in [6] and [18]. Thus, the kinematics of the UAV is [6,18]:

\[
\begin{align*}
\delta_s &= \left(\delta_2 + \delta_1 + \delta_4 + \delta_5\right)/4, \\
\delta_a &= \left(-\delta_2 - \delta_1 + \delta_4 + \delta_5\right)/4, \\
\delta_r &= \left(\delta_1 + \delta_5\right)/2.
\end{align*}
\]
\[
L = p_s S(C_e C_p \cdot C_i - C_e S_\alpha \bar{C} \cdot C_m - S_\alpha b \cdot C_n), \\
M = p_s S(S_\alpha C_p \cdot C_i + C_e \bar{C} \cdot C_m), \\
N = p_s S(S_a C_p \cdot C_i - S_\alpha S_\alpha \bar{C} \cdot C_m + C_a b \cdot C_n),
\]

where \( p_s \) is the dynamic pressure, \( S \) – the wing surface, \( \alpha, \beta \) – attack and sideslip angles, \( b \) – the wing span, \( \bar{C} \) – mean aerodynamic chord and \( C_a = \cos \alpha, C_b = \cos \beta \), \( S_\alpha = \sin \alpha, S_\beta = \sin \beta \). In (3), the aerodynamic moment coefficients \( C_i, C_m, C_n \) (roll, pitch and yaw, respectively) have the expressions [6]:

\[
C_i = C_i \beta + C_{i,\beta} \delta_\beta + C_{i,\alpha} \delta_\alpha + \frac{b}{2V} (C_{i,p} + C_{i,r}), \\
C_m = C_m \alpha + C_{m,\alpha} \alpha + C_{m,\beta} \beta + \frac{\bar{C}}{2V} C_{m,q}, \\
C_n = C_n \beta + C_{n,\alpha} \alpha + C_{n,\beta} \beta + \frac{b}{2V} (C_{n,p} + C_{n,q}).
\]

In (4) \( C_m \) is the static moment coefficient and it has values close to zero; that is why, in simulations, \( C_m \) will be considered null. The equations describing the projections of the UAV angular velocities along OXYZ frame axes are [6]:

\[
\dot{\rho} = \frac{1}{I_z} [qr(I_1 - I_z) + p_s SC_i], \\
\dot{q} = \frac{1}{I_z} [pr(I_1 - I_z) + p_s SC_i], \\
\dot{r} = \frac{1}{I_z} [pq(I_1 - I_z) + p_s SC_i],
\]

with

\[
C_i = C_a C_p b \cdot C_i - C_a S_\alpha \bar{C} \cdot C_m - S_\alpha b \cdot C_n, \\
C_m = S_\alpha C_p b \cdot C_i + C_e \bar{C} \cdot C_m, \\
C_n = S_a C_p b \cdot C_i - S_\alpha S_\alpha \bar{C} \cdot C_m + C_a b \cdot C_n;
\]

\( I_1, I_2, I_3 \) are the axial inertia moments.

For the design of the Sekwa UAV controller, for the stabilization and control of the attitude angle \((\theta, \phi, \psi)\), the kinematics equations of the UAV, the equations of the aerodynamic moment coefficients \( C_i, C_m, C_n \) and the equations describing the projections of the UAV angular velocities along the OXYZ frame (equations (2), (4) and (5)) are needed.

One of the best controller design procedures is the backstepping method. The backstepping technique is based on Lyapunov theory and it offers multiple possibilities for the use of nonlinearities with respect to the dynamic inversion technique. Thus, some nonlinearities may be maintained, while others, which are not necessary, are to be cancelled [5,20]; the control law is simpler. In order to use the backstepping method, the system must be written under the following form:

\[
x_1 = f_1(x_1, x_2), x_2 = f_2(x_1, x_2, x_3), x_3 = f_3(x_1, x_2, x_3, \mu),
\]

where \( f_1, f_2, f_3 \) are nonlinear functions. \( x_2 \) plays the role of a virtual control for the command of the variable \( x_1 \) toward zero, without taking into account the dynamics of \( x_3 \). Then, \( x_3 \) is used as a virtual control for the convergence to zero of the variable \( x_2 \). Finally, the input variable \( \mu \) is the one that stabilizes the variable \( x_1 \).

