2023 9th International Conference on Control, Decision and Information Technologies
CoDIT 2023 | Rome, Italy / July 03-06, 2023 Technically co-sponsored by IEEE & IFAC

Bézier Curve Based Guidance for Autonomous Landing of Quadrotor

Amit Shivam[1], Neon S[2] and Ashwini Ratnoo[3]

2023 9th International Conference on Control, Decision and Information Technologies (CoDIT) | 979-8-3503-1140-2/23/$31.00 ©2023 IEEE | DOI: 10.1109/CODIT58514.2023.10284179

Abstract— **This paper proposes Bézier curve based guidance for autonomous landing of quadrotor. The trajectory is generated using a fourth-order Bézier curve which satisfies curvature and heading constraints of the landing guidance problem. Numerical simulations validate the proposed tracking guidance law.**

Index Terms— **Bézier Curve, Landing Guidance**

I. INTRODUCTION

Autonomous landing is arguably the most intricate and challenging phase of UAV flight. UAVs must autonomously land in order to effectively complete the objectives in a variety of UAV applications, including surveillance, agricultural, and both civilian and military activities, such as search and rescue missions. Therefore, it is necessary to formulate a guidance law that gives the UAV steering instructions it needs to follow the specified trajectory and land on a static platform.

A typical Guidance, Navigation, and Control (GNC) architecture for a quadrotor involves the use of onboard sensors to measure the actual state of the quadrotor which is then processed by the navigation subsystem to estimate the current state. The guidance subsystem uses this information to generate the desired trajectory and provide attitude and thrust commands to the quadrotor's actuators, while the low-level control subsystem nullify any errors between the current and desired states. However, this paper solely focuses on the design of the trajectory guidance algorithm. In this regard several methods have been proposed to formulate the guidance algorithm for fixed-wing and rotor UAV's.

Autonomous landing for fixed wing UAV's on a stationary platform comprising of approach followed by descent phase are discussed in Refs. [1], [2]. Those works involve a constant glide slope during the approach phase, followed by an exponential path for descent, leading to a smooth touchdown. Another approach consists of a spiral maneuver during the approach phase, followed by a glide path towards a specified landing point is discussed in Ref. [3]. That method requires two user-defined waypoints; the approach point and the landing point. The UAV orbits around the approach point as it descends to a specified altitude, then it breaks out of its orbit and follows a glide slope to the landing point. Ref. [4] presents a bees inspired autonomous landing algorithm using optic flow information for sensing and controlling the height above the ground. At low heights, when optic flow is unreliable, stereo-based range information is used to guide descent close to touchdown. In Ref. [5] UAV attitude and speed profiles is expressed as exponential functions of horizontal separation between UAV and platform, which ensures smooth landing on that platform. These methods may not be appropriate for rotor UAVs since they do not rely on forward speed to generate lift.

Multirotor UAVs have capabilities for hovering, but this has limited most of the guidance-based landing techniques to vertical landing [6]–[8]. In Ref. [6] quadrotor tracks a moving target along a linear path until it is directly above the landing pad and then descends vertically to land. Similarly, Ref. [7] proposes vector field method for hovering and landing. In Ref. [8] proposes a log of polynomial velocity profile with two design parameters for vertical descent. Therein, the closing velocity of UAV to target logarithmically decays to zero as the distance to landing point goes to zero. It is to be noted that this hovering and landing approaches increases flight time. In contrast, Refs [9]–[11] provides a smooth trajectory from the UAV's initial point to its final landing point avoiding the need to hover before vertical descent. Ref. [9] uses pure pursuit based guidance law to land on a stationary target with explicit control over the UAV velocity proposed as a log of polynomial. Ref. [10] uses vector-fields for autonomous landing. Therein, the desired trajectory is a combination of three straight line segments that connects initial UAV position to the landing site. Further, the discontinuous trajectory is approximated by smooth function using Maslov Dequantization. Another approach is where the trajectory expressed as a polynomial function of time [11] and the coefficients of the polynomial are determined satisfying constraints on initial and final position. Refs. [12]–[15] uses vision based approach for autonomous landing of quadrotors on static and moving platforms. Vision-based landing algorithms require image processing techniques to extract the position information of the UAV using the images captured by the onboard cameras(stereo vision, monocular) which is computationally demanding.

