Enhanced Non-Linear UAV Guidance with Look-Ahead Vectors Utilizing Arc-Length Parameterized Paths

S. Agha[1], M. Abozaid[2] and G. Mabrouk Abdel–Hamid[1]

[1] Mathematics Department, Military Technical College, Cairo, Egypt.
[2] Missile Guidance Department, Military Technical College, Cairo, Egypt.

E-mail: salahagha@mtc.edu.eg

Abstract. Unmanned aerial vehicles (UAVs) have a wide range of applications in both civilian and military sectors. The UAV guidance system is important for achieving the proposed mission. Traditional Unmanned Aerial Vehicle (UAV) guidance laws often rely on look-ahead points, limiting their ability to track complex 3D paths accurately. This paper proposes a novel look-ahead vector guidance law that depends on differential geometry to enhance path-tracking performance, by arc length representation of the curves. the look-ahead vector encodes both direction and curvature information, enabling smoother and more precise manoeuvres, improving tracking accuracy in tight turns, and simplifying control law calculations. There are two design parameters of the proposed guidance law. Numerical simulations are performed on the 3_D kinematic model of a UAV with two input states by linking the output of the proposed guidance law and the angles of the model. The proposed algorithm was evaluated using MATLAB Simulink simulation for a UAV model. The results demonstrate the Nonlinear Differential Geometry Path-following Guidance Law (NDGPFG) and design parameters effect of the proposed guidance law. Simulation results presented the performance of the proposed look-ahead vector guidance law, paving the way for improved UAV path tracking in various applications.

1. Introduction

UAVs have many applications that are used for civilian and military applications. Civilian applications are classified into aerial photography, videography, mapping, surveying, delivery, and logistics. In addition to civilian applications, UAVs play a crucial role in military applications classified as target acquisition, precision strikes, force protection, training, simulation, and anti-drone operations [1, 2]. The integrated system of UAVs consists of a path planning system, navigation system, guidance system, control system, and UAV model, as shown in Figure 1.

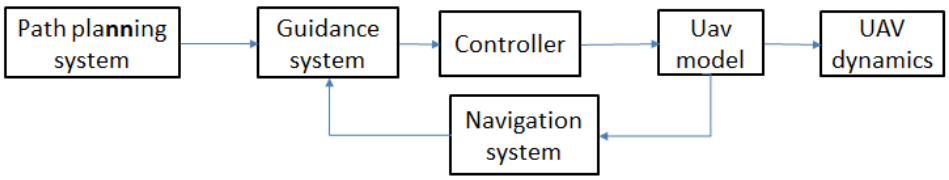


Figure 1. UAV's Integrated System

The path planner creates the desired path in terms of the vehicle's capabilities, mission requirements, and operational efficiency [3-5]. The guidance law is used to generate the command signal to the controller. The selection of guidance laws depends on the mission's constraints and the UAV's requirements.

Many linear and non-linear guidance laws of UAVs for path-path following have been developed. Linear guidance laws are developed to generate commands for linear control techniques such as proportional navigation (PN) guidance laws, line-of-sight (LOS) guidance laws, proportional navigation with line-of-sight (PNLOS) guidance laws, and optimal linear quadratic (LQ) control.

PN guidance law has many advantages such as continuous speed, robustness, and effectiveness of targets. However, the PN guidance law is used in many applications and has disadvantages such as not being able to achieve the movement target, vulnerability to non-cooperative targets, resistance to initial conditions, and limited flexibility [6, 7]. The LOS guidance law applies to many UAV guidance systems that are simple, real-time tracking, minimal target information, and are almost effective. However, the LOS guidance law has many advantages such as limited resistance to target manipulation, sensitivities to sensor limitations, vulnerability to sensor noise and delay, and a lack of autonomy path planning [8]. PNLOS' guidance law has many advantages, such as simplicity, robustness, and effectiveness against target-orientated operations. Although this type of guidance law has been used in many applications, it has many disadvantages, such as dependence on the line of sight. Although this type of guidance law has been used in many applications, it has many disadvantages, such as dependence on the line of sight, limited scope of engagement, and lack of target prediction [9, 10].

Linear guidance laws are limited, so nonlinear guidance laws developed to generate commands for nonlinear control techniques are needed in terms of target operation, nonlinear system dynamics, optimal control considerations, and robustness to uncertainties. The vector field guide laws apply to straight lines and the circular guide laws mean that they do not apply to space curves because the vector fields are designed for planar curves. It was developed for a specific type of curve, such as a circle or a straight line [11, 12]. It needs curves from the switching algorithm, which complicates the guidance system.

Many guidance systems use a virtual moving point in the desired path that is ahead of the vehicle. The non-linear path following guidance (NPFG) law uses the principle of the virtual target point and a look head point to guide UAVs to the desired path [13, 14]. The NPFG law has many advantages such as robustness against external disturbances and smooth incidence on the desired path. However, it has many disadvantages, as the initial position must be less than the look head distance, has an overshoot at the initial phase, needs an itching algorithm that makes the system more complex, and is not applicable for 3-D path following.

Mathematical analysis plays a crucial role in characterizing reference paths, as explored in the field of differential geometry. This mathematical discipline focuses on the study of surfaces and curves, providing insights into the properties of these geometric entities. By leveraging mathematical analysis to examine the reference path at discrete points along its trajectory, we can develop non-linear guidance laws. These laws are instrumental in addressing limitations observed in alternative guidance methodologies, thereby enhancing overall guidance system performance.

Nonlinear Differential Geometry Guidance (NDGPFG) law used the Look-Ahead vector to design the Look-Ahead angle instead of using the point of the desired path. It eliminates other disadvantages of other guidance laws, allows accurate tracking of all the desired paths, and allows sets of initial conditions to contain large deviations and simple vector formulas; therefore, it is easy to implement and does not require switching algorithms that reduce the effort of the controller [15-17].

