International Journal Of Engineering Research And Development
e- ISSN: 2278-067X, p-ISSN: 2278-800X, www.ijerd.com
Volume 20, Issue 8 (August, 2024), PP. 511-521

Developing an optimal landing algorithm for fixed-wing Unmanned Aerial Vehicles (UAVs) utilizing the principle of Pontryagin's maximum

Minh Truong Trinh, Duy Tuyen Khuat, Minh Ngoc Nguyen, Hong Son Tran[1]
[1]*Faculty of Aviation Engineering, Air defense-Air Force Academy, Hanoi, Vietnam*
Corresponding Author: Hong Son Tran

ABSTRACT
Building a landing trajectory for a UAV is an important factor in ensuring the safety and efficiency of automated flights. When faced with challenges such as changes in weather and environmental conditions, as well as high accuracy requirements during landing, the Pontryagin principle becomes an effective optimization tool. This principle provides a method for determining the optimal landing trajectory through the establishment of optimal conditions and maximal functions. In this paper, the optimization technique based on the Pontryagin principle is applied to build the landing trajectory for the UAV. The simulations performed by Matlab - Simulink software show the effectiveness of this method in improving accuracy and minimizing risk during landing.

Date of Submission: 23-08-2024 | Date of Acceptance: 07-09-2024

I. INTRODUCTION

An unmanned Aerial Vehicle (UAV) is a flying device that is controlled and remotely operated by humans on the ground. It plays an important role in many areas of social life, commercial and entertainment activities as well as military and defense activities. In general, automatic control in all UAV operations is very important, especially the take-off and landing process. Because this process is affected by many factors such as weather, UAV operating status parameters, etc., and UAVs are most susceptible to unsafe failures during this stage. In their research, the authors focused on studying the landing process of UAVs. The landing process is the stage of the UAV gradually slowing down from the specified height until it stops completely on the runway. When the UAV lands on the runway, it must also move back to the parking lot, so when the UAV reaches a rolling speed (about 5km/h), it is considered the end of the landing process. The landing stages of a fixed-wing UAV are shown specifically in Figure 1.

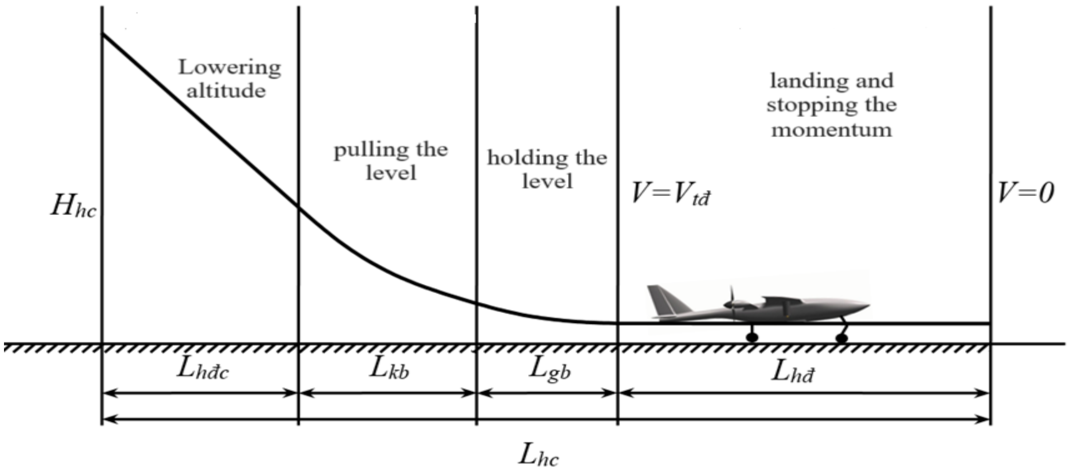

Figure 1. Diagram of UAV landing stages

The paper presents the problem of constructing optimal landing trajectories for UAVs in the vertical plane. The landing trajectories are built in the case of no vertical overload restriction and the case of vertical overload restriction to create a reference landing trajectory to construct automatic systems for the programmatic landing of UAVs. With the method of constructing optimal trajectory proposed by the group of authors, the group hopes that their research can be applied in practice to develop automation of UAV operation stages.

II. MATERIAL AND METHODS

2.1. Developing the problem of optimizing the landing trajectory for UAV-70V

Pontryagin maximum principle

Pontryagin's maximal principle is a mathematical method developed by Pontryagin to solve the optimal problem. In particular, the focus is on proposing mathematical modeling methods and building concise results on strong optimal necessary conditions. When considering the optimal control problem, Pontryagin's principle will predefine the control vector. This is in line with the principle layer of maximum variation. Therefore, this principle is often used in practice. Pontryagin's maximal principle presents a series of optimal conditions, which are the basis for determining optimal control and optimal trajectory. Pontryagin's maximal principle focuses on solving the optimal problem with fixed or non-fixed boundaries, and times with limited control signals.

To solve the problem of optimizing the landing trajectory of UAVs, it is necessary to determine and select quality indicators appropriately. The main quality indicator is a quality indicator of the control system, which is given in the form of:

$$J = J[x(t_0), x(t_f); u(t), x(t)] \tag{1}$$

The selection of quality indicators is to ensure that the UAV moves optimally according to specific tasks. The process of solving the problem of optimizing the landing trajectory of a UAV, depending on the quality requirements, can choose a specific problem. For the landing process of a UAV, the requirement for accurate landing control is always set. In addition, the minimum energy criterion also needs to be considered. Therefore, we choose the Bolza problem to build the optimal landing trajectory for UAVs:

The Bolza problem has the form: $$J = g_0[x(t_0), x(t_f), t_0, t_f] + \int_{t_0}^{t} f_0(x,u,t)dt \tag{2}$$

Landing trajectory optimization problem for UAV-70V

Let's consider the case of UAV motion as a point mass in a vertical plane. Then the equation system describes the UAV movement in the form of:

$$\begin{cases} \dot{V} = g(n_x - \sin\theta) \\ \dot{\theta} = \dfrac{g}{V}(n_y - \cos\theta) \\ \dot{x} = V\cos\theta \\ \dot{y} = V\sin\theta \end{cases} \tag{3}$$