### III. CONTROLLER DESIGN FOR THE PITCH ANGLE CHANNEL

To stabilize the UAV pitch angle, the paper author considers \( \theta \) – the imposed (desired) value of the pitch angle \( \theta \) and defines the error \( e_\theta = \theta - \theta \) [5]. It is evident that \( \dot{e}_\theta = 0 \) and, taking into account (2), it yields:

\[
\dot{e}_\theta = -\mu e_\theta + \left( q + \frac{\mu e_\theta}{\cos \phi} - r \sin \phi \right) \cos \phi,
\]

with \( \mu \) – positive constant. The design aim is \( e_\theta \to 0 \); as a result, it is desirable that:

\[
q + \frac{\mu e_\theta}{\cos \phi} = 0 \Leftrightarrow q = -\frac{\mu e_\theta}{\cos \phi}.
\]

The equation of the error \( e_\phi \), using (9), becomes:

\[
\dot{e}_\phi = -\mu e_\theta \sin \phi;
\]

in order to demonstrate that (10) corresponds to a stable system, the Lyapunov function \( V_\theta(e_\theta) = \frac{1}{2} e_\theta^2 \) has been chosen [5] and, by computing \( \dot{V}_\theta(e_\theta) \), it follows:

\[
\dot{V}_\theta(e_\theta) = e_\theta \cdot \dot{e}_\theta = -\mu e_\theta^2 < 0, \forall \mu > 0.
\]

The Lyapunov function is negative and, as a consequence, the system (10) is a stable one and it converges to zero \( e_\theta \to 0 \). The second error variable is now chosen as \( e_q = q - \bar{q} \), with \( \bar{q} \) having the form (9), i.e.:

\[
\bar{q} = -\frac{\mu e_\theta}{\cos \phi} + r \sin \phi.
\]
As a result, the following equation system is obtained:

\[
\begin{align*}
\dot{e}_q &= -\mu_\theta e_\theta + e_\varphi \cos \varphi, \\
\dot{e}_\varphi &= q - \ddot{q}.
\end{align*}
\] (13)

By the substitution of \( \dot{q} \), with the form (5), in the second equation (13), it follows:

\[
\dot{e}_q = \frac{1}{I_y} \left[ pr(I_z - I_x) + \frac{p_s S}{I_y} S_y \left( C_{\varphi \theta} + C_{\varphi \theta} \dot{\delta}_e \right) + \frac{p_s S}{I_y} S_y \left( C_{\varphi \theta} \delta_e + \frac{b}{2V} \left( C_{\varphi \theta} p + C_{\varphi \theta} \right) \right) + \frac{p_s S}{I_y} S_y \left( C_{\varphi \theta} \delta_e + \frac{b}{2V} \left( C_{\varphi \theta} p + C_{\varphi \theta} \right) \right) \right] - \ddot{\bar{q}}: (14)
\]

By the substitution of \( \dot{q} \), with the form (5), in the second equation (13), it follows:

\[
\dot{e}_q = \frac{1}{I_y} \left[ pr(I_z - I_x) + \frac{p_s S}{I_y} S_y \left( C_{\varphi \theta} + C_{\varphi \theta} \dot{\delta}_e \right) + \frac{p_s S}{I_y} S_y \left( C_{\varphi \theta} \delta_e + \frac{b}{2V} \left( C_{\varphi \theta} p + C_{\varphi \theta} \right) \right) + \frac{p_s S}{I_y} S_y \left( C_{\varphi \theta} \delta_e + \frac{b}{2V} \left( C_{\varphi \theta} p + C_{\varphi \theta} \right) \right) \right] - \ddot{\bar{q}}
\]