The proposed algorithm was evaluated using MATLAB Simulink simulation for the kinematic UAV model extended to 3D Dimensional. The model input is the rate of change of the angle of the pitch and yaw angle, but the guidance law command is a normal acceleration, so normal acceleration is modified to the rate of change of the angle. This paper has been divided into Differential Geometry Representation The Differential Geometry section which contains mathematical representations of curves, the UAV kinematic model contains motion equations as shown in section 3, the Guide Law section which contains the Design of the Differential Geometry Guide Law and the Numerical Simulation and Results Analysis section which contains simulation results for different types of path ending analysis for the obtained results.

2. Differential Geometry Representation

Differential geometry is a mathematical field that studies surfaces and curves [18, 19]. Considering curves, they are represented as a function in parametric form as shown in (1).

$$f(t) = [x(t)\ y(t)\ z(t)] \tag{1}$$

Where $x(t), y(t), z(t)$ are the function of the curve at each coordinate.

The parameter function can be represented as any parameter function, and if it is a smooth and simple one-to-one function. For example, arc length functions meet these conditions, and curves can be represented as arc length functions, and as shown in Figure 2 arc length functions can be represented as shown in equation (2).

$$s = \int_a^b (x`(t)^2 + y`(t)^2 + z`(t)^2)^{\frac{1}{2}} \quad dt \tag{2}$$

From equation (2), the arc length can be represented as shown in (3).

$$r(s) = [x(s), y(s), z(s)] \tag{3}$$

The curve can be described by moving frames. The moving frames consist of three-unit vectors, including the: Tangent, Normal, and binormal vectors, based on the velocity and acceleration of the curve which are described by arc length representation as shown in (4) and (5) and the curve can be described by any parameter representation as shown in (6).

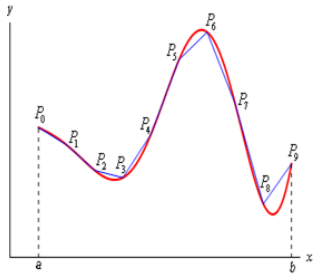


Figure 2. Arc length representation

2.1. Curve Arc Length Representation

Curve representation can be calculated by using arc_length parameter as shown in (4).

$$\mathrm{T} = \frac{\mathrm{dr(s)}}{\mathrm{ds}} \qquad \mathrm{k} = \left|\frac{\mathrm{dT(s)}}{\mathrm{ds}}\right| \ , \ \mathrm{N} = \frac{1}{\mathrm{k}} \times \frac{\mathrm{dT}}{\mathrm{ds}} \ , \ \mathrm{B} = \mathrm{T} \times \mathrm{N} \tag{4}$$

Where: T, N and B Are Tangent, Normal, and binormal vectors respectively. k is the curvature of the curve.

Concerning representation of the curve, the Fernet–Serret frame can be represented as shown in equation (5).

$$\begin{bmatrix} \frac{\mathrm{dT}}{\mathrm{ds}} \\ \frac{\mathrm{dN}}{\mathrm{ds}} \\ \frac{\mathrm{dB}}{\mathrm{ds}} \end{bmatrix} = \begin{bmatrix} 0 & \mathrm{k} & 0 \\ -\mathrm{k} & 0 & \tau \\ 0 & -\tau & 0 \end{bmatrix} \begin{bmatrix} \mathrm{T} \\ \mathrm{N} \\ \mathrm{B} \end{bmatrix} \tag{5}$$

2.2. Curve Time Representation

Curve parameters can be calculated by using a parameter function of any parameter as shown in (6).

$$T = \frac{\frac{dr(t)}{dt}}{\left|\frac{dr(t)}{dt}\right|}, \quad k = \frac{|r`(t)\times r``(t)|}{|r`(t)|^3}, \quad N = \frac{\frac{dT(t)}{dt}}{\left|\frac{dT(t)}{dt}\right|}, \quad B = T \times N, \quad \tau = \frac{r`(t).(r``(t)\times r```(t))}{|r`(t)\times r``(t)|^2} \qquad (6)$$

Where: T, N, and B Are Tangent, Normal, and binormal vectors respectively and k, τ are the curvature and torsion of the curve.

2.3. Path following geometry

Using the curve parameters defined by differential geometry, the following path geometry can be described as shown in Figure 3.

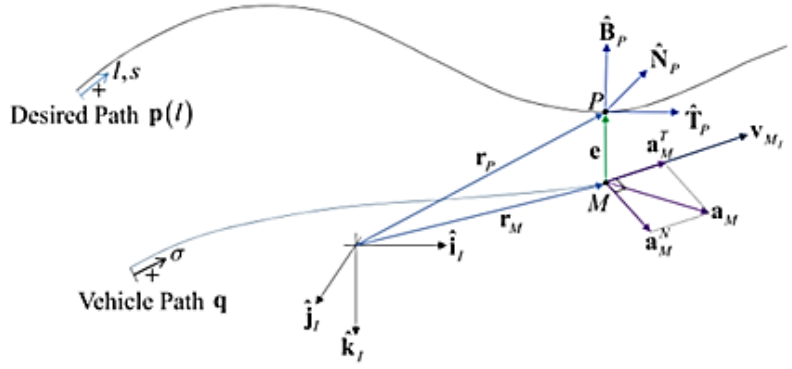


Figure 3. Geometry of 3-D path following

In Figure 3, the variables s and σ are the arc length of the desired path and the path of the vehicle, P is the closest projection point to the desired path, M is the vehicle, and I is the inertia.

Path-following problems can be solved by designing a guidance law command $a^{N}_{m_{cmd}}$ to satisfy the following conditions for a vehicle as $\sigma \to \infty$:

$$r_m(\sigma) \to r_p(s(\sigma)) \qquad (7)$$

$$\hat{T}_m(\sigma) \to \hat{T}_p(s(\sigma)) \qquad (8)$$

$$\frac{d\hat{T}_m(\sigma)}{d\sigma} \to \frac{d\hat{T}_p(s(\sigma))}{ds} \qquad (9)$$

Where $r_m(\sigma)$is UAV position vector of, $r_p s(\sigma)$ is reference point position vector
If conditions (7), (8), and (9) are satisfied with apposition convergence, velocity direction convergence, and curvature convergence, then it is said that the vehicle is on the track, aligned, and the curvature vector of the vehicle will be the same as that of the desired path.