Where: V - Velocity; θ - Orbital inclination; x - Distance; y - Altitude; g - Gravitational acceleration ($g = 9.80665 (m/s^2)$); $X = [V, \theta, x, y]^T$ - UAV status vector.

n_x - Tangential overload, calculated according to the formula [4]:

$$n_x = \frac{T.\cos\alpha - X}{G} = \frac{T.\cos\alpha - \dfrac{\rho.V^2}{2}.S.C_x(\alpha, H)}{m.g} \tag{4}$$

Where: T - Traction of the motor; $C_x(\alpha, H)$ - UAV drag coefficient.

n_y - Velocity normal overload, and calculated according to the formula:

$$n_y = \frac{Y + T.\sin\alpha}{G} = \frac{\dfrac{\rho.V^2}{2}.S.C_y(\alpha, H) + T.\sin\alpha}{m.g} \tag{5}$$

In which: $C_y(\alpha, H)$ - Lifting force coefficient of UAV. The lifting force coefficient of UAVs can be approximate $C_y(\alpha, H) = C_y^{\alpha}.\alpha$. The angle of attack of the UAV is small, so it can be considered $\sin\alpha \approx \alpha$. Then the expression (5) is rewritten as:

$$n_y = \frac{\left(\dfrac{\rho.V^2}{2}.S.C_y^{\alpha} + T\right)}{m.g}.\alpha \tag{6}$$

Selecting the control signal $u = [n_x, n_y]^T$ The indicator function (quality indicator) selected according to the Bolza problem is in the form of:

$$J = 0{,}5\rho_1(\mathrm{x}(\mathrm{t}_f) - x_f)^2 + 0{,}5\rho_2(y(\mathrm{t}_f) - y_f)^2 + 0{,}5\rho_3(V(\mathrm{t}_f) - V_f)^2 + 0{,}5\rho_4(\theta(\mathrm{t}_f) - \theta_f)^2 + 0{,}5\int_{t_0}^{t} u^T k^{-2} u dt \tag{7}$$

Where: $\rho_1, \rho_2, \rho_3, \rho_4$ - Weights; $k^2 = diag(k_1^2, k_2^2)$ -Coefficient; t_0 and t_f – The beginning and end of the control process; V_f, θ_f, x_f, y_f - The desired state vector value of the UAV given at the end t_f; $V(t_f), \theta(t_f), x(t_f), y(t_f)$ - The status vector value of the UAV given at the end t_f.

According to Pontryagin's maximal principle, Hamilton's function corresponds to the form:

$$\mathrm{H} = P_V g(\mathrm{n}_x - \sin\theta) + P_\theta \frac{g}{V}(n_y - \cos\theta) + \mathrm{P}_x V\cos\theta + P_y V\sin\theta + \frac{1}{2}u^T k^{-2} u \tag{8}$$

In which: P_V, P_θ, P_x, P_y -The corresponding co-state variables according to the variable V, θ, x, y

At that time, the system of equations for the co-state variable has the form:

$$\begin{cases} \dot{\mathrm{P}}_V = -\dfrac{\partial \mathrm{H}}{\partial V} = P_\theta \dfrac{g}{V^2}(n_y - \cos\theta) - \mathrm{P}_x\cos\theta - P_y\sin\theta \\ \dot{P}_\theta = -\dfrac{\partial \mathrm{H}}{\partial \theta} = P_V g\cos\theta - P_\theta \dfrac{g}{V}\sin\theta + \mathrm{P}_x V\sin\theta - P_y V\cos\theta \\ \dot{\mathrm{P}}_x = -\dfrac{\partial \mathrm{H}}{\partial x} = 0 \\ \dot{\mathrm{P}}_y = -\dfrac{\partial \mathrm{H}}{\partial y} = 0 \end{cases} \tag{9}$$

If the control signal is not restricted, we find the optimal overload at each moment that makes the Hamilton function H reach its maximum. That is $\max \mathrm{H}(x^*, u, P^*, t) = \mathrm{H}(x^*, u^*, P^*, t)$. From the optimum condition, we get the optimum overload: $n_x = -P_V g k_1^2;\ n_y = -P_\theta \dfrac{g}{V} k_2^2$.

In case the control signal is restricted (overload stand n_y is restricted). The reason for only choosing to limit the standing overload n_y is because it has a large range of change and has a direct effect on the angle of attack of the UAV. The angle of attack of the UAV must always be ensured not to exceed the critical value because if the critical angle of attack value is exceeded, it will cause a slowdown and unsafe flight. Thus, limiting standing overload will also help limit the angle of attack of the UAV. We find the overload n_x, n_y at each point that causes the Hamilton function H to peak in the zone N_y (the vertical overload restriction zone n_y). That is $\max\limits_{n_y \in N_y} \mathrm{H}(x^*, u, P^*, t) = \mathrm{H}(x^*, u^*, P^*, t)$.

When the UAV lands, the UAV speeds up to the smallest value $V = V_{hc}$, according to the equation (6), which reaches n_y it's maximum when α it reaches its maximum.

Thus, if we limit the maximum value of the angle of attack, the value of the limited overload will be determined according to the formula:

$$n_{yhc} = \frac{\left(\dfrac{\rho . V_{hc}^2}{2} . S . C_y^\alpha + T\right) . \alpha_{\max}}{m.g} \tag{10}$$

In addition, according to the above assumption, the standing overload of the UAV must meet the conditions $n_y \geq -1$. Therefore, we must find the maximum value of the Function H for the variable n_x, n_y (in which n_y the condition must be satisfied $-1 \leq n_y \leq n_{yhc}$). According to the Hamilton function expression, it is a H 2nd-order function for variables n_y. Therefore, finding the jaw maximum H is not difficult. The necessary problem is to find the initial conditions $P_V(t_0)$, $P_\theta(t_0)$, $P_x(t_0)$, $P_y(t_0)$, t_f satisfaction of boundary conditions $V(t_f)=V_f$, $\theta(t_f)=\theta_f$, $x(t_f)=x_f$, $y(t_f)=y_f$, $H(X,P,t_f)=0$. This is the solution to the boundary problem, the solution of this problem will be difficult because of the connection with the calculation time, the choice of the initial approximate parameters, and the convergence of the method. Some studies have used the Newton-Raphson method, but when the control signals are limited, the Newton-Raphson method is very complex. Other studies have proposed a method of continuous parametric solution, which has shown outstanding superiority. Thus, the method of continuous solving by parameters will find the initial set of conditions that satisfy the boundary conditions.

