Automatic approach procedure of a flying vehicle on a mobile platform using backstepping controller

Florin COSTACHE*,[1], Sandra-Elena NICHIFOR*,[1], Mihaela-Luminita COSTEA[1], Achim IONITA[1]

*Corresponding author
[1]INCAS – National Institute for Aerospace Research "Elie Carafoli",
B-dul Iuliu Maniu 220, Bucharest 061126, Romania,
costache.florin@incas.ro, nichifor.sandra@incas.ro, costea.mihaela@incas.ro,
ionita.achim@incas.ro

DOI: 10.13111/2066-8201.2022.14.4.5

Received: 2 August 2022/ Accepted: 27 October 2022 / Published: December 2022

Abstract: *This paper presents the automatic approach procedure of a flying vehicle, attached to an ABB 7600 robot, and a mobile platform, attached to a Stewart platform. Due to a nonlinear dynamic behavior, it is necessary to implement complex control, stabilization and guidance schemes. The proposed solution for this system includes the development of an algorithm based on a backstepping control method, the controller design methodology being based on Lyapunov's stability theory. The proposed command law requires that the states are known, but it is also necessary to introduce a series of state estimators. Tracking a mobile platform is critical in surveillance, reconnaissance and tracking missions, with the control methodology defining a clear distinction between translational and rotational dynamics. The proposed algorithm is developed by separating two types of states involving an inverse kinematics, known as algebraic kinematics, in which the dynamic movements of the two pieces of equipment are used. The dynamics of the ABB 7600 robot involves a movement with seven degrees of freedom, while the Stewart platform can be used with a movement of six degrees of freedom. The proposed algorithm is implemented in both Matlab software and experimental testing. This paper provides results in terms of generating dynamics for both devices that can be used for simulating different scenarios of aerospace missions.*

Key Words: *automatic approach, ABB 7600 Robot, Stewart Platform, backstepping, inverse kinematics, Control Lyapunov Function*

1. INTRODUCTION

Recent developments in miniaturizing equipment and nonlinear techniques have shown that unmanned aerial vehicles (UAVs) already have a wide range of possible applications in both civil and military fields, due to reduced manufacturing costs as well as operational costs, compared to manned vehicles. Their potential uses are diversified, including border patrol missions, search, fire monitoring and road traffic. Generally, UAVs are preferred for missions that are considered dangerous and for missions with long pilot effort [1-4]. Most of the aforementioned missions require vertical take-off and landing (VTOL). The ABB 7600 Robot (Figure 1) offers and alternative in executing flight phases specific to fixed-wing vehicles (long distance flights, tracking, monitoring, identification) as well as vehicles with rotary wing (hovering, VTOL, monitoring, identification) [1-4].

Fig. 1 ABB 7600 Robot

The development in complex systems that allow UAV vehicles to perform autonomous movements is a distinct direction in aerospace research. Specific to this area of research is the realization of autonomous approaching techniques of an ABB 7600 Robot on a Stewart Platform (SP). Due to the nonlinear dynamic behavior, it is necessary to implement complex control, stabilization and guidance schemes. The proposed solution for this system includes the development of an algorithm based on a backstepping control method, the use of GPS, IMU and video camera, the latter requesting the estimation of the states based on a filtering technique. The design methodology of the controller is based on Lyapunov's stability theory [1-4].

In a first stage, this paper proposes a control algorithm for autonomous landing of a hybrid aerial vehicle (airplane/quadcopter), represented by the ABB 7600 Robot, on a moving Stewart Platform. The imposed control law requires knowledge of the states, but it is necessary to introduce some state estimators.

The numerical simulations for the ABB 7600 Robot dynamics regarding the SP movement, suggest that the proposed control scheme (Figure 2) is adequate for implementation in the control system of the hybrid platform for landing on a moving SP.

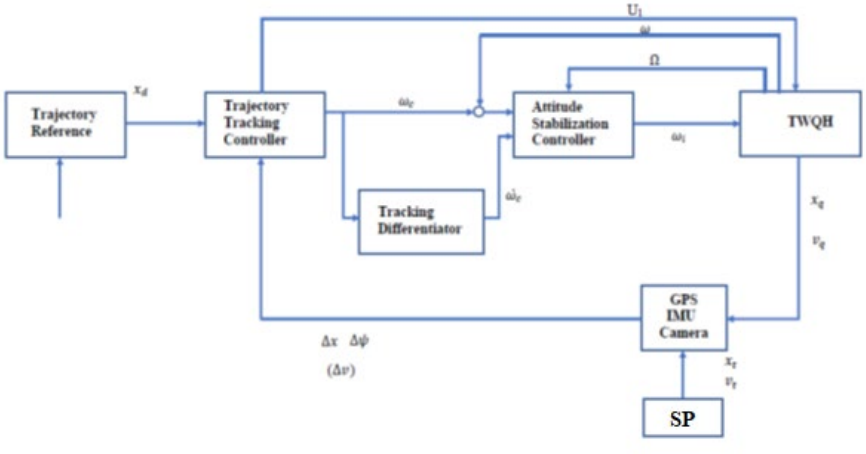

Fig. 2 Control Scheme

2. APPROACHING TECHNIQUES ON MOBILE PLATFORM

The main goal of this activity is to develop, implement and experiment a fully automated approaching system on a mobile platform using a sensor structure (GPS, IMU, camera-vision system). A limitation on the use of the visualization system (camera) when it is pointed downwards is the difficulty in distinguishing the rotational and the translational movements in the image from the perspective projection. For autonomous approaching, ABB 7600 Robot must have full GNC capabilities [1-3].