3. UAV kinematic model

UAV's kinematic model is used to verify compliance with guidance law. In the case of constant speed, the UAV kinetic model is designed as shown in (10),(11),(12) [20].

$$\dot{x} = v_a\, cos(\theta)\, cos(\emptyset) + v_w\, cos(\theta_w)\, co\, s(\emptyset_w) = v_g\, cos(\theta_g)\, cos(\emptyset_g), \qquad (10)$$

$$\dot{y} = v_a\, cos(\theta)\, sin(\emptyset) + v_w\, cos(\theta_w)\, sin(\emptyset_w) = v_g\, cos(\theta_g)\, sin(\emptyset_g), \qquad (11)$$

$$\dot{z} = v_a \sin(\theta) + v_w \sin(\theta_w) = v_g \sin(\theta_g) \tag{12}$$

$$u = \dot{\theta}_g, \dot{\emptyset}_g = \dot{\theta}, \dot{\emptyset}$$

Where v_a m/s is UAV airspeed, θ is the heading angle of UVA, $\emptyset$ is the pitch angle of UAV, $\dot{\theta}, \dot{\emptyset}$ are the rate change of the heading and pitch angles which are constrained by $-\omega_{max} < \dot{\theta}, \dot{\emptyset} < \omega_{max}$, v_w is the wind speed and its direction is defined by $\theta_w, \emptyset_w$ and u is the control. The equation of ground speed is calculated from the equation $v_g = (\dot{x}^2 + \dot{y}^2 + \dot{z}^2)^{\frac{1}{2}}$ and $\dot{\theta}, \dot{\emptyset}$ which are computed from the velocity vector component $(\dot{x}, \dot{y}, \dot{z})$. In this paper, we consider $v_a = 15$ m/s for all simulations in which we take into account a minimum rotation radius of 45 m and take into account $v_w = 0$ m/s for simplicity.

4. Diffrential gemetry guidance Law

The design of the guide law is considered by geometry as shown in Figure 4. The nonlinear differential geometric path-following guidance law is proposed as shown in (13)

$$\boldsymbol{a}^N_{m_{cmd}} = k\left(\boldsymbol{v}_{m_I} \times \hat{\boldsymbol{L}}\right) \times \boldsymbol{v}_{m_I} \tag{13}$$

Where $a^N_{m_{cmd}}$ is the guidance command, which is the normal component of the accelerated vector, which is normal for the UAV speed vector, $\boldsymbol{v}_{m_I}$ Is the inertial velocity vector of UAV, k is the guidance gain (k>0), and $\hat{L}$ is the look-ahead vector. As shown in Figure 3 the look-ahead vector is a unit vector that rotates with angle θ_L (look-ahead angle) from the direction of d in the direction of $\hat{T}_p$ (unit tangential vector of reference path at the virtual point).

By simplification of equation (13) the guidance command can be written as shown in (14).

$$\mathbf{a}^{\mathbf{N}}_{\mathbf{m_{cmd}}} = \mathbf{k}(\|\mathbf{v_{m_I}}\|^2 \mathbf{rej}(\hat{\mathbf{L}}, \mathbf{v_{m_I}})) \tag{14}$$

Where

$$\mathbf{rej}(\hat{\mathbf{L}}, \mathbf{v_{m_I}})) = \hat{\mathbf{L}} - \frac{\hat{\mathbf{L}}.\mathbf{v_{m_I}}}{\|\mathbf{v_{m_I}}\|} \frac{\mathbf{v_{m_I}}}{\|\mathbf{v_{m_I}}\|}$$

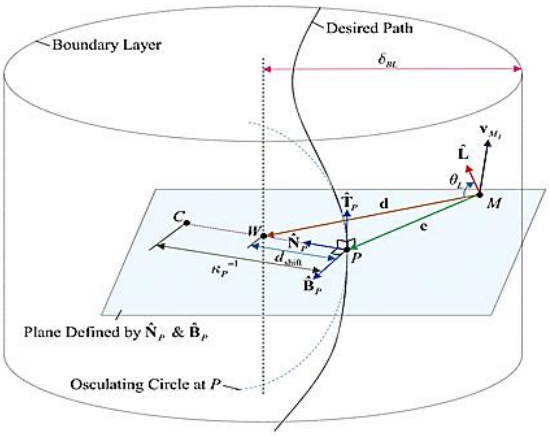


Figure 4. Guidance law geometry

As shown in Figure 4 the look–ahead vector $\hat{\boldsymbol{L}}$ is defined by θ_L andd_{shift}, where $\hat{\boldsymbol{L}}$ can be constructed as shown in (15)

$$\hat{L} = \begin{cases} \cos(\theta_L)\,\hat{d} + \sin(\theta_L)\,\hat{T}_p & \text{if} \|d\| \neq 0 \\ \hat{L} = \hat{T}_p \quad \text{if} \quad \|d\| = 0 \end{cases} \tag{15}$$

Where θ_L is the look-ahead angle and $\hat{d}$ is a unit vector of shifted error vector m . θ_L can be designed as a function of $\|d\|$ (which is designed later), and $\hat{d}$ can be designed by defining r_w where as shown in Figure 4, C is the centre of curvature at P, W is a point defined on the line between **C** and **P**. Therefore, r_w can be represented as shown in (16)

$$\mathbf{r_w} = \mathbf{r_p} + d_{shift}\text{sign}(kp)\hat{\mathbf{N}}_\mathbf{p} \tag{16}$$

Where d_{shift} is a radially shifted distance, $\mathbf{r_p}$ is a reference point vector, kp is a curve curvature at reference point and $\hat{\mathbf{N}}_\mathbf{p}$ is unit normal vector of the reference curve.
As shown also in Figure 4 the shifted error vector can be defined as

$$e = r_w - r_m$$

Then the $\hat{d}$ (unit vector) can be represented as shown in (17).