The system of equations that fully describe the movement of the UAV will be:

$$\begin{cases} \dot{V} = g(n_x - \sin\theta); \\ \dot{\theta} = \dfrac{g}{V}(n_y - \cos\theta); \\ \dot{x} = V\cos\theta; \\ \dot{y} = V\sin\theta; \\ \dot{P}_V = P_\theta \dfrac{g}{V^2}(n_y - \cos\theta) - P_x\cos\theta - P_y\sin\theta; \\ \dot{P}_\theta = P_V g\cos\theta - P_\theta \dfrac{g}{V}\sin\theta + P_x V\sin\theta - P_y V\cos\theta; \\ \dot{P}_x = 0; \\ \dot{P}_y = 0; \end{cases} \tag{11}$$

2.2. Solving the problem of optimizing the landing trajectory for UAV-70V

Based on considering the methods of solving the boundary problem, we choose the method of continuous solving according to parameters to solve the problem of optimal landing trajectory of UAVs.

When using the method of continuous parametric solving to the UAV trajectory optimization problem, the case in the vertical plane or in space is essentially the same, except for the number of equations describing the movement of the UAV as well as the corresponding number of co-state equations. In addition, the number of control signals in these 2 cases is also different. So, using the method of continuous parametric solution, it is only necessary to consider the case in the vertical plane, and the case in the completely similar space.

The use of the method of continuous parametric solving to the optimal problem of the trajectory board of the UAV in the vertical plane will be carried out according to the following steps:

Step 1: Set any (approximate) initial value of the co-state variables (necessary so that they are not simultaneously equal to 0), the co-state variables that start to perform the problem at the initial time t_0 have the form:

$$P^i(t_0) = \begin{bmatrix} P_V(t_0) \\ P_\theta(t_0) \\ P_X(t_0) \\ P_Y(t_0) \\ t_f \end{bmatrix} = \begin{bmatrix} P_1(t_0) \\ P_2(t_0) \\ P_3(t_0) \\ P_4(t_0) \\ P_5(t_0) \end{bmatrix} = \begin{bmatrix} P_1(t_0) \\ P_2(t_0) \\ P_3(t_0) \\ \vdots \\ P_N(t_0) \end{bmatrix} \tag{12}$$

In which: $V_f = V_{mm}$ at t_f the time ; $\theta_f = \theta_{mm}$ at t_f the time ; $X_f = X_{mm}$ at t_f the time ; $Y_f = Y_{mm}$ at t_f the time.

With $V_{mm}, \theta_{mm}, X_{mm}, Y_{mm}$ - Velocity, orbital tilt angle, distance, desired altitude at the end t_f. This means that the desired velocity, coordinates, and angle of movement at the end is foreknowledge, we control the UAV to the right end of the trajectory;

i - Positive integer (number of repetitions).

N - The total number of co-state variables and variables t_f (the number of co-state variables is equal to the number of equations, describing the movement of the object).

Step 2: Solve the problem of controlling the movement of UAVs from t_0 to t_f.

Step 3: According to the trajectory, calculate the movement of the UAV, and receive a double error vector:

$$Z_0(P^i(t_0)) = \begin{bmatrix} V(t_f) - V_{mm} \\ \theta(t_f) - \theta_{mm} \\ X(t_f) - X_{mm} \\ Y(t_f) - Y_{mm} \\ \mathrm{H}(t_f) - 0 \end{bmatrix} = \begin{bmatrix} Z_{01}(P^i(t_0)) \\ Z_{02}(P^i(t_0)) \\ Z_{03}(P^i(t_0)) \\ Z_{04}(P^i(t_0)) \\ Z_{05}(P^i(t_0)) \end{bmatrix} \tag{13}$$

Step 4: Give the family number $\Delta P_j(t_0)$ of the function for the second state covariable j. It is possible to $\Delta P_j(t_0)$ get equal to $0,1$ words $P_j(t_0)$ with any sign ($\pm$) if $P_j(t_0) \neq 0$ or you can choose the homogeneous number of state covariables as $0,001$.

From there, it is calculated: $P_j(t_0) = P^i(t_0) + \Delta P_j(t_0)$

For example, if $j = 1$ you get:

$$P_1(t_0) = P^i(t_0) + \Delta P_1(t_0) = \begin{bmatrix} P_1(t_0) + \Delta P_1(t_0) \\ \vdots \\ \vdots \\ \vdots \\ P_N(t_0) \end{bmatrix}, \text{ còn } \Delta P_1(t_0) = \begin{bmatrix} \Delta P_1(t_0) \\ 0 \\ \vdots \\ \vdots \\ 0 \end{bmatrix}$$

Step 5: Solve the UAV motion control problem (according to the previous expression of optimal control) from t_0 to t_f.