The measurement of the orientation of the aerial vehicle is solved using an IMU sensor (gyro meter + accelerometer + magnetometer) and a GPS for position. Using the GPS sensor for precise landing is not an acceptable solution because it has a measurement error up to several meters. An alternative solution is to use a camera (computer vision system). We distinguish two visualization techniques: direct and indirect visualization [1-3].

Control strategies used: [1-3]

- Positioning using a 3D function expressed in the Euclidian reference system;
- Positioning using 2D measures extracted from processed images;
- Positioning through the combined use of the aforementioned techniques.

The SP must be equipped with an identification sensor (blob – tracking approach).

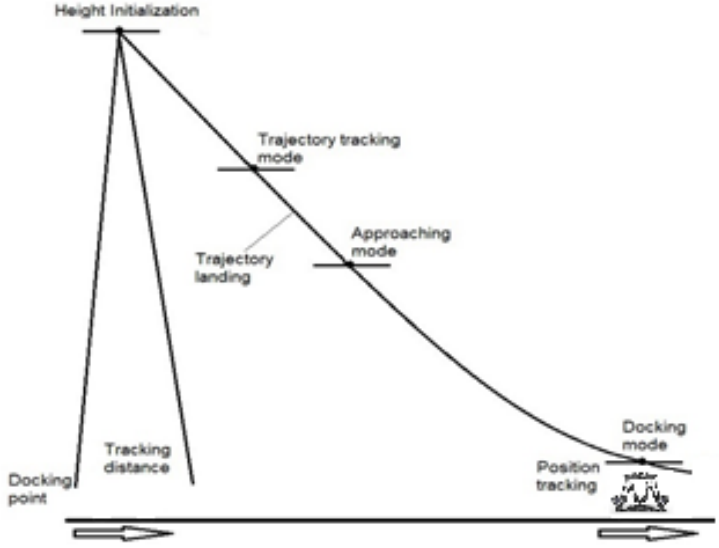

Fig. 3 Approaching Procedure

Stages of approaching procedure (Figure 3 and Figure 4): [1-3]

1. Searching phase (observation/initialization)

When initializing the procedure, the ABB 7600 Robot can locate the SP platform using its coordinates via the GPS sensor (navigation mode). With the help of the camera (FOV) the altitude can be easily estimated (MSL). The altitude must be estimated continuously when the landing procedure is activated. Switching to "vision based control" function is required when the SP has been identified. [1-3]

2. Tracking phase

In this mode, the ABB 7600 Robot is aligned to the center (the reference point) of the SP platform and begins to descend to the required altitude. The altitude is measured by the pressure sensor (inertial altitude). It is necessary to take measures (implemented algorithm)

when it is possible to lose the SP position (due to adverse weather conditions, the SP image may disappear in the tracking process) [1-3].

3. Observation phase

The estimation of the movement of the SP platform is necessary for the activation of the approaching phase, since the descent begins. Also, the loss of the SP position must be taken into account. It is possible that the two platforms will not be aligned and the alignment process must be resumed [1-3].

4. Approaching phase

In this mode, the descent speed must be specified and it is necessary to mention how the ABB 7600 Robot reaches the contact point. It is possible to miss the approach due to image loss (due to atmospheric turbulence) [1-3].

The ABB 7600 Robot will be equipped according to the autonomous approaching strategy with an IMU sensor, an accurate altitude sensor (based on pressure measurement) and a camera. Consequently, it is necessary to implement a state estimation block based on a filtering technique (controller + state estimation) [1-3].