$$\boldsymbol{d} = \boldsymbol{r_w} - \boldsymbol{r_m} = \boldsymbol{e} + \boldsymbol{d_{shift}} sign(kp)\hat{\mathbf{N}}_\mathbf{p} \tag{17}$$

Where $\boldsymbol{d_{shift}}$ is radially shifted distance measured in meter, $\boldsymbol{e}$ is tracking error vector measured in meter and $\boldsymbol{d}$ is shifted error vector.

$$\hat{\boldsymbol{d}} = \frac{\boldsymbol{d}}{\|\boldsymbol{d}\|}$$

The look-ahead angle θ_L can be calculated as a function of $\|d\|$ and δ_{BL} (constant the radius of the boundary layer) where there are two functions as shown in (18), (19)

$$\boldsymbol{d} = \boldsymbol{r_w} - \boldsymbol{r_m} = \boldsymbol{e} + \boldsymbol{d_{shift}} sign(kp)\hat{\mathbf{N}}_\mathbf{p} \tag{17}$$

$$\theta_L(\|d\|) = \frac{\pi}{2}\left(1 - (1-\epsilon)sat\left(\frac{\|d\|}{\delta_{BL}}\right)\right)^{\frac{1}{2}} \tag{18}$$

$$\theta_L(\|d\|) = \cos^{-1}((1-\epsilon)sat(\frac{\|d\|}{\delta_{BL}})) \tag{19}$$

Where: $0 \le \epsilon < 1$ and $\text{sat}(x) = x$ if $|x| \le 1$ & $\text{sat}(x) = \text{sign}(x)$ if $|x| > 1$.
The function of $\theta_L(\|d\|)$ at the exact path following where $\|d\| = \|d_{shift}\|$ is represented as shown in 20).

$$\theta_L(d_{shift}) = \cos^{-1}(\frac{|k_p|}{k}) \tag{20}$$

By substitution of (20) into (18),(19) the function of $\boldsymbol{d_{shift}}$ as shown in (21), (22) as follows

$$\boldsymbol{d_{shift}} = \left[1 - \left(\frac{2}{\pi}\cos^{-1}\left(\frac{|k_p|}{k}\right)\right)^2\right]\frac{\delta_{BL}}{1-\epsilon} \tag{21}$$

$$\boldsymbol{d_{shift}} = \frac{|k_p|}{k}\frac{\delta_{BL}}{1-\epsilon} \tag{22}$$

Where k is gain of guidance law and δ_{BL} is a boundary-layer thickness measured in meter.
After the evaluation of d_{shift} into (21), (22), a unit vector $\hat{d}$ can be evaluated from (17) and by adding the evaluation of (18), (19), the look-ahead vector $\hat{L}$ is defined as shown in (15) and the guidance command can be calculated as shown in (14).

5. Numerical simulation and results analysis

A numerical simulation was performed to demonstrate the performance of the NLDGPFG. The selection of design parameters in the proposed guidance law (k, δ_{BL}) will affect the characteristics of the initial approach stage, such as overshooting, accelerating, peak, and stabilization. The design parameter k is determined first from the point of view of the maximum value of the guidance command and overshot tendency, and then from the point of view of the characteristics of the initial approach to the desired path, the design parameter δ_{BL} is adjusted. Overshoot amount as related to k and δ_{BL} which the effect of these parameters is demonstrated in the following simulations which are performed for each desired path as shown in figure 5 for a straight line and a figure 6 for a circular path and a figure 7 for a helix path and a figure 8 for a cubic spline curve. The values of k and δ_{BL} parameters are choosen as presented in[15]

5.1. Straight Line Path Following

The desired path for the straight line is represented as a function of arc length as shown in equation (23) as follows:

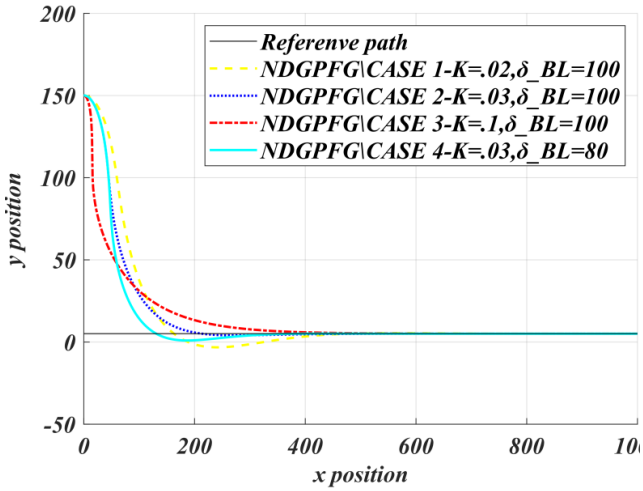
$$p(s) = \begin{bmatrix} s \\ 5 \\ 0 \end{bmatrix} \text{m} \tag{23}$$

The simulation parameters for a straight-line simulation are summarized in Table 1.