Step 6: According to the trajectory of the UAV's motion calculation, it will receive a double error vector:

$$Z_j(P_j(t_0)) = \begin{bmatrix} V(t_f) - V_{mm} \\ \theta(t_f) - \theta_{mm} \\ X(t_f) - X_{mm} \\ Y(t_f) - Y_{mm} \\ \mathrm{H}(t_f) - 0 \end{bmatrix} = \begin{bmatrix} Z_{j1}(P_j(t_0)) \\ Z_{j2}(P_j(t_0)) \\ Z_{j3}(P_j(t_0)) \\ \vdots \\ Z_{j5}(P^i(t_0)) \end{bmatrix} \tag{14}$$

Step 7: Create the second column j of the matrix Z (Jacobi matrix), if $j = 1$, then:

$$Z = \begin{bmatrix} \frac{\partial Z_{11}}{\partial P_1} & \frac{\partial Z_{21}}{\partial P_2} & \frac{\partial Z_{31}}{\partial P_3} & \frac{\partial Z_{41}}{\partial P_4} & \frac{\partial Z_{51}}{\partial P_5} \\ \frac{\partial Z_{12}}{\partial P_1} & \frac{\partial Z_{22}}{\partial P_2} & \frac{\partial Z_{32}}{\partial P_3} & \frac{\partial Z_{42}}{\partial P_4} & \frac{\partial Z_{52}}{\partial P_5} \\ \frac{\partial Z_{13}}{\partial P_1} & \frac{\partial Z_{23}}{\partial P_2} & \frac{\partial Z_{33}}{\partial P_3} & \frac{\partial Z_{43}}{\partial P_4} & \frac{\partial Z_{53}}{\partial P_5} \\ \frac{\partial Z_{14}}{\partial P_1} & \frac{\partial Z_{24}}{\partial P_2} & \frac{\partial Z_{34}}{\partial P_3} & \frac{\partial Z_{44}}{\partial P_4} & \frac{\partial Z_{54}}{\partial P_5} \\ \frac{\partial Z_{15}}{\partial P_1} & \frac{\partial Z_{25}}{\partial P_2} & \frac{\partial Z_{35}}{\partial P_3} & \frac{\partial Z_{45}}{\partial P_4} & \frac{\partial Z_{55}}{\partial P_5} \end{bmatrix} \tag{15}$$

$$\Leftrightarrow Z = \begin{bmatrix} \frac{\partial Z_{11}}{\partial P_V} & \frac{\partial Z_{21}}{\partial P_\theta} & \frac{\partial Z_{31}}{\partial P_X} & \frac{\partial Z_{41}}{\partial P_Y} & \frac{\partial Z_{51}}{\partial P_{t_f}} \\ \frac{\partial Z_{12}}{\partial P_V} & \frac{\partial Z_{22}}{\partial P_\theta} & \frac{\partial Z_{32}}{\partial P_X} & \frac{\partial Z_{42}}{\partial P_Y} & \frac{\partial Z_{52}}{\partial P_{t_f}} \\ \frac{\partial Z_{13}}{\partial P_V} & \frac{\partial Z_{23}}{\partial P_\theta} & \frac{\partial Z_{33}}{\partial P_X} & \frac{\partial Z_{43}}{\partial P_Y} & \frac{\partial Z_{53}}{\partial P_{t_f}} \\ \frac{\partial Z_{14}}{\partial P_V} & \frac{\partial Z_{24}}{\partial P_\theta} & \frac{\partial Z_{34}}{\partial P_X} & \frac{\partial Z_{44}}{\partial P_Y} & \frac{\partial Z_{54}}{\partial P_{t_f}} \\ \frac{\partial Z_{15}}{\partial P_V} & \frac{\partial Z_{25}}{\partial P_\theta} & \frac{\partial Z_{35}}{\partial P_X} & \frac{\partial Z_{45}}{\partial P_Y} & \frac{\partial Z_{55}}{\partial P_{t_f}} \end{bmatrix} \tag{16}$$

In which: $\frac{\partial Z_{jk}}{\partial P_j} = \frac{Z_{jk}(P_j(t_0)) - Z_{0k}(P^i(t_0))}{\Delta P_j}$ $(1 \le k \le N)$

Step 8: If $j < N$ and the $j = j+1$ calculation is performed starting with step 4; if $j = N$ then the full Jacobi matrix (Z) is calculated and continues with step 9.

Step 9: The value of the new initial state covariate is written in the form:

$$P^{i+1}(t_0) = P^i(t_0) - \int_0^1 (Z)^{-1} . Z_0(P^i(t_0)) d\tau \tag{17}$$

If the matrix Z^{-1} does not exist, to calculate when the definition approaches zero, the matrix Z is often Z^{-1} replaced by its approximation. The matrix Z^{-1} can be replaced Z^{+} by an inverse pseudo-matrix. Inverse pseudomatrices can be found using the Greville method or the Moore-Penrose method (using the Pinv function in Matlab).

Start the problem with new initial conditions $P^{i+1}(t_0)$, calculate the movement of the UAV from t_0 to t_f and calculate the double error.

Step 10: If the condition is fulfilled $\left\| Z(P^{i+1}(t_0)) \right\| < \varepsilon_p$, the initial co-state variable result is obtained.

Where: $\left\| Z(P^{i+1}(t_0)) \right\|$ - Dual error vector modulus, defined:

$$\left\| Z(P^{i+1}(t_0)) \right\| = \sqrt{(V(t_f) - V_{mm})^2 + (\theta(t_f) - \theta_{mm})^2 + (X(t_f) - X_{mm})^2 + (Y(t_f) - Y_{mm})^2 + (\mathrm{H}(t_f))^2}$$

ε_p - The pre-selected constant, which characterizes the approximate prize desired to receive.

If this is not possible, then $P^i(t_0) = P^{i+1}(t_0)$, and continue solving the problem starting from step 2.

Thus: By solving the steps as presented above, the result is that we will find the state variable at the initial time (including: $P_V(t_0), P_\theta(t_0), P_X(t_0), P_Y(t_0), t_f$). From there, we can also determine the program trajectory (including: $V(t), \theta(t), x(t), y(t)$)) and control signals (n_x, n_y).