By the substitution of \( \dot{q} \), with the form (5), in the second equation (13), it follows:

\[
\dot{e}_q = \frac{1}{I_y} \left[ pr(I_z - I_x) + \frac{p_s S}{I_y} S_y \left( C_{\varphi \theta} + C_{\varphi \theta} \dot{\delta}_e \right) + \frac{p_s S}{I_y} S_y \left( C_{\varphi \theta} \delta_e + \frac{b}{2V} \left( C_{\varphi \theta} p + C_{\varphi \theta} \right) \right) + \frac{p_s S}{I_y} S_y \left( C_{\varphi \theta} \delta_e + \frac{b}{2V} \left( C_{\varphi \theta} p + C_{\varphi \theta} \right) \right) \right] - \ddot{\bar{q}}
\]

IV. CONTROLLER DESIGN FOR THE ROLL ANGLE CHANNEL

The procedure is similar to the one presented in section III; the author of the paper considers \( \bar{\varphi} \) – the desired (imposed) value of the roll angle and he defines the error [5]

\[
e_\varphi = \varphi - \bar{\varphi}.
\]

For \( \bar{\varphi} \) – constant, it follows \( \dot{e}_\varphi = \ddot{\varphi} \); taking into account the expression of \( \varphi \), it results:

\[
\dot{e}_\varphi = -\mu_\varphi e_\varphi + \left[ p + \left( \mu_\varphi e_\varphi + q \tan \theta \cdot \sin \varphi + r \tan \theta \cdot \cos \varphi \right) \right]. (22)
\]

For the convergence of the error \( e_\varphi \) to zero, (22) must have the form \( \dot{e}_\varphi = -\mu_\varphi e_\varphi \), with \( \mu_\varphi \) – positive constant. As a consequence, the roll angular velocity has the expression [5]:

\[
p = -\mu_\varphi e_\varphi - q \tan \theta \cdot \sin \varphi - r \tan \theta \cdot \cos \varphi.
\] (23)

Choosing the Lyapunov function \( V_\varphi (e_\varphi) = \frac{1}{2} e_\varphi^2 \), one gets:

\[
V_\varphi (e_\varphi) = e_\varphi \cdot \dot{e}_\varphi = -\mu_\varphi e_\varphi^2 < 0 \quad (\forall) \mu_\varphi > 0.
\] (24)

Thus, the system is a stable one and convergent to zero \( (e_\varphi \to 0 \Leftrightarrow \varphi \to \bar{\varphi}) \). The second error variable is now chosen: \( e_p = p - \bar{\theta} \); \( \bar{\theta} \) has the form:

\[
\bar{\theta} = -\mu_\varphi e_\varphi - q \tan \theta \cdot \sin \varphi - r \tan \theta \cdot \cos \varphi.
\] (25)

Under these conditions, I obtain the following system:

\[
\begin{align*}
\left\{ \begin{array}{l}
\dot{e}_\varphi = -\mu_\varphi e_\varphi + e_p \cdot \\
\dot{e}_p = p - \bar{\theta}
\end{array} \right.
\end{align*}
\] (26)

and replacing \( \dot{p} \), with the form in (5), into the second equation (26), it follows:
Taking into account the expressions of \( C_1, C_m \) and \( C_s \), it results:

\[
\hat{e}_r = \frac{1}{I_z} q r(I_y - I_z) - \bar{p} + \frac{p_S}{I_z} C_a C_p \beta + \frac{p_S}{I_z} C_o C_p \beta + \frac{p_S}{I_z} S_a \bar{p} \tilde{C} \alpha + \frac{b}{2V} (C_x p + C_r) r - \bar{p} - \frac{p_S}{I_z} S_a \bar{p} \tilde{C} \alpha + \frac{b}{2V} (C_y p + C_r) r.
\]  

(28)

The error \( e_r \) must be zero. For this, the Lyapunov function

\[
V_r(e_r, \dot{e}_r, \dot{e}_p) = \frac{1}{2} e_r^2 + \frac{1}{2} \dot{e}_r^2
\]

(29)

is chosen. By computing \( \dot{V}_r(e_r, \dot{e}_r) \), it results:

\[
\dot{V}_r(e_r, \dot{e}_r) = -\mu_p e_r^2 - \mu e_r^2 + e_r + \mu_p \dot{e}_r + \dot{e}_r.
\]

(30)

where \( \mu_p \) is a positive constant. The first two terms in the right side of (30) are negative and, because it is desirable that \( \dot{V}_r(e_r, \dot{e}_r) < 0 \), it yields:

\[
e_r + \mu_p \dot{e}_r + \dot{e}_p = 0.
\]