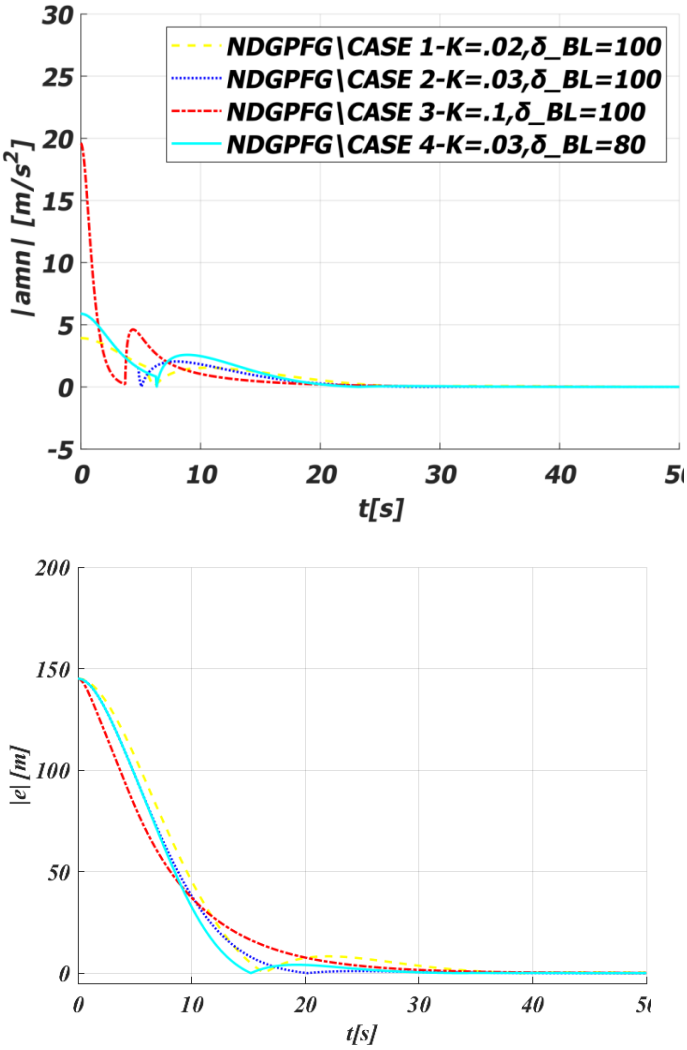
Table 1. Guidance law design simulation parameters

Initial condition	$r_m(t_0) = \begin{bmatrix} 0 \\ 150 \\ 0 \end{bmatrix}$.m	$\theta(t_0) = 0$ rad	$\emptyset(t_0) = 0$rad
Design parameter	NLDGPFG/Case 1	$k = 0.02\ m^{-1}$	$\delta_{BL} = 100$ m
	NLDGPFG/Case 2	$k = 0.03\ m^{-1}$	$\delta_{BL} = 100$ m
	NLDGPFG/Case 3	$k = 0.1\ m^{-1}$	$\delta_{BL} = 100$ m
	NLDGPFG/Case 4	$k = 0.03\ m^{-1}$	$\delta_{BL} = 80$ m

The simulation of a straight line path following can be demonstrated as shown in figure 5.



a) 2-D trajectory (y-axis)



b)Guidance command/error history

Figure 5. Straight-line path-following

The value of k is increased from the value of NLDGPFG/Case 3 to NLDGPFG/Case 2, and can also be increased from NLDGPFG/Case 1 to demonstrate the effect of the design parameter k. Similarly, NLDGPFG/case 4 δ_{BL}value is reduced from NLDGPFG/case 3 to demonstrate its effect. as shown in figures (5) the value of gain has affected the value of guidance command and increasing the value of gain in NLDGPFG/Case 2 than in NLDGPFG/Case 1 reduce overshoot value and the settling time. In NLDGPFG/Case 3, the overshoot value tends to zero but there is an increase in the settling time but the effect of increasing the gain NLDGPFG/Case 3 Doesn't have an effective effect on the overshoot value although there are increasing value of guidance command and settling time. So, we must adjust the gain value for the best performance and minimum guidance command. NLDGPFG/Case 4 the overshot is increased, but the rise time and the error are decreased with a small increase in guidance concerning reducing the value of the thickness of the boundary-layer δ_{BL}.

5.2. Circular path following

The desired path of the circular path is represented as a function of arc length as shown in equation (24) as follows

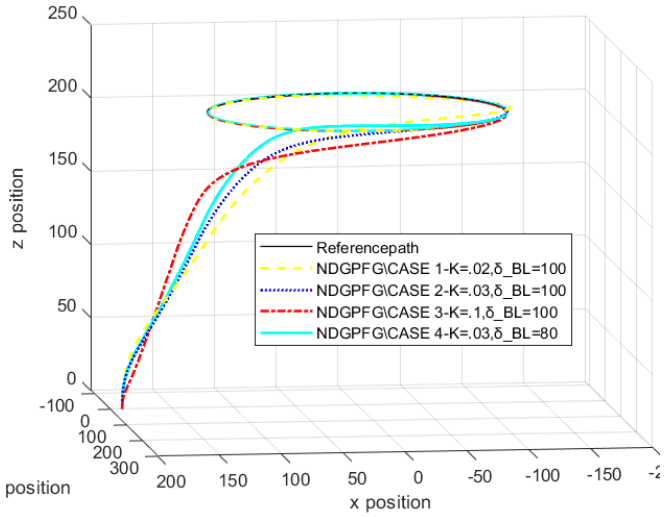
$$p(s) = \begin{bmatrix} 120\cos(\frac{s}{120}) \\ 120\sin(\frac{s}{120}) \\ 200 \end{bmatrix} m \quad (24)$$

The simulation parameters of circular path simulation are summarized in Table 2.

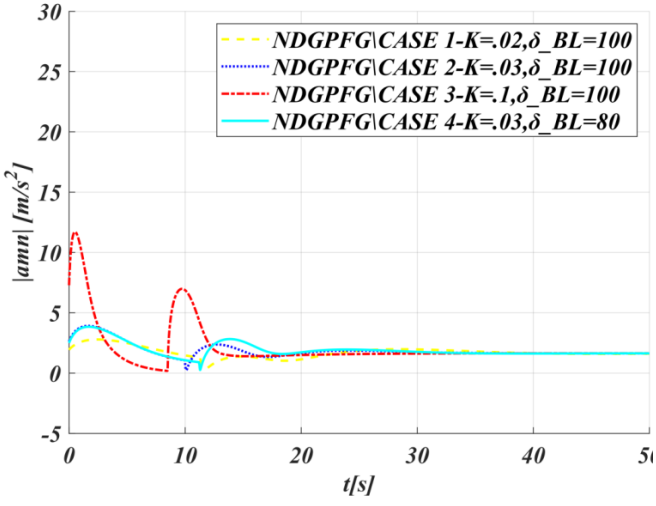
Table 2. Guidance law design simulation parameters