We use the simulation method using Matlab Simulink software to test and evaluate the research results. At that time, the conditions that need to be ensured for the UAV to land are as follows:

Altitude error at the time of landing $0m \le |\Delta y| \le 0,3m$; Distance error: $|\Delta x| \le 30m$;

The formula for determining the landing speed is derived from the UAV's gravitational equilibrium with the landing lift (the time just before landing when the ground jets are applied to the UAV).

$$Y = C_{yHC} \frac{\rho V_{hc}^2}{2} S = G \Rightarrow V_{hc} = \sqrt{\frac{2G}{C_{yHC} \rho S}} \tag{18}$$

Where: C_{yHC} - Lifting force coefficient at the time of landing; ρ - Air density at the ground; G - Gravity of the UAV; S - Effective wing area of the UAV.

Calculating with the UAV-70V model, we can determine the landing speed of the UAV as follows:

$$V_{hc} = \sqrt{\frac{2G}{C_{yHC} \rho S}} = \sqrt{\frac{2mg}{\alpha C_y^\alpha \rho S}} = \sqrt{\frac{2 \times 56,5 \times 9,81}{12 \times \frac{3,14}{180} \times 5,9123 \times 1,225 \times 1,05}} \Rightarrow V_{hc} = 26,3817 (m/s) \tag{19}$$

Vertical velocity on landing: $|V_{yhc}| \le 1 m/s$;

The angle of the UAV when landing $0 \le \vartheta \le 12^o$. This condition is to ensure that the UAV does not hit its head down and does not touch its tail when landing. According to the size parameters of the UAV (including body size, and claw size), for the UAV not to touch the tail when landing, the angle of the UAV must not be exceeded 12^o;

The vertical overload of the UAV during flight in general and landing in particular needs to be ensured within the range $-1 \le n_y \le 3,5$ (to ensure that the UAV is not destroyed by the structure), especially when landing, the vertical overload of the UAV must be approximately 1.

Suppose the initial state of the UAV when it enters the landing at point A (Figure 2). The UAV flies at the same speed $V(0) = 50\ m/s$, the initial trajectory tilt angle $\theta(0) = 0\ (rad)$, the position of the original UAV on landing is: $y(0) = 60\ m$; $x(0) = 0\ m$.

Where: l - Runway length; Δl - Distance from the end of the runway to the desired landing location $\Delta l = 40\ m$

The desired final state of the UAV at point B position:

$$y_f = 0,7\ m; x_f = 500\ m; \theta_f = 0\ radian; V_f = 31\ m/s$$

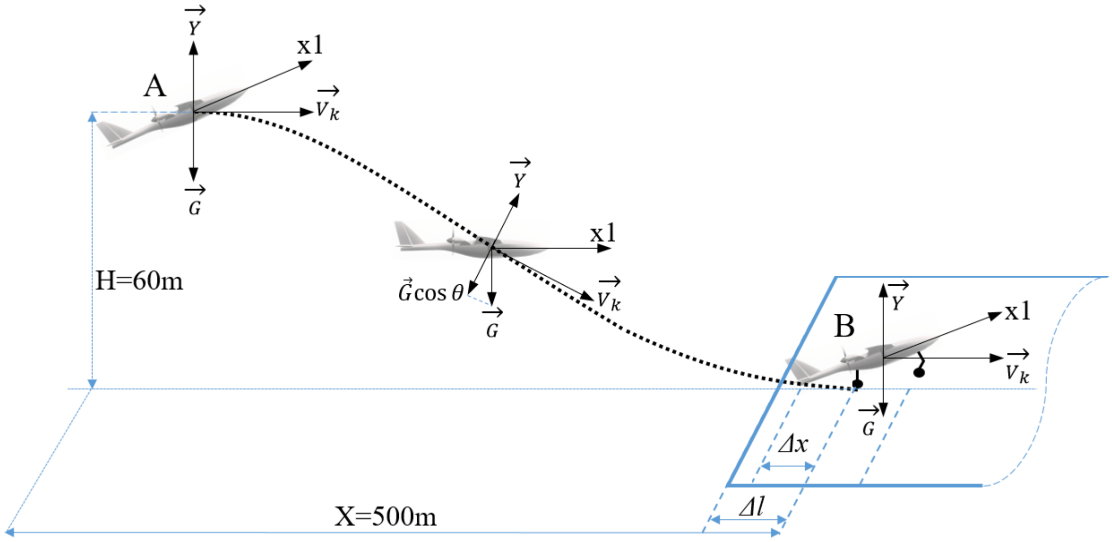

Figure 2. Diagram of forces acting on the UAV during landing

According to the calculation of UAV size, when the UAV lands, the rear gear of the UAV lands first if the angle of the UAV when landing is equal 12^o, the distance from the UAV's center of gravity to the lowest position of the rear gear is about $0,7m$. Therefore, when calculating, consider the UAV to land when the height of the UAV is equal to $0,7m$.

Cases where there is no restriction on standing overload

Considering the initial state of the UAV with: $V(0) = 50\ (m/s);\ \theta(0) = 0\ (rad);\ x(0) = 0\ (m);\ y(0) = 60\ (m)$. The desired end state of the UAV: $V_f = 31\ m/s;\ \theta_f = 0\ (rad);\ x_f = 500\ (m);\ y_f = 0,7\ (m)$.

Consider that: $k_1 = 0,1;\ k_2 = 0,1$. Using Matlab software gives the following results:

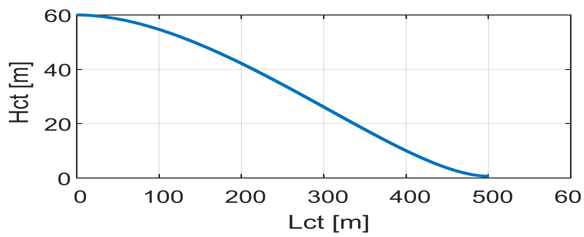

Figure 3. The trajectory of the UAV

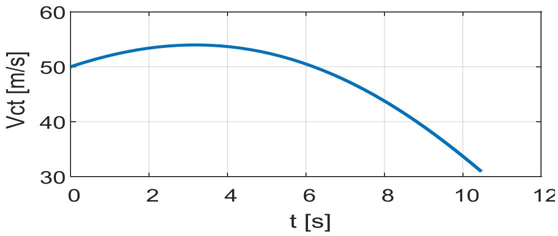

Figure 4. The dependence of the flight velocity of time

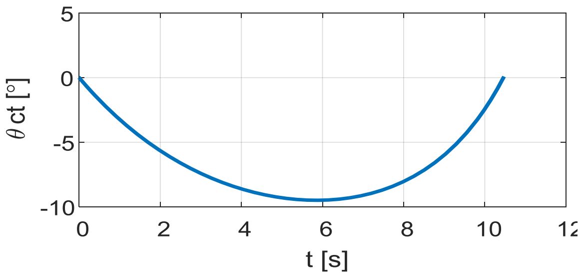

Figure 5. The dependence of the flight-path angle of time

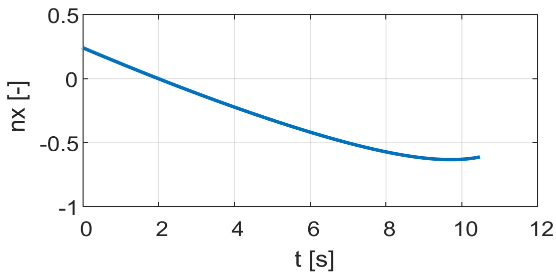

Figure 6. The dependence of the tangential load factor of time

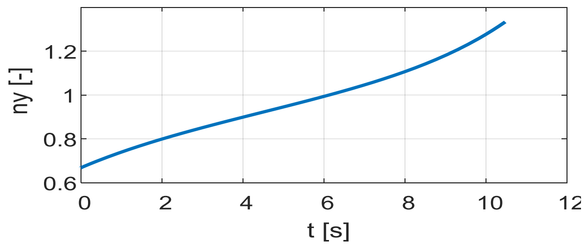

Figure 7. The dependence of the normal load factor of time

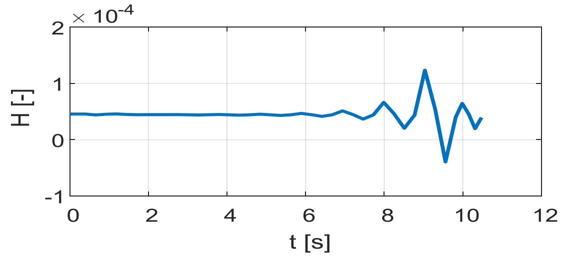

Figure 8. The dependence of the Pontryagin function values of time

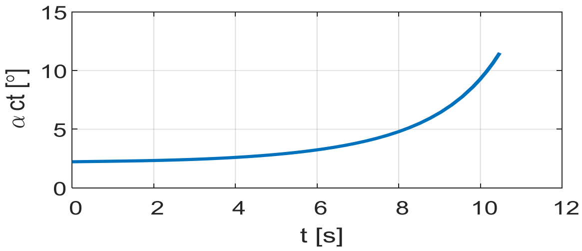

Figure 9. The dependence of attack angle of time

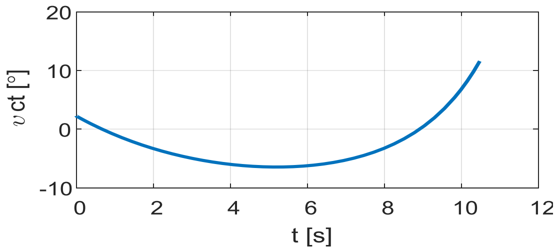

Figure 10. The dependence of pitch angle of time

Thus, with the desired set of initial and final states of the UAV (V, θ, x, y) *, the calculation program has found out the trajectory of the UAV's landing program as well as the corresponding overload* n_x, n_y *. However, in this case, considering the desired landing speed (*$V_f = 31 m/s$*), it is found that the attack angle and angle of the UAV exceed the permissible range* $0 \le \vartheta \le 10^o$*. Therefore, next, we will change the desired landing speed (*V_f*) to evaluate the effect of the desired landing speed on the UAV's state parameters when landing.*

Use Matlab software to write and run the program in each case of the desired velocity at different end times ($V_{f1} = 31\ m/s$; $V_{f2} = 35\ m/s; V_{f3} = 39\ m/s$), the results are as follows:

Figure 11 shows the trajectory of the UAV corresponding to the desired velocities at different end times (V_f). Figure 12 shows the velocity of the UAV. Figure 13 shows the change in the trajectory angle of the UAV over time corresponding to different conditions V_f . Figure 14, and Figure 15 show the change in velocity tangential overload and velocity normal overload over time. Figure 16 shows the change in the value of Hamilton's function. Figures 17 and 18 show the change in the angle of attack and the angle of the UAV.