(31)

Replacing \( \dot{e}_r \), having the form (28), in the (31), I obtain:

\[
e_r + \mu_r \dot{e}_r + \dot{e}_p = 0 + \frac{1}{I_z} qr(I_y - I_z) + \frac{p_S}{I_z} C_a C_p \beta + \frac{p_S}{I_z} C_o C_p \beta + \frac{b}{2V} (C_x p + C_r) r - \bar{p} - \frac{p_S}{I_z} S_a \beta C_o \alpha + \frac{b}{2V} (C_y p + C_r) r - \bar{p} - \frac{p_S}{I_z} S_a \beta C_o \alpha + \frac{b}{2V} (C_y p + C_r) r = 0 ;
\]

(32)

thus, the second relationship between the deflections of the elevator, ailerons and rudder is obtained as follows:

\[
b_1 \delta_e + b_2 \delta_a + b_3 \delta_r = B, \tag{33}
\]

where

\[
b_1 = -\frac{1}{I_z} p_S C_a C_p \beta C_m \alpha - b_2 = \frac{1}{I_z} p_S S_h (C_a C_p \beta - C_m \alpha),
\]

\[
b_2 = \frac{1}{I_z} p_S S_h (C_a C_p \beta - C_m \alpha),
\]

\[
b_3 = -e_r - e_r - \frac{1}{I_z} q r(I_y - I_z) - \frac{p_S}{I_z} C_a C_p \beta - \frac{p_S}{I_z} C_o C_p \beta + \frac{p_S}{I_z} S_a \beta C_o \alpha + \frac{b}{2V} (C_x p + C_r) r + \frac{p_S}{I_z} S_a \beta C_o \alpha + \frac{b}{2V} (C_y p + C_r) r + \bar{p}.
\]

(34)

V. CONTROLLER DESIGN FOR THE YAW ANGLE CHANNEL

For the yaw angle stabilization, the approach is the same with the one used in sections III and IV. The error variables

\[
e_y = \psi - \psi, \quad e_r = r - \tilde{r}
\]

are used here; \( \psi \) is the imposed (desired) value of the yaw angle \( \psi \), while \( \tilde{r} \) has the form [5]

\[
\tilde{r} = -\mu_v \cos \theta \psi - \frac{\sin \theta}{\cos \theta} q.
\]

(36)

The Lyapunov functions, which have been used, are the following ones:

\[
V_y(e_y) = \frac{1}{2} e_y^2, \quad V_r(e_r, e_r) = \frac{1}{2} e_r^2 + \frac{1}{2} e_r^2.
\]

(37)

I obtain, in this case too, a relationship (the third one) between the deflections of the elevator, ailerons and rudder:

\[
c_1 \delta_e + c_2 \delta_a + c_3 \delta_r = C \tag{38}
\]

with

\[
c_1 = -\frac{1}{I_z} p_S S_h \beta C_m \alpha, \quad c_2 = \frac{1}{I_z} p_S S_h (C_a C_p \beta - C_m \alpha),
\]

\[
c_3 = \frac{1}{I_z} p_S S_h (C_a C_p \beta - C_m \alpha),
\]

\[
C = -e_y - e_r - \frac{1}{I_z} q r(I_y - I_z) - \frac{p_S}{I_z} C_a C_p \beta - \frac{p_S}{I_z} C_x p + C_r r + \frac{b}{2V} (C_x p + C_r) r + \frac{p_S}{I_z} S_a \beta C_o \alpha + \frac{b}{2V} (C_y p + C_r) r + \bar{p}.
\]