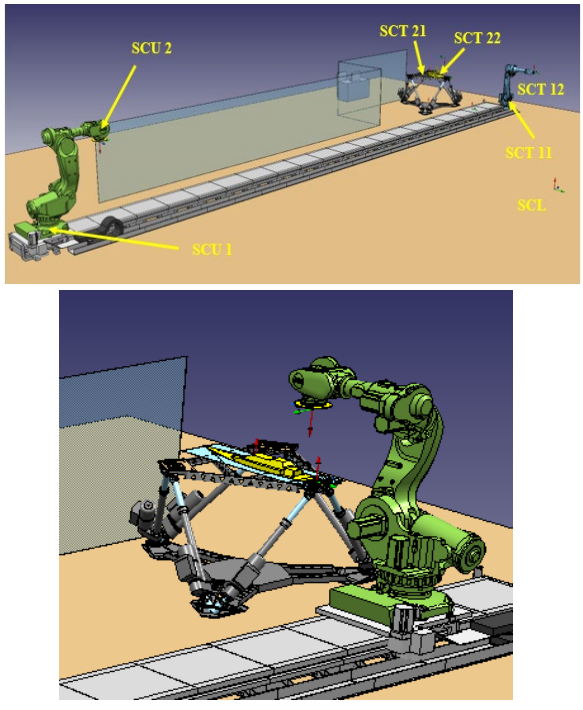

Fig. 4 Approaching Procedure

3. TWQH DYNAMIC MODELLING

3.1 Working hypotheses [1-3]

- For this study, three main reference systems have been used: aerodynamic system, body system and Earth system, defined in STAS 10276/1-75.
- The body of the plane is rigid.
- The center of gravity of the quadrotor (QH) coincides with the center of gravity of the airplane (TW).

- The forces and moments of the electric rotors are proportional to the square of the angular velocity of propellers.
- The hybrid platform has a symmetry plan (xOz).

3.2 Equations of motions [1-3]

$$\dot{\bar{X}} = C\bar{V}$$
$$\dot{\bar{\omega}} = B^{-1}\bar{\Omega} \; or \; \Omega = B\dot{\omega}$$
$$m(\dot{\bar{V}} - S(\Omega)\cdot\bar{V}) = \overline{F_G} + \overline{F_A} + \overline{F_T}$$
$$I\dot{\bar{\Omega}} + S(\Omega)\left(I\cdot\bar{\Omega} + \overline{h_r}\right) = \overline{M_A} + \overline{M_T}$$

where: $C, B, X, V, F_G, F_T, F_A, \Omega, \omega, M_T, M_A$ are known;

$$C = \begin{bmatrix} c\theta c\psi & c\psi s\theta s\phi - s\phi c\phi & c\psi s\theta c\phi + s\psi s\phi \\ s\psi c\theta & s\psi s\theta s\phi + c\psi c\phi & c\psi s\theta c\phi - c\psi s\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix};$$
$$B = \begin{bmatrix} 1 & 0 & -s\theta \\ 0 & c\phi & c\theta s\phi \\ 0 & -s\phi & c\theta c\phi \end{bmatrix};\; \bar{X} = \begin{pmatrix} x \\ y \\ z \end{pmatrix};\; \bar{V} = \begin{pmatrix} u \\ v \\ w \end{pmatrix};$$
$$V = \sqrt{\dot{x}^2 + \dot{y}^2 + \dot{z}^2};$$
$$\overline{F_G} = C\cdot\begin{pmatrix} 0 \\ 0 \\ mg \end{pmatrix};\; \overline{F_T} = \begin{pmatrix} 0 \\ 0 \\ -U_1 \end{pmatrix};\; \overline{F_A} = \begin{pmatrix} X_A \\ Y_A \\ Z_A \end{pmatrix};$$
$$\bar{\Omega} = \begin{pmatrix} p \\ q \\ r \end{pmatrix};\; \bar{\omega} = \begin{pmatrix} \varphi \\ \theta \\ \psi \end{pmatrix};$$
$$\overline{M_T} = \begin{pmatrix} U_2 \\ U_3 \\ U_4 \end{pmatrix};\; \overline{M_A} = \begin{pmatrix} L_A \\ M_A \\ N_A \end{pmatrix}.$$

The linearized equation system is:

$$\dot{u} = rv - qw - g\theta + X_A$$
$$\dot{v} = pw + ru + g\phi + Y_A$$
$$\dot{w} = qu - pv + g + Z_A + \frac{1}{m}U_1$$
$$\dot{p} = i_5 qr + i_2 L_A - i_8 q + i_2 U_2$$
$$\dot{q} = i_6 rp + i_3 M_A + i_9 p + i_3 U_3$$
$$\dot{r} = i_7 pq + i_4 N_A + i_4 U_4$$
$$\dot{\phi} = p + q\phi\theta + r\theta$$
$$\dot{\theta} = q - r\phi$$
$$\dot{\psi} = q\phi + r$$
$$\dot{x} = u + v\psi + w\theta$$
$$\dot{y} = u(\varphi\theta - \psi) + v(\varphi\theta\psi + 1) + w\varphi$$
$$\dot{z} = u(\theta + \varphi\psi) + v(\theta\,\psi - \varphi) + w$$

where:

$$h_{rz} = I_r \Omega_r$$

$$\Omega_r = (\omega_1 - \omega_2 + \omega_3 - \omega_4)$$

$$i_1 = \frac{1}{m}; \quad i_2 = \frac{1}{I_{xx}}; \quad i_3 = \frac{1}{I_{yy}}; \quad i_4 = \frac{1}{I_{zz}}; \quad i_5 = \frac{I_{yy} - I_{zz}}{I_{xx}}$$

$$i_6 = \frac{I_{zz} - I_{xx}}{I_{yy}}; \quad i_7 = \frac{I_{xx} - I_{yy}}{I_{zz}}; \quad i_8 = \frac{I_r \Omega_r}{I_{xx}}; \quad i_9 = \frac{I_r \Omega_r}{I_{yy}}$$