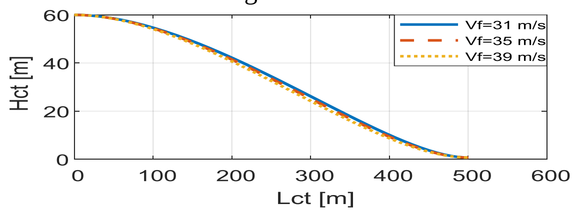

Figure 11. The trajectory of the UAV

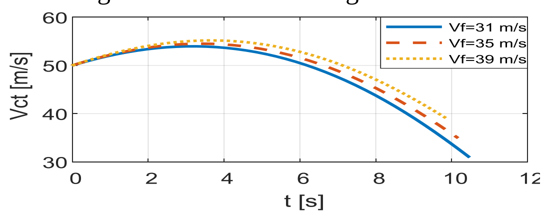

Figure 12. The dependence of the flight velocity of time

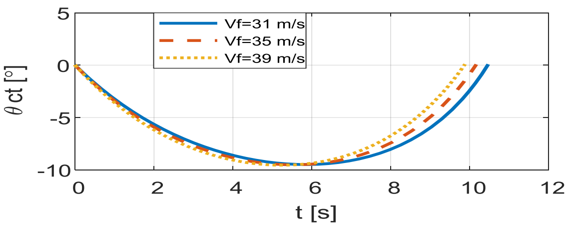

Figure 13. The dependence of the flight-path angle of time

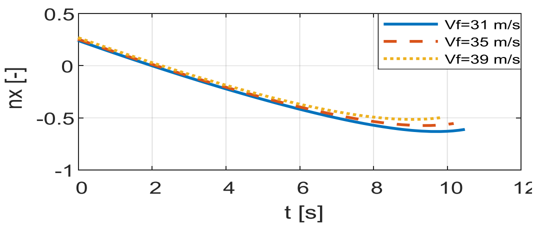

Figure 14. The dependence of the tangential load factor of time

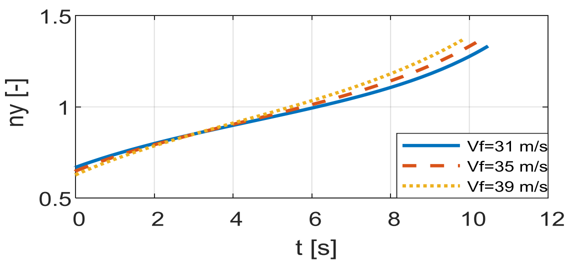

Figure 15. The dependence of the normal load factor of time

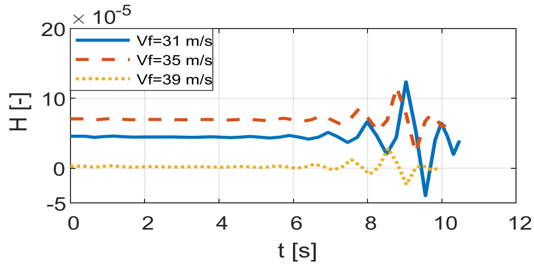

Figure 16. The dependence of the Pontryagin function values of time

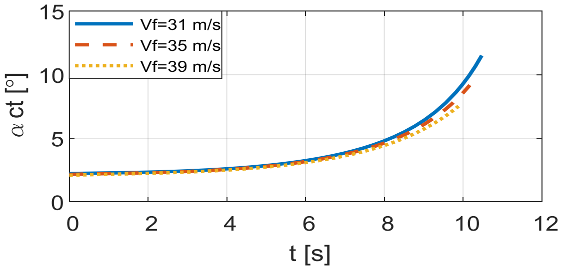

Figure 17. The dependence of attack angle of time

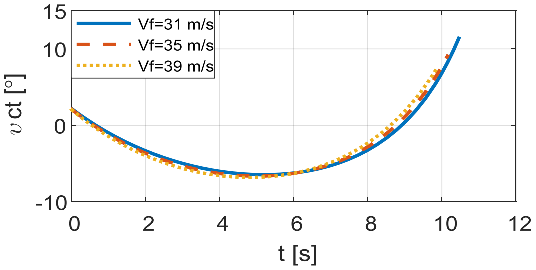

Figure 18. The dependence of pitch angle of time

Thus, the angle of attack and the angle of the UAV at the end depends on V_f . Through the survey, it was found that to ensure safe landing conditions, it is only allowed to reduce V_f to $V_f = 35\ (m/s)$ (because if the reduction is smaller, the angle of attack and the angle of the UAV exceed the permissible value). At such a speed V_f , it is quite large compared to the smallest landing speed V_{hc} . This leads to the UAV's rolling distance will be significantly large, and it is unlikely that the UAV will land on a short runway. One solution offered is to limit standing overload.

Cases of restriction of standing overload

The concept of standing overload restriction here is to maintain the standing overload not exceeding the permissible value. From the formula for determining the velocity normal overload:

$$n_y = \frac{Y + T.\sin\alpha}{G} \Rightarrow n_y.G = Y + T.\sin\alpha \tag{20}$$

Where:

$$Y = C_y^{\alpha}.\alpha.\frac{\rho.V^2.S}{2} \tag{21}$$

Y - UAV lift; T - Motor traction; α - The angle of attack of the UAV.

Transforming the formula (20), we are:

$$n_y = \frac{C_y^{\alpha}.\alpha.\frac{\rho.V^2}{2}.S + T.\sin\alpha}{G} \Rightarrow n_{yhc} = \frac{C_y^{\alpha}.\alpha_{\max}.\frac{\rho.V_f^2}{2}.S + T.\sin\alpha_{\max}}{G} \tag{22}$$

For each velocity V_f , we will determine the normal overload of the limited velocity (n_{yhc}) to ensure that the angle of attack does not exceed the permissible value. However, when it V_f decreases, it n_{yhc} also decreases. And when n_{yhc} it decreases beyond a certain value, the program will not find the optimal solution.

In case of restriction of standing overload, the results of the program are as follows:

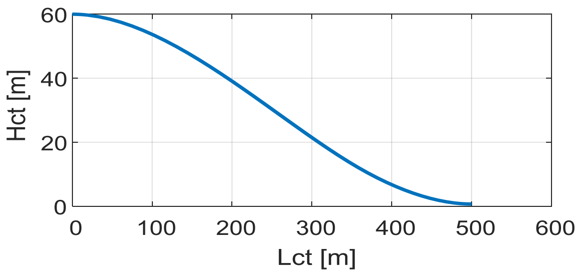

Figure 19. The trajectory of the UAV

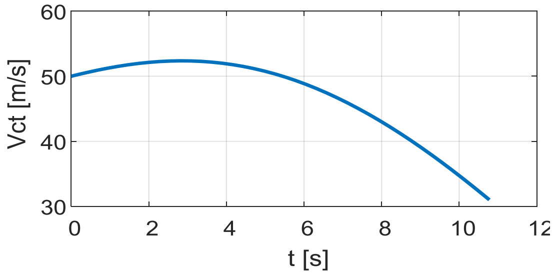

Figure 20. The dependence of the flight velocity of time

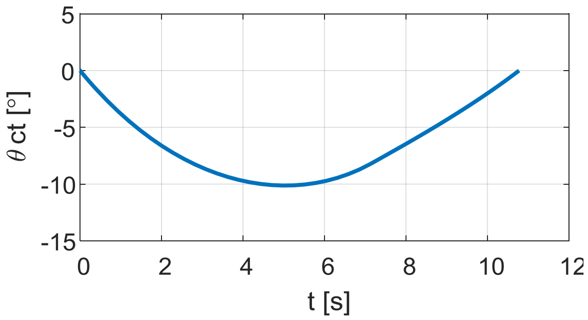

Figure 21. The dependence of the flight-path angle of time

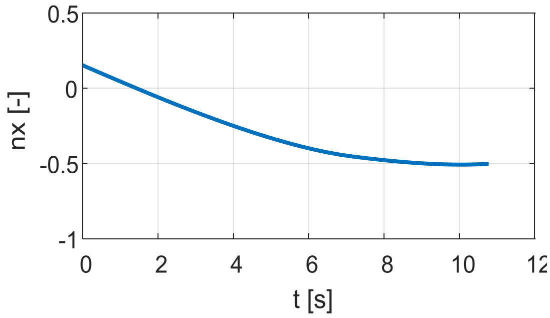

Figure 22. The dependence of the tangential load factor of time

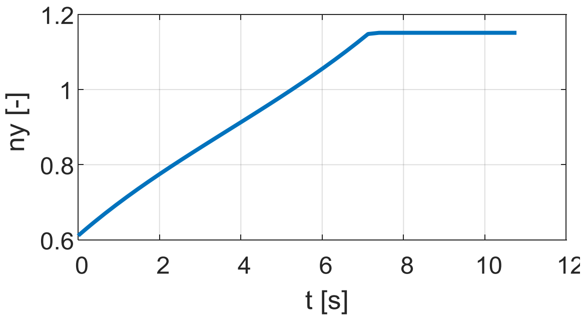

Figure 23. The dependence of the normal load factor of time

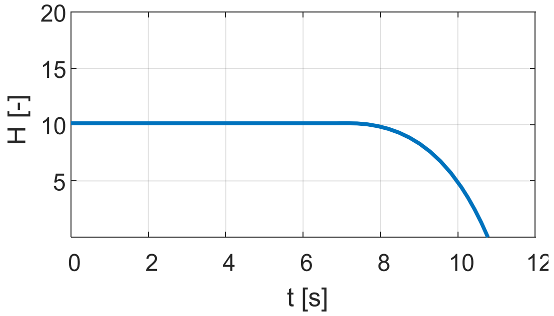

Figure 24. The dependence of the Pontryagin function values of time

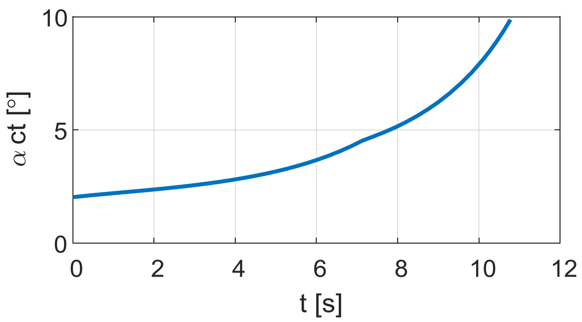

Figure 25. The dependence of attack angle of time

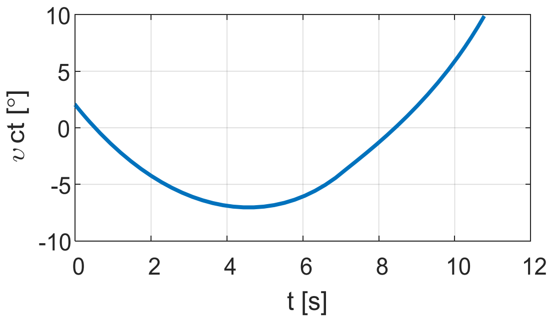

Figure 26. The dependence of pitch angle of time

When limiting standing overload, allows the UAV to land at a significantly small speed $V_f = 31 m/s$ while still ensuring the angle of attack and angle of attack of the UAV within the permissible limits. This will significantly reduce the rolling distance of the UAV in case it is necessary to control the UAV to land on a short runway.

III. DISCUSSION AND CONCLUSION

Thus, with the desired set of initial and final states of the UAV, the calculation program has determined the landing trajectory as well as the corresponding overload. However, when considering the desired landing speed, we find that the UAV's angle of attack and tilt angle exceed the permissible range. To solve this problem, it is necessary to change the desired landing speed to assess its effect on the UAV's state parameters during landing. Through the survey, we found that to ensure safe landing conditions, the landing speed can only be reduced to a certain level, because if it is reduced too much, the angle of attack and angle of attack of the UAV will exceed the permissible limit. With such a landing speed, although the angle of attack and angle of incidence are reduced, it is still quite large compared to the smallest landing speed that can be achieved. This leads to a significant increase in the UAV's rolling distance, making it difficult to land on short runways. An effective solution is to apply standing overload restrictions. When the standing overload restriction is applied, the UAV can land at a significantly smaller speed while maintaining the angle of attack and angle of rotation within the permissible limits. This significantly reduces the UAV's rolling distance, increases the ability to land safely on short runways, and improves the overall performance of the landing process.

REFERENCES

[1]. Yong Can Cao, HaiYang Chao, Yang Quan Chen (2010), Autopilots for Small Unmanned Aerial Vehicles A Survey, International Journal of Control Automation and Systems– № 8 – C.36-44.

[2]. J.A. Marin, Radtke, R., Innis, D., Barr, D.R., Schultz, A.C (1999), Using a genetic algorithm to develop rules to guide unmanned aerial vehicles, Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, vol. 1, pp. 1055–1060.

[3]. C.J. Schumacher, Kumar, R (2000), Adaptive control of UAVs in close-coupled formation flight, Proceedings of the American Control Conference, vol. 2, pp. 849–853.

[4]. Miguel A, Olivares-Mendez (2010), Fuzzy Controller for UAV-Landing Task Using 3D-Position Estimation, International Conference on Fuzzy Systems.