(39)
The determination of the elevator, ailerons and rudder deflections \( \{ \delta_e, \delta_a, \delta_r \} \) is made by solving the system formed by equations (20), (33) and (38). Thus, it results the system:

\[
\begin{align*}
\delta_e + a_1 \delta_a + b_1 \delta_e &= A, \\
b_2 \delta_a + c_2 \delta_a + b_3 \delta_r &= B, \\
c_1 \delta_a + c_1 \delta_e + c_3 \delta_r &= C,
\end{align*}
\] (40)

with \( a_i, b_i, c_i, A, B, C, i = 1,3 \) being described by (21), (34) and (39). The equation system (40) has the matricial form:

\[
\begin{bmatrix}
a_1 & a_2 & a_3 \\
b_1 & b_2 & b_3 \\
c_1 & c_2 & c_3
\end{bmatrix}
\begin{bmatrix}
\delta_e \\
\delta_a \\
\delta_r
\end{bmatrix}
= 
\begin{bmatrix}
A \\
B \\
C
\end{bmatrix},
\] (41)

with the solution

\[
\begin{bmatrix}
\delta_e \\
\delta_a \\
\delta_r
\end{bmatrix}
= 
\begin{bmatrix}
a_1 & a_2 & a_3 \\
b_1 & b_2 & b_3 \\
c_1 & c_2 & c_3
\end{bmatrix}^{-1}
\begin{bmatrix}
A \\
B \\
C
\end{bmatrix}
= 
\begin{bmatrix}
\delta_e \\
\delta_a \\
\delta_r
\end{bmatrix},
\] (42)

or

\[
\delta_e = \frac{1}{\det T} \left[ (b_2 c_1 - c_2 b_1) A + (c_2 a_1 - a_2 c_1) B + (a_2 b_1 - a_1 b_2) C \right],
\]

\[
\delta_a = \frac{1}{\det T} \left[ (c_1 b_3 - b_1 c_3) A + (a_1 c_3 - c_1 a_3) B + (c_1 a_2 - a_1 c_2) C \right],
\]

\[
\delta_r = \frac{1}{\det T} \left[ (b_3 c_2 - c_3 b_2) A + (a_2 c_2 - c_2 a_2) B + (a_2 b_3 - b_2 a_3) C \right],
\] (43)

where

\[
\det T = a_2 b_3 c_3 + a_3 b_2 c_3 + b_2 c_3 a_3 - a_2 b_1 c_1 - b_1 c_1 a_1 - a_1 b_2 c_2 .
\] (44)
In conclusion, the paper author has successfully implemented the nonlinear attitude controller on the nonlinear UAV aircraft model. The attitude controller controls the attitude angles, the angular rates, the angular accelerations and other variables that describe the dynamics of UAV. Putting together equations (2), (4), (5), (12), (21), (34), (36), (39), (43), (44) and the expression of the six errors $e_\psi, e_\varphi, e_\theta, e_r, e_q, e_p$, the block diagram for the constructing of the attitude controller has been obtained (Fig. 4).

VI. SIMULATION RESULTS

In this section, I present the obtained results after performing the numeric simulations. The designed control system has been implemented in a Matlab/Simulink environment and its effectiveness has been tested with a campaign of numerical simulations. The Matlab/Simulink model of the structure in Fig. 4 is presented in Fig. 5.

The UAV has small dimensions: wing span $b = 1.7$ m, wing surface $S = 0.39$ m$^2$, mean aerodynamic chord $\bar{c} = 0.248$ m and mass $m = 3.2$ kg. The simulation data have been obtained from literature; thus, the values of the aerodynamic coefficients and of the inertia moments are [6]:

$$C_{\alpha 0} = 0, C_{\beta 0} = -5.3338 \times 10^{-4} + 33.094 \times 10^{-4} - 1.694,$$
$$C_{\alpha 4} = 34 \times 10^{-4} - 0.1287, C_{\beta 4} = 12.828 \times 10^{-4} - 0.458,$$
$$C_{\alpha 1} = -0.484, C_{\beta 1} = 0.17, C_{\alpha 6} = -0.35, C_{\beta 6} = 0.105,$$
$$C_{\alpha 2} = -0.23809, C_{\beta 2} = -0.002061, C_{\alpha 4} = -0.035424,$$
$$C_{\alpha 3} = 0.001833, C_{\beta 3} = -0.04778, C_{\alpha 5} = 0.06581,$$
$$I_x = 0.19 \text{ kg} \cdot \text{m}^2, I_y = 0.05 \text{ kg} \cdot \text{m}^2, I_z = 0.25 \text{ kg} \cdot \text{m}^2.$$