Control forces are:

$$U = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} c_f(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \\ c_f(\omega_1^2 - \omega_2^2 - \omega_3^2 + \omega_4^2) \\ c_f[a(\omega_1^2 + \omega_2^2) - b(\omega_3^2 + \omega_4^2)] \\ c_M(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \end{bmatrix}$$

The rotor speeds will be extracted from the relations:

$$\omega_1^2 = \frac{1}{4(a+b)}\left(\frac{2b}{c_f}U_1 + \frac{a+b}{c_f l}U_2 + \frac{2}{c_f}U_3 + \frac{a+b}{c_M}U_4\right)$$

$$\omega_2^2 = \frac{1}{4(a+b)}\left(\frac{2b}{c_f}U_1 - \frac{a+b}{c_f l}U_2 + \frac{2b}{c_f}U_3 - \frac{a+b}{c_M}U_4\right)$$

$$\omega_3^2 = \frac{1}{4(a+b)}\left(\frac{2a}{c_f}U_1 - \frac{a+b}{c_f l}U_2 - \frac{2}{c_f}U_3 + \frac{a+b}{c_M}U_4\right)$$

$$\omega_4^2 = \frac{1}{4(a+b)}\left(\frac{2a}{c_f}U_1 + \frac{a+b}{c_f l}U_2 - \frac{2}{c_f}U_3 - \frac{a+b}{c_M}U_4\right)$$

3.3 Electric rotor dynamics

The system of equations of the DC motor is:

$$L\frac{di}{dt} = u - Ri - k_e\omega$$

$$I_R\frac{d\omega}{dt} = k_m i - k_r\omega^2 - k_s$$

where:

$$i = \frac{1}{R}(u - k_e\omega)$$

Rewriting the equation of ω:

$$\frac{d\omega}{dt} = \frac{1}{I_R}\left[k_m\frac{1}{R}(u - k_e\omega) - k_r\omega^2 - k_s\right]$$

$$\frac{d\omega}{dt} = -\frac{k_s}{I_R} - \frac{k_r}{I_R R}\omega^2 - \frac{k_e}{I_R}k_m\omega + \frac{k_m}{I_R R}u$$

which leads to:

$$\frac{d\omega}{dt} = -a_0 - a_1\omega - a_2\omega^2 + bu$$

4. BACKSTEPPING CONTROL

The backstepping control methodology makes a clear distinction between translational and rotational dynamics. The variables of the translation states influence the variables of the attitude states and not vice versa. The control architecture in backstepping by separating the two types of states implies an inverse kinematics which is in fact an algebraic kinematics. Robustness at high disturbances can be achieved by using a fully backstepping control technique. The backstepping control technique is based on Lyapunov's stability method [6-8]. The objective is to ensure the convergence of the states, $\boldsymbol{x}(t), y(t), z(t)\ and\ \psi(t)$, to the desired trajectories provided by the variables, $x_d(t), y_d(t), z_d(t)\ and\ \psi_d(t)$, thus stabilizing the attitude angles $\phi(t)$ and $\theta(t)$ [6-8].

State variables:

$$[x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11}, x_{12}] \text{ or } \left[\phi, \dot{\phi}, \theta, \dot{\theta}, \psi, \dot{\psi}, z, \dot{z}, x, \dot{x}, y, \dot{y},\right]$$

Command variables: [6-8]

$$[U_1, U_2, U_3, U_4]$$

The following linearized system is obtained: [6-8]

$$\dot{x_1} = x_2$$
$$\dot{x_2} = i_5 x_4 x_6 - i_8 x_4 + i_2 L_A + i_2 U_2$$
$$\dot{x_3} = x_4$$
$$\dot{x_4} = i_6 x_2 x_6 + i_9 x_2 + i_3 M_A + i_3 U_3$$
$$\dot{x_5} = x_6$$
$$\dot{x_6} = i_7 x_2 x_4 + i_4 N_A + i_4 U_4$$
$$\dot{x_7} = x_8$$
$$\dot{x_8} = g + F_{Z_A} - i_1 U_1 cosx_1 cosx_3$$
$$\dot{x_9} = x_{10}$$
$$\dot{x_{10}} = F_{X_A} - i_1 U_1 (cosx_1 sinx_3 cosx_5 + sinx_1 sinx_5)$$
$$\dot{x_{11}} = x_{12}$$
$$\dot{x_{12}} = F_{Y_A} - i_1 U_1 (cosx_1 sinx_3 sinx_5 - sinx_1 cosx_5)$$