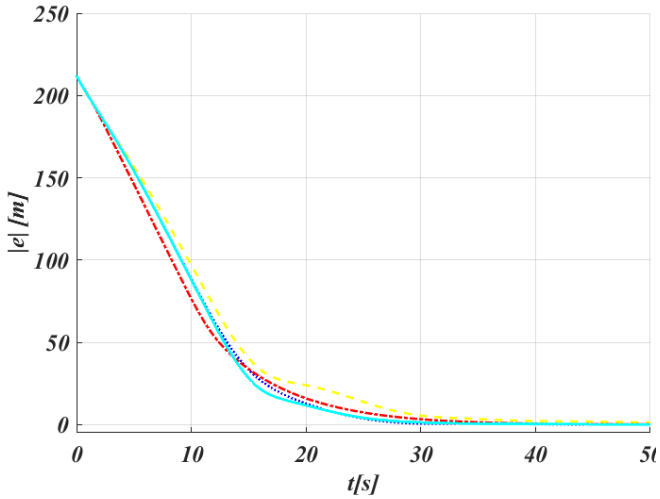
Initial condition	$r_m(t_0) = \begin{bmatrix} 190 \\ 0 \\ 0 \end{bmatrix}.m$	$\theta(t_0) = 1.5$ rad	$\emptyset(t_0) = 1.5$ rad
Design parameter	NLDGPFG/Case 1	$k = 0.02\ m^{-1}$	$\delta_{BL} = 100$ m
	NLDGPFG/Case 2	$k = 0.03\ m^{-1}$	$\delta_{BL} = 100$ m
	NLDGPFG/Case 3	$k = 0.1\ m^{-1}$	$\delta_{BL} = 100$ m
	NLDGPFG/Case 4	$k = 0.03\ m^{-1}$	$\delta_{BL} = 80$ m

The simulation of a circular path following can be demonstrated as shown in figure 6.



a) 3-D trajectory





b) Guidance command/error history

Figure 6. Circular path-following.

As shown in figures (6) the value of gain has affected the value of guidance command. By increasing the gain value, it reducing the overshoot value and the settling for a certain limit and increasing value of guidance command and settling time after this limit. Variation of δ_{BL} Doesn't have an effective effect on the normal acceleration but its effect on the over shoot and settling time as shown in case 4.

5.3. Helix path following

The desired path of the helix path is represented as a function of arc length as shown in equation (25) as follows:

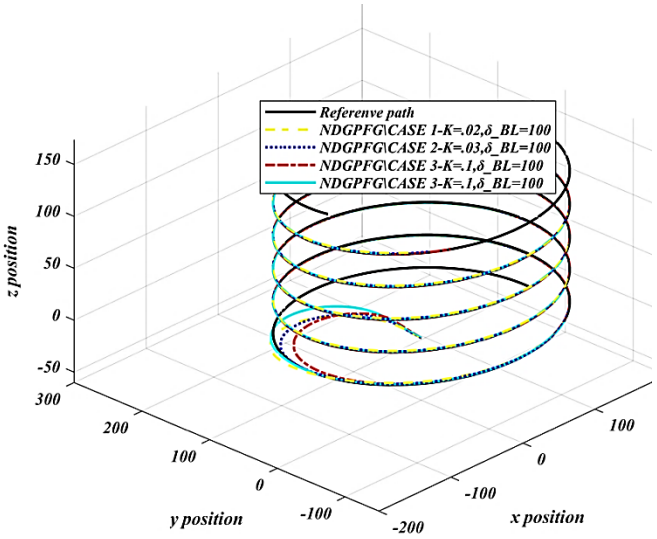
$$p(s) = \begin{bmatrix} 150\cos\left(\frac{s}{(5*901)^{\frac{1}{2}}}\right) \\ 1250\,s\,in\left(\frac{s}{(5*901)^{\frac{1}{2}}}\right) \\ \frac{5}{(5*901)^{\frac{1}{2}}} \end{bmatrix} m \tag{25}$$

The simulation parameters of circular path simulation are summarized in Table 3

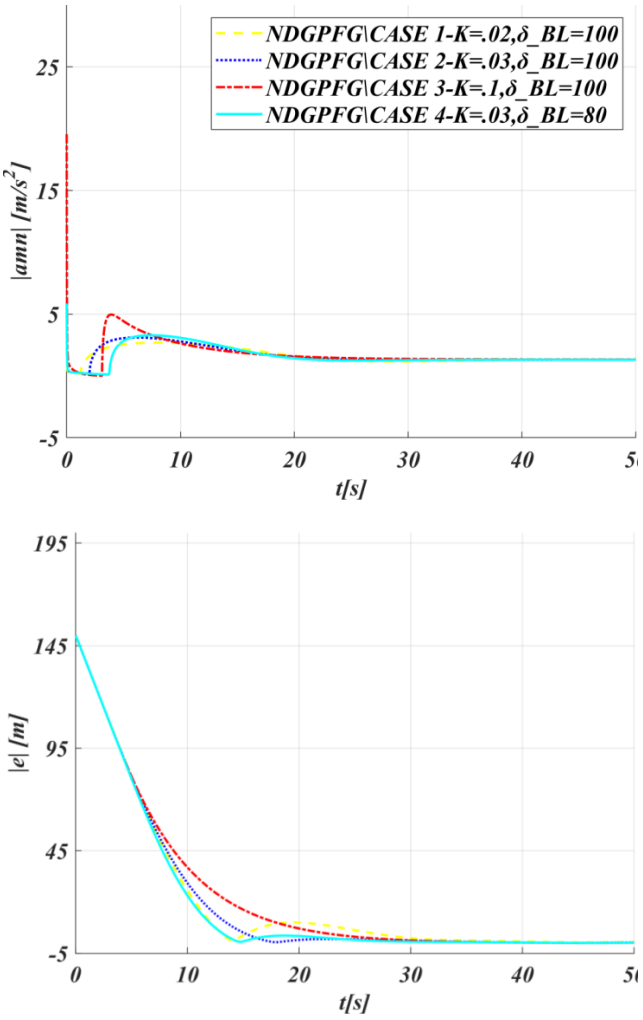
Table 3. Guidance law design simulation parameters