As the main contribution of this work, desired quadrotor trajectory is generated using fourth-order Bezier curve which satisfies heading and curvature constraints. The guidance algorithm proposed in the paper requires UAV position information with respect to the landing point which can be acquired using simple range and bearing measuring onboard sensors. The proposed approach also eliminates the requirement of hovering before vertical landing. Overall, the

[1] Amit Shivam is a Graduate student in Autonomous Vehicle Laboratory, Department of Aerospace Engineering, Indian Institute of Science, Bengaluru, Karnataka, India, amitshivam@iisc.ac.in

[2] Neon S. is a Research Associate in Autonomous Vehicle Laboratory, Department of Aerospace Engineering, Indian Institute of Science, Bengaluru, Karnataka, India, neons@alum.iisc.ac.in

[3] Ashwini Ratnoo is an Associate Professor in Department of Aerospace Engineering, Indian Institute of Science, Bengaluru, Karnataka, India, ratnoo@iisc.ac.in

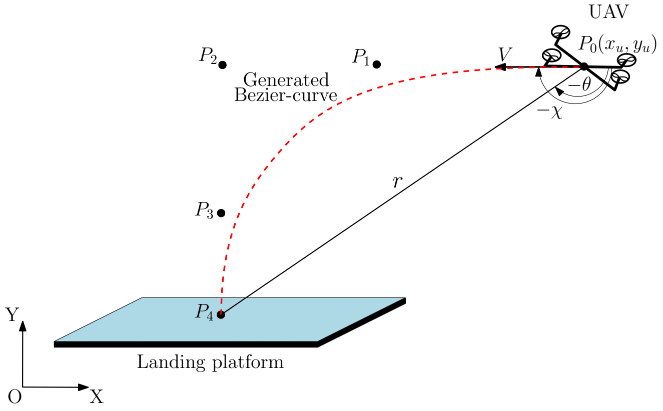


Fig. 1: Landing scenario

proposed guidance method is computationally inexpensive and easy to implement.

The remainder of this paper is organized as follows: The problem statement is defined in Section II. Section III discusses the methodology used in trajectory planning. Tracking guidance law is discussed in Sec. IV. Section V presents simulation studies and Sec. VI concludes this paper.

II. PROBLEM STATEMENT

Consider a quadrotor with its instantaneous Cartesian position is given by $(x_u, y_u) \in \mathbb{R}^2$ as shown in Fig. 1. Quantities r and θ represent the instantaneous position and orientation of the quadrotor in polar coordinates, respectively. V and χ denote the UAV speed and heading angle, respectively. The kinematic equations of motion of the UAV can be described as

$$\dot{r} = V\cos(\chi - \theta) \tag{1}$$

$$\dot{\theta} = \frac{V}{r}\sin(\chi - \theta) \tag{2}$$

The objective is to generate a feasible trajectory and formulate the tracking guidance command χ which governs the instantaneous heading angle of quadrotor to track the proposed trajectory. Furthermore, the trajectory is desired to satisfy the following quadrotor heading angle χ and curvature κ constraints.

$$\chi(t=0) = \chi_i \qquad \chi(t=t_f) = \chi_f \tag{3}$$

$$\kappa(t=0) = \kappa(t=t_f) = 0 \tag{4}$$

In Eqs. (3) and (4), subscripts i and f denote corresponding initial and final quantities, respectively. To accomplish the above noted objectives, a fourth order Bézier curve based trajectory is proposed in this work as shown in red dashed curve in Fig. 1. The curve is characterized by five control points $P_i : i = \{0,1,2,3,4\}$ shown in Fig. 1. The control points P_0 and P_4 corresponds to the initial and final position of the quadrotor. The remaining three control points P_1, P_2, P_3 account for the design flexibility of the trajectory which will be discussed subsequently. Next section briefly introduces the generic Bézier curve and subsequently discusses specific fourth-order Bézier curve with the formulation of the guidance command.

III. TRAJECTORY PLANNING

A. Salient features of Bézier curve

A Bézier curve is a parametric curve which uses Bernstein polynomials as a basis function [16]. The formula for general n^{th} order Bézier curve $Q(\tau) = (x(\tau), y(\tau))$ is given by

$$Q(\tau) = (x(\tau), y(\tau)) = \sum_{i=0}^{n} P_i B_{i,n}(\tau) \tag{5}$$

where, P_i represents control points and $\tau = [0,1]$ is the curve parameter. The Bernstein polynomial $B_{i,n}(\tau)$ is given as

$$B_{i,n}(\tau) = \binom{n}{i} \tau^i (1-\tau)^{n-i} \tag{6}$$

The n^{th} order Bézier curve $Q(\tau)$ with the control points $P_0, P_1, ..., P_n$ has the following properties.

1) Endpoints are the first and last control points.

$$\tau = 0 \Rightarrow B_{0,n}(0) = 1 \Rightarrow Q(0) = P_0 \tag{7}$$

$$\tau = 1 \Rightarrow B_{n,n}(0) = 1 \Rightarrow Q(1) = P_1 \tag{8}$$

2) Curve is tangent to the control polygon at the endpoints.

$$\frac{dQ(\tau)}{d\tau} = \sum_{i=0}^{n} P_i \dot{B}_{i,n}(\tau) \tag{9}$$

$$\dot{Q}(0) = n(P_1 - P_0) \qquad \dot{Q}(1) = n(P_n - P_{n-1}) \tag{10}$$

3) Bézier curve is invariant under affine transformation (translation and rotation).
4) Bézier curve always lies inside the convex hull of it's control points.