- **Step 1**. The first tracking error is considered $z_1 = x_{1d} - x_1$. In order to obtain the stability of the equilibrium point $z_1 = 0$, a candidate Lyapunov function is described as $V_1 = \frac{1}{2} z_1^2$. Deriving the function, the following equation is obtained: $\dot{V}_1 = z_1 \left(\dot{x_{1_d}} - x_2\right)$ [6-8].
  The stabilization of z_1 can be achieved by introducing a virtual control law, $x_2 = \dot{x_{1d}} + \alpha_1 z_1$ with $\alpha_1 \in R_+$, being a gain to be chosen according to the desired transient performance. Then, $\dot{V}_1 = -\alpha_1 z_1^2 < 0$ [6-8].
- **Step 2**. To ensure the convergence of state x_2, the tracking error is set for the virtual control variable, $z_2 = x_2 - \dot{x_{1d}} - \alpha_1 z_1$, resulting in $\dot{z_1} = -\alpha_1 z_1 - z_2$. For this step the following candidate Lyapunov function is considered: $V_2 = \frac{1}{2} \sum_1^2 z_i^2$ [6-8].
  Its time derivative is: $\dot{V}_2 = -\alpha_1 z_1^2 + z_2[-z_1 + i_5 x_4 x_6 + i_2 U_2 - i_2 L_A + i_8 x_4 + \alpha_1 (z_2 + \alpha_1 z_1)]$ [6-8] The function is negatively defined if the Euler angles vary slowly ($\ddot{x}_{1d} = 0$). Therefore, the z_2 stabilization can be obtained by introducing the

following control law: $U_2 = \frac{1}{i_2}[z_1 - i_5 x_4 x_6 - i_2 L_A + i_8 x_4 - \alpha_1(z_2 + \alpha_1 z_1) - \alpha_2 z_2]$; $\alpha_2 \in R_+$. The result is: $\dot{V}_2 = -\sum_1^2 \alpha_i z_1^2 < 0$, which is asymptotically stable [6-8].

- **Step 3**. For the third step, the tracking error $z_3 = x_{3d} - x_3$ and the Lyapunov candidate function $V_3 = \frac{1}{2} z_3^2$, are taken into consideration. The stabilization of z_3 can be obtained by introducing a virtual control law $x_4 = \dot{x}_{3d} + \alpha_3 z_3$, $\alpha_3 \in R_+$, resulting a negative defined function: $\dot{V}_3 = -\alpha_3 z_3^2 < 0$ [6-8].
- **Step 4**. Let be an associated access error $z_4 = x_4 - \dot{x}_{3d} - \alpha_3 z_3$, which has the result $\dot{z}_3 = -z_4 - \alpha_3 z_3$, and a candidate Lyapunov function is defined: $V_4 = \frac{1}{2}\sum_3^4 z_i^2$. The stabilization of z_4 can be obtained from the following control law: $U_3 = \frac{1}{i_3}(z_3 - i_6 x_2 x_6 - i_3 M_A - i_9 x_2 - \alpha_3(z_4 + \alpha_3 z_3) - \alpha_4 z_4)$, $\alpha_3, \alpha_4 \in R_+$, which leads to $\dot{V}_4 = -\sum_3^4 \alpha_i z_i^2 < 0$ [6-8].
- **Step 5**. According to the same procedure as the previous step, the tracking error $z_5 = x_{5d} - x_5$ and the virtual control law $x_6 = \dot{x}_{5d} - \alpha_5 x_5$, $\alpha_5 \in R_+$. Therefore, $V_5 = \frac{1}{2} z_5^2$ and $\dot{V}_5 = -\alpha_5 z_5^2 < 0$ [6-8].
- **Step 6**. Let be the tracking error $z_6 = x_6 - \dot{x}_{5d} - \alpha_5 z_5$, resulting $\dot{z}_5 = -z_6 - \alpha_5 z_5$ and the candidate Lyapunov function $V_6 = \frac{1}{2}\sum_5^6 z_i^2$. The control law obtained is: $U_4 = \frac{1}{i_4}[z_5 - i_7 x_2 x_4 - i_4 N_A - \alpha_5(z_6 + \alpha_5 z_5) - \alpha_6 z_6]$, $\alpha_6 \in R_+$, which leads to: $\dot{V}_6 = -\sum_5^6 \alpha_i z_i^2 < 0$ [6-8].
- **Step 7**. For this procedure, the altitude dynamics is taken into consideration. In this sense, the chosen tracking error is $z_7 = x_{7d} - x_7$. The virtual control input can be described as $x_8 = \dot{x}_{7d} + \alpha_7 x_7, \alpha_7 \in R_+$, which leads to: $\dot{V}_7 = -\alpha_7 z_7^2 < 0$ and $V_{z_7} = \frac{1}{2} z_7^2$ [6-8].
- **Step 8**. Let be associated error $z_8 = x_8 - \dot{x}_{7d} - \alpha_7 z_7$, resulting in $\dot{z}_7 = -z_8 - \alpha_7 z_7$, and the candidate Lyapunov function $V_8 = \frac{1}{2}\sum_7^8 z_i^2$. The control law is described as: $U_1 = \frac{1}{i_1 cosx_3 cosx_4}\left[-z_7 + g + F_{Z_A} - \alpha_7(z_8 + \alpha_7 z_7) + \alpha_8 z_8\right]$, $\alpha_8 \in R_+$. Knowing that $\alpha_8 \in R_+$ and $cos(x_3)\, cos(x_1) > 0$, the result is: $\dot{V}_8 = -\sum_7^8 \alpha_i z_i^2 < 0$ [6-8].
- **Step 9**. Unlike the degrees of freedom of attitude, the dynamics of the horizontal plane position is not directly controlled using the U_i inputs. The control of the dynamics of the position must be done using virtual laws, μ_x and μ_y. For these, let be the tracking error $z_9 = x_{9d} - x_9$. The virtual control input $x_{10} = \dot{x}_{9d} + \alpha_9 z_9$, which is associated with the error $z_{10} = x_{10} - \dot{x}_{9d} - \alpha_9 z_9$, has the result $\dot{z}_9 = -z_{10} - \alpha_9 z_9$. Finally, the candidate Lyapunov function is defined: $V_9 = \frac{1}{2}\sum_9^{10} \alpha_i z_i^2$, $\alpha_9, \alpha_{10} \in R_+$, and the control law can be describes as: $\mu_x = \frac{1}{i_1 U_1}\left[-z_9 + F_{X_A} + \alpha_9(z_{10} + \alpha_9 z_9) + \alpha_{10} z_{10}\right]$. For the position stability on the x axis, knowing that $\alpha_9, \alpha_{10} \in R_+$ and $U_1 > 0$ are positively defined, the result is: $\dot{V}_9 = -\sum_9^{10} \alpha_i z_i^2 < 0$ [6-8].
- **Step 10**. Similarly, the tracking error is defined $z_{11} = x_{11d} - x_{11}$, the virtual control input $x_{12} = \dot{x}_{11d} + \alpha_{11} z_{11}$, associated error $z_{12} = x_{12} - \dot{x}_{11d} - \alpha_{11} z_{11}$. A candidate Lyapunov function is chosen as follows: $V_{10} = \frac{1}{2}\sum_{11}^{12} z_i^2$, obtaining the following