Initial condition	$r_m(t_0) = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$ m	$\theta(t_0) = 1.5$ rad	$\emptyset(t_0) = 0$ rad
Design parameter	NLDGPFG/Case 1	$k = 0.02\ m^{-1}$	$\delta_{BL} = 100$ m
	NLDGPFG/Case 2	$k = 0.03\ m^{-1}$	$\delta_{BL} = 100$ m
	NLDGPFG/Case 3	$k = 0.1\ m^{-1}$	$\delta_{BL} = 100$ m
	NLDGPFG/Case 4	$k = 0.03\ m^{-1}$	$\delta_{BL} = 80$ m

The simulation of a helix path following can be demonstrated as shown in figure 7



a) 3-D trajectory



b) Guidance command/error history

Figure 7. Helix path-following.

As shown in figures (7) normal acceleration increased in case 2 by increasing the guidance law gain and time settling in case 3 reduced by decreasing the value of guidance gain and Variation of δ_{BL} Doesn't have an effective effect on the normal acceleration but its effect on the over shoot and settling time. So, the values of k and δ_{BL}must be adjusted for optimum performance and minimum effort.

5.4. Three-dimensional cubic spline path following

The desired path is a 3D cube segment constructed between spatial points. For $i = 1,2,3, \ldots, N$ the segmental functions connecting the point and the next point, the function of arc length is represented as shown in equation (26) as follows:

$$p(s) = \begin{bmatrix} a_x^i(s-s_i)^3 + b_x^i(s-s_i)^2 + c_x^i(s-s_i) + d_x^i \\ a_y^i(s-s_i)^3 + b_y^i(s-s_i)^2 + c_y^i(s-s_i) + d_y^i \\ a_z^i(s-s_i)^3 + b_z^i(s-s_i)^2 + c_z^i(s-s_i) + d_z^i \end{bmatrix} \text{m} \quad (26)$$

The position of ith point can be represented as $r_{pi} = [x_i\ y_i\ z_i]^T$. the simulation parameters of cubic spline curve simulation are summarized in Tables 4 and 5

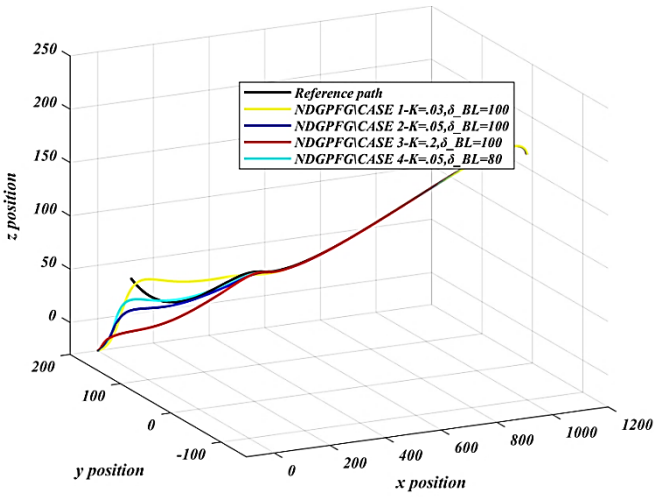
Table 4. Guidance law design simulation parameters

Initial condition	$r_m(t_0) = \begin{bmatrix} 0 \\ 200 \\ -30 \end{bmatrix}$ m	$\theta(t_0) = 2.1$ rad	$\emptyset(t_0) = 0.312$ rad
Design parameter	NLDGPFG/Case 1	$k = 0.03\ m^{-1}$	$\delta_{BL} = 100$ m
	NLDGPFG/Case 2	$k = 0.05\ m^{-1}$	$\delta_{BL} = 100$ m
	NLDGPFG/Case 3	$k = 0.2\ m^{-1}$	$\delta_{BL} = 100$ m
	NLDGPFG/Case 4	$k = 0.05\ m^{-1}$	$\delta_{BL} = 80$ m