The trim conditions are very important because they are the starting values of the variables; the trim conditions are:

$$u = 18 \text{ m/s}, p = q = r = 0 \text{ deg/s}, \varphi = 2 \text{ grd},$$
$$\theta = -2 \text{ deg}, \psi = 5 \text{ deg}, \alpha = 1.24 \text{ deg}, \beta = 0.1 \text{ deg};$$
the desired values of the attitude angles have been chosen as:
$$\varphi = -5 \text{ deg}, \theta = 2 \text{ deg}, \psi = 3 \text{ deg}.$$

In Fig. 6, I present the time evolutions of the variables $\varphi(t), \theta(t), \psi(t), \rho(t), \sigma(t), r(t), \delta(t), \delta(t), \delta(t)$. One remarks the proper function of the backstepping method, because all the angles, which define the UAV attitude (the roll angle, the pitch angle and the yaw angle), track their imposed values.

![Fig. 7 Time variation of the UAV roll angle for different values of $\mu_\varphi$ and $\mu_\rho$ between 0.4 and 1.4](image)

![Fig. 8 Time variation of the UAV pitch angle for different values of $\mu_\psi$ and $\mu_\sigma$ between 0.4 and 1.4](image)

![Fig. 9 Time variation of the UAV yaw angle for different values of $\mu_\psi$ and $\mu_\rho$ between 0.4 and 1.4](image)
The choice of the values for the positive constants $\mu_p, \mu_q, \mu_v$ and $\mu_p, \mu_q, \mu_r$ influence the responses of the system. While the constants are positive, the system is stable; the bigger the values of constants are, the better the system properties are (the overshoot and the transient regime period decrease). In Fig. 7, Fig. 8 and Fig. 9, I present the time variations of the three attitude angles for different values of the constants $\mu_p, \mu_q$ and $\mu_v$ (values between 0.4 and 1.4).

The best results are obtained for the maximum value (in this case 1.4). For example, if one considers the case of the roll angle stabilization, for $\mu_q = 0.4, \mu_p = 0.4$, the overshoot is 2 deg, while, for $\mu_q = 1.4, \mu_p = 1.4$, it decreases ten times to the value 0.2 deg. The same conclusion may be obtained regarding the transient regime period – the increase of the constants $\mu_q$ and $\mu_p$ from the values 0.4 to the values 1.4 leads to the decrease of the transient regime period from 13 seconds to 5 seconds (61.53% decrease); same remarks for the pitch and yaw angles.

VII. CONCLUSIONS

The focus of the backstepping controller synthesis process was the choice of the error variables and of the Lyapunov functions. The deflections of the control surfaces have been obtained – equations (43) and (44). The obtained attitude controller may be implemented on UAVs; the technique that has been used for its design is a general one and may be used to all kind of flying objects, offering very good operation and characteristics of the output signals.

The design of the three controllers and the backstepping technique has directly used the nonlinear equations and the Lyapunov analysis. The method has an important advantage with respect to the classical techniques (for example the design of PID controllers): the nonlinear system (describing the dynamics of UAV) must not be linearised in a trim point. This linearization must be done if we want to design PID controllers. The linearization of the UAV dynamics limits the applicability of PID controllers.

Using the backstepping method, the author obtained the expressions of the elevator, rudder and aileron deflections that control the UAV, at every moment, and stabilize it to the desired values of the attitude angles and angular rates.

The paper author made complex simulations in Matlab/ Simulink by using data from the UAV flight tests. The obtained results are very good (UAV attitude angles and the UAV angular rates tend to their imposed values in a proper transient regime without big overshoots).

While the constants $\mu_p, \mu_q, \mu_v$ and $\mu_p, \mu_q, \mu_r$ have positive values, the system is stable; the bigger the values of constants are, the better the system properties are (the overshoot and the transient regime period decrease).

ACKNOWLEDGMENT

This work was supported by CNCSIS-UEFISCSU, project PN II-RU “High-precision strap-down inertial navigators, based on the connection and adaptive integration of the nano and micro inertial sensors in low cost networks, with a high degree of redundancy”, code102/2010.

REFERENCES