control law: $\mu_y = \frac{1}{i_1 U_1}\left[-z_{11} + F_{Y_A} + \alpha_{11}(z_{12} + \alpha_{11} z_{11}) + \alpha_{12} z_{12}\right]$, $\alpha_{11}, \alpha_{12} \in R_+$ [6-8].

Therefore, the position is stabilized along the y axis for $\alpha_{11}, \alpha_{12} \in R_+$ and $U_1 > 0$, thus obtaining: $\dot{V}_{10} = -\sum_{11}^{12} \alpha_i z_i^2 < 0$, where we initially noted: $\mu_x = cosx_1 sinx_3 cosx_5 + sinx_1 sinx_5$ and $\mu_y = cosx_1 sinx_3 sinx_5 - sinx_1 cosx_5$ [6-8].

From the equations of μ_x and μ_y from steps 9 and 10, the reference signs for the angles ϕ_d and θ_d, the horizontal plane can be expressed by: $x_{1d} = atan\left(\frac{\mu_x sinx_5 - \mu_y cosx_5}{\Delta}\right)$ and $x_{3d} = asin\left(\frac{\mu_x cosx_5 + \mu_y sinx_5}{\Delta}\right)$, where $\Delta = \sqrt{1 - sin^2 x_5\, \mu_x^2 + 2\mu_x \mu_y sinx_5 cosx_5 - \mu_y^2 cos^2 x_5}$ [6-8].

Consequently, from the control laws formulated in the previous stages, the closed loop UAV control results are in a form of an asymptotic stable system [6-8].

5. RESULTS

In order to verify the effectiveness and the efficiency of the proposed backstepping control law. An application of the TWQH vehicle, represented by the ABB 7600 Robot, is conducted by simulation using Runge – Kutta 4th order integration method.

The following parameters for simulations have been used:

- Mass and inertias: $m = 36kg, I_{xx} = 3.36kg * m^2, I_{yy} = 9.88kg * m^2, I_{zz} = 12.35kg * m^2$;
- Rotors characteristics: $I_r = 0.007kg * m^2, \Omega_r = 341.5rad/s, c_f = 0.00076N/(rad/s)^2, c_q = 0.0000112N * m/(rad/s)^2$;
- Geometric parameters: $S = 0.94m^2, c = 0.33m, b = 3m, aa = 0.57m, bb = 0.67m, cc = 1.1m.$

The initial location of the ABB 7600 Robot is at (x_0, y_0, z_0) = (0, 0, 2.445) m and SP location is at (x_d, y_d, z_d) = (2.6, 23, 1.655) m. For this simulation, only 30 points have been implemented in the robot's system. Figure 5 represent the trajectory of the ABB 7600 Robot as it approaches the Stewart Platform with its own movement. Therefore, the ABB robot starts to follow the SP dynamics in order to have a synchronous movement.