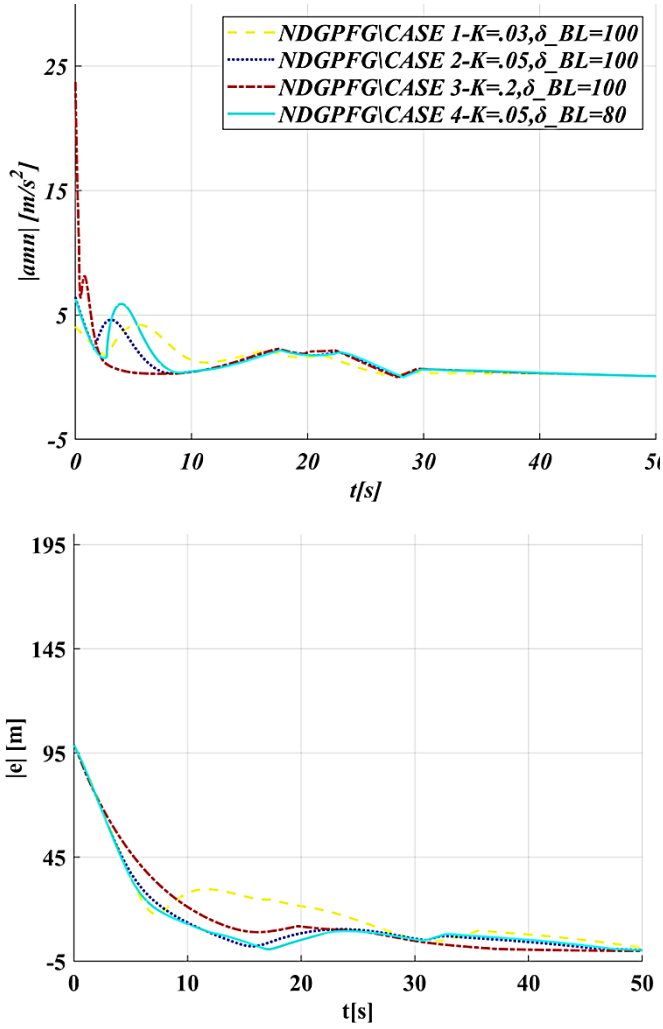
Table 5. Waypoints of cubic spline path

Variable	Value	Variable	Value
r_{p1}, m	$[30\ 150\ 50]^T$	r_{p5}, m	$[200\ -30\ \ 100]^T$
r_{p2}, m	$[30\ \ 70\ 50]^T$	r_{p6}, m	$[500\ -80\ \ 150]^T$
r_{p3}, m	$[50\ \ 0\ \ 80]^T$	r_{p7}, m	$[1000\ -100\ \ 200]^T$
r_{p4}, m	$[100\ -30\ 100]^T$		

The simulation of the cubic spline path following can be demonstrated as shown in figure 8.



a) 3-D trajectory



b) Guidance command/error history

Figure 8. 3-D cubic spline path-following.

The three-dimensional cubic spine path function of each segment is designed by Matlab by numerical methods, and there is an error in the description of the differential geometry of the path, but this error is eliminated by increasing the gains value of guidance law for each case as shown in Table 4 .for the guidance law performance concerning its design parameters by reducing the gains of the guiding law concerning k and similar to each path as shown in Figures (8), the normal acceleration increased in case 2 by increasing the gains of the guiding law and the time settlement decreased in case 3 by reducing the values of δ_{BL}Thus, the value of k and δ_{BL}must be adjusted to optimize performance and minimize efforts.

Conclusions

A nonlinear three-dimensional differential geometry path following guidance law was proposed. The proposed guidance law used a forward look vector to evaluate the forward look angle rather than the forward look points to satisfy the exact path following for general desired path. The desired path can be achieved by the proposed guidance law without any restrictions on the initial position and the velocity. The proposed guidance law has two design parameters (k, δ_{BL}) which affected the characteristics of the initial approach stage, such as overshooting, accelerating, peak, and stabilization. The numerical simulation is applied to a kinematic model of different types of desired paths such as straight lines, circular paths, helix paths, and three-dimensional cube spline paths, demonstrating that the proposed guidance law applies to the desired 3-D

path and how to adjust the design parameters(k, δ_{BL}) to optimize the tracking error and the value of the guidance command.

References

[1] Chen, S., D. F Laefer, and E.J.R.P.o.E. Mangina, *State of technology review of civilian UAVs.* 2016. **10**(3): p. 160-174.
[2] Mohsan, S.A.H., et al., *Towards the unmanned aerial vehicles (UAVs): A comprehensive review*. 2022. **6**(6): p. 147.
[3] Gupta, A., R. Padhi, and I.I.O.S.B.D.O.A.E.J.D. Document, *Nonlinear Geometric and Differential Geometric Guidance of UAVs with Vision Sensing for Reactive Obstacle Avoidance*. 2010.
[4] Sariff, N. and N. Buniyamin. *An overview of autonomous mobile robot path planning algorithms*. in *2006 4th student conference on research and development*. 2006. IEEE.
[5] Gasparetto, A., et al., *Path planning and trajectory planning algorithms: A general overview.* 2015: p. 3-27.
[6] Dhananjay, N., D.J.J.o.G. Ghose, Control,, and Dynamics, *Accurate time-to-go estimation for proportional navigation guidance*. 2014. **37**(4): p. 1378-1383.
[7] Babu, K.R., et al., *Switched bias proportional navigation for homing guidance against highly maneuvering targets.* 1994. **17**(6): p. 1357-1363.
[8] Liu, L., D. Wang, and Z.J.I.J.o.O.E. Peng, *ESO-based line-of-sight guidance law for path following of underactuated marine surface vehicles with exact sideslip compensation.* 2016. **42**(2): p. 477-487.
[9] Waldmann, J.J.I.T.o.c.s.t., *Line-of-sight rate estimation and linearizing control of an imaging seeker in a tactical missile guided by proportional navigation.* 2002. **10**(4): p. 556-567.
[10] Jeon, I.-S., J.-I. Lee, and M.-J.J.I.T.o.c.s.t. Tahk, *Impact-time-control guidance law for anti-ship missiles.* 2006. **14**(2): p. 260-266.
[11] Nelson, D.R., et al., *Vector field path following for miniature air vehicles.* 2007. **23**(3): p. 519-529.
[12] Nelson, D.R., et al. *Vector field path following for small unmanned air vehicles*. in *2006 American Control Conference*. 2006. IEEE.
[13] Park, S., et al., *Performance and lyapunov stability of a nonlinear path following guidance method.* 2007. **30**(6): p. 1718-1728.
[14] Park, S., J. Deyst, and J. How. *A new nonlinear guidance logic for trajectory tracking*. in *AIAA guidance, navigation, and control conference and exhibit*. 2004.
[15] Cho, N., et al., *Three-dimensional nonlinear differential geometric path-following guidance law*. 2015. **38**(12): p. 2366-2385.
[16] Ariff, O., et al., *Differential geometric guidance based on the involute of the target's trajectory*. 2005. **28**(5): p. 990-996.
[17] Cho, N., Y. Kim, and S.J.I.P.V. Park, *Three-dimensional nonlinear path-following guidance law based on differential geometry*. 2014. **47**(3): p. 2503-2508.
[18] Oprea, J., *Differential geometry and its applications*. 2007: MAA.
[19] Millman, R.S. and G.D.J. Parker, *Elements of differential geometry*. 1977.
[20] Sujit, P., S. Saripalli, and J.B. Sousa. *An evaluation of UAV path following algorithms*. in *2013 European Control Conference (ECC)*. 2013. IEEE.