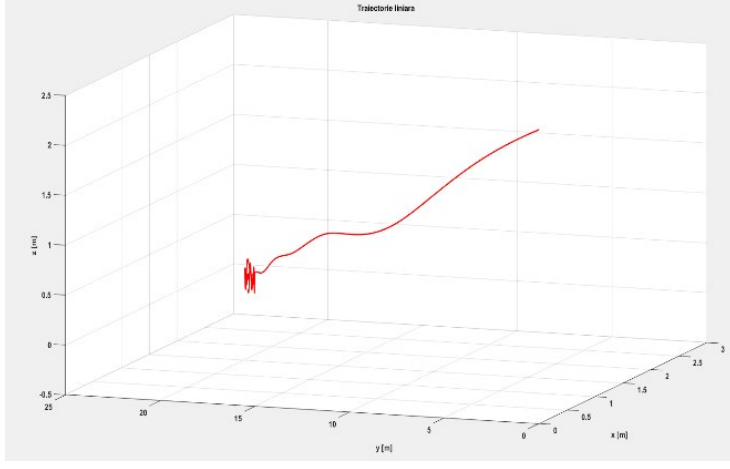

Fig. 5 ABB 7600 Robot Trajectory

Furthermore, the trajectory has been simulated both offline and online on the ABB 7600 Robot. The initial and desired locations are as follows: (x_0, y_0, z_0) = (10337, 2604.6, 571.05) mm and (x_d, y_d, z_d) = (592, 2648.4, 3.7) mm

Figure 6 shows the trajectory of the ABB 7600 Robot as it is calculated in Matlab (red line) and the real trajectory implemented offline on the ABB Robot in RobotStudio environment. It can be seen that the two trajectories almost coincide, the differences between them are very small, as shown in Figure 7, thus the backstepping control methodology is suitable for this case and it can be implemented online on the real ABB 7600 Robot (Figures 8 and 9).

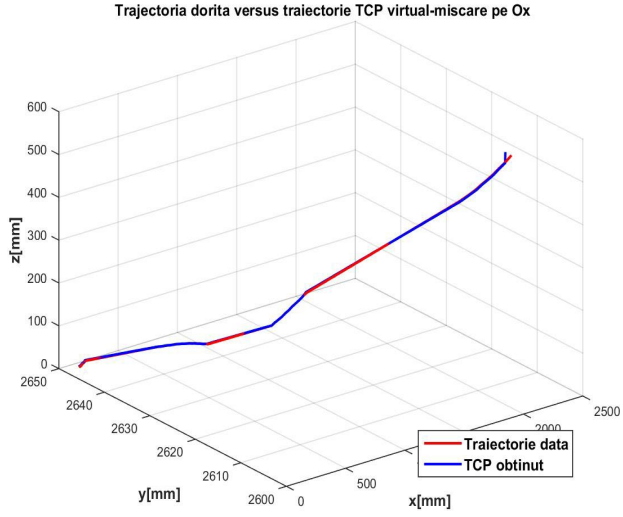

Fig. 6 Calculated vs. Real Trajectory offline

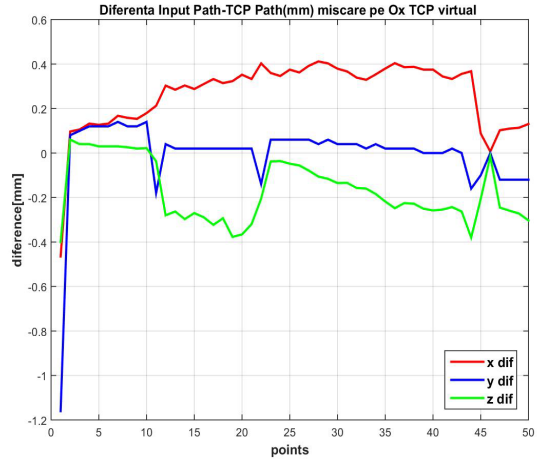

Fig. 7 Difference between calculated and real trajectory offline

Figure 8 shows the trajectory of the ABB 7600 Robot as it is calculated in Matlab (red line) and the online real trajectory implemented on the ABB Robot.

It can be seen that the two trajectories almost coincide, the differences between them are very small, as can be seen in Figure 9, thus the backstepping control methodology is suitable for this case.

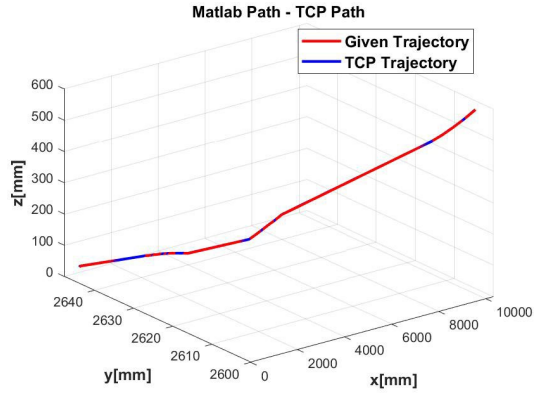

Fig. 8 Calculated vs. Real Trajectory online

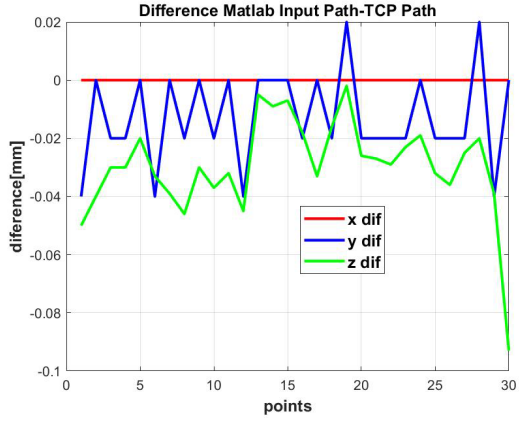

Fig. 9 Difference between calculated and real trajectory online

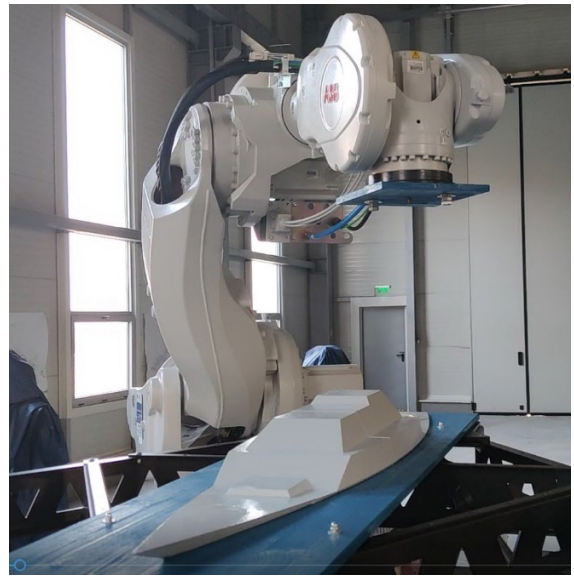

Fig. 10 Approaching phase

Figure 10 shows the end of the trajectory process of the real ABB 7600 Robot on the mobile Stewart platform.

6. CONCLUSIONS

The numerical simulations for the ABB 7600 Robot dynamics in relation to the SP movement suggest that the proposed control scheme is suitable for implementation in the hybrid platform control system for autonomous landing on the moving ground vehicle.

The advantage of backstepping is its robustness to external disturbances and uncertainties, while its disadvantages are that it is non-optimal and time inefficient.

ACKNOWLEDGEMENTS

This paper is part of the National Nucleu Program – Advanced research to increase the competitiveness and capacity of design, analysis and specific expertise in the aerospace field – AEROEXPERT 2019-2022 project entitled "Technological demonstrator for autonomous control of landing on mobile platforms – ALAMOPLAT", contract number 8N/07-02-2019, developed by the National Institute for Aerospace Research "Elie Carafoli" – INCAS Bucharest.

This article is an extension of the paper presented at *The 39th "**Caius Iacob**" Conference on Fluid Mechanics and its Technical Application*, 28-29 October 2021, Bucharest, Romania.

REFERENCES

[1] N. Apostolescu, M. L. Costea, F. Costache, A. Lungoci and S. E. Nichifor, *FAZA nr. 1: Analiza sistem de control autonom vehicule aerospatiale*, INCAS Bucuresti, Bucuresti, 2020.

[2] R. Bogățeanu, N. Apostolescu, F. Costache, A. Lungoci, S. E. Nichifor, A. Burghiu, D. Vișan and M. L. Costea, *FAZA nr. 2: Modelare sistem de control autonom vehicule aerospatiale*, INCAS Bucuresti, Bucuresti, 2021.

[3] R. Bogateanu, A. Lungoci, S. Nichifor, N. Apostolescu and M. Costea, *Proiect de ansamblu demonstrator tehnologic ALAMOPLAT (software/hardware) elemente hardware-senzori si componente*, INCAS Bucuresti, Bucuresti, 2020.

[4] * * * INCAS, *Proiect Demonstrator tehnologic pentru controlul autonom al aterizării pe platforme mobile – ALAMOPLAT*, Contract nr. 8N/07.02.2019 Cod Proiect: PN 19 01 02 04, Bucuresti: INCAS, 2018.

[5] * * * INCAS, *Platforme UAV (Vehicule Aeriene fără Pilot) cu capabilități dedicate și infrastructură suport pentru aplicații în misiuni de securitate națională*, INCAS București, București, 2018.

[6] S. Bouabdallah and R. Siegwart, *Backstepping and Sliding-mode Techniques Applied to an Indoor Micro Quadrotor*, in IEEE, Barcelona, 2005.

[7] A. S. Sanca, *Modelagem e controle de um microveiculo aereo: uma aplicacao de estabilidade robusta com a tecnica backstepping em uma estrutura hexarrotor*, Universidade Federal do Rio Grande do Norte , Natal, 2013.

[8] S. Bouabdallah, *Design and control of quadrotors with application to autonomous flying*, Ecole Polytechnique Federale de Lausanne, Lausanne, 2007.