The tangent-to-curve $\gamma(\tau)$ and curvature $\kappa(\tau)$ variation along the trajectory is

$$\gamma(\tau) = \tan^{-1}\left(\frac{dy(\tau)/d\tau}{dx(\tau)/d\tau}\right) \tag{11}$$

$$\kappa(\tau) = \frac{d\chi(\tau)/d\tau}{ds/d\tau} \tag{12}$$

where $ds/d\tau = \sqrt{\left(dx(\tau)/d\tau\right)^2 + \left(dy(\tau)/d\tau\right)^2}$. Using Eq. (11) and Eq. (12), the curvature can be deduced as

$$\kappa(\tau) = \frac{\dot{x}(\tau)\ddot{y}(\tau) - \dot{y}(\tau)\ddot{x}(\tau)}{\left(\dot{x}(\tau)^2 + \dot{y}(\tau)^2\right)^{3/2}} \tag{13}$$

where, terms $\dot{x}(\tau) = \frac{dx(\tau)}{d\tau}$, $\dot{y}(\tau) = \frac{dy(\tau)}{d\tau}$, $\ddot{x}(\tau) = \frac{d^2x(\tau)}{d\tau^2}$ and $\ddot{y}(\tau) = \frac{d^2y(\tau)}{d^2\tau}$.

B. Fourth order Bézier curve

The fourth order Bézier curve consisting of five control points $P_i = (x_i, y_i); i = \{0,1,2,3,4\}$ is represented as

$$\begin{bmatrix} x(\tau) \\ y(\tau) \end{bmatrix} = \sum_{i=0}^{4} P_i B_{i,4}(\tau) \tag{14}$$

Differentiating Eq. (14) with respect to τ

$$\left[\frac{dx(\tau)}{d\tau} \; \frac{dy(\tau)}{d\tau}\right] = AP \tag{15}$$

where,

$$A = \begin{bmatrix} -4(1-\tau)^3 & 4(1-\tau)^2(1-4\tau) \\ & 12\tau(1-\tau)(1-2\tau) & 4\tau^2(3-4\tau) & 4\tau^3 \end{bmatrix} \tag{16}$$

and

$$P = \begin{bmatrix} P_0 & P_1 & P_2 & P_3 & P_4 \end{bmatrix}^T \tag{17}$$

The tangent-to-curve is given as

$$\gamma(\tau) = \tan^{-1}\left(\frac{dy(\tau)/d\tau}{dx(\tau)/d\tau}\right) \tag{18}$$

Using Eq. (7) and Eq. (8) initial and final control points of the proposed trajectory are $P_0 = (x_0, y_0)$ and $P_4 = (x_4, y_4)$. It is desired that the tangent-to-curve is $-\pi$ at P_0 and $-\pi/2$ at P_4, which is ensured by choosing control points P_1 and P_3 collinear to P_0 and P_4, respectively. Further, zero curvature of the Bezier curve at points P_0 and P_4 is achieved by placing the control point P_2 to be collinear to both the tangent segments P_0P_1 and P_3P_4. The equations of tangent segments P_0P_2 and P_2P_4 can be expressed as

$$y_2 - y_0 = m_1 (x_2 - x_0) \tag{19}$$

$$y_2 - y_4 = m_2 (x_2 - x_4) \tag{20}$$

Solving Eqs. (19) and (20) for $P_2 (x_2, y_2)$ leads to

$$\left(\frac{(y_0 - m_1x_0) - (y_4 - m_2x_4)}{m_2 - m_1}, \frac{m_1m_2 (x_4 - x_0) + (m_2y_0 - m_1y_4)}{m_2 - m_1}\right) \tag{21}$$

Here, slopes $m_1 = -\tan\chi_i$ and $m_2 = -\tan\chi_f$ are derived from the tangent-to-curve constraints given in Eq. (3). The control points P_1 and P_3 are determined using

$$P_1(x_1, y_1) = \left(\frac{\alpha x_2 + x_0}{\alpha + 1}, \frac{\alpha y_2 + y_0}{\alpha + 1}\right) \tag{22}$$

and

$$P_3(x_3, y_3) = \left(\frac{\alpha x_4 + x_2}{\alpha + 1}, \frac{\alpha y_4 + y_2}{\alpha + 1}\right) \tag{23}$$

where, $\alpha \in (0, 1]$ is the quantity with which the tangent segments P_0P_2 and P_2P_4 are divided by control points P_1 and P_3, respectively.

IV. Tracking Guidance Law

In order to steer the quadrotor along the designed Bézier curve discussed in Section. 3, we must determine the appropriate heading angle, χ. We uniformly divide the parametric domain $\tau = [0, 1]$ of Bézier curve $Q(\tau) = (x(\tau), y(\tau))$ into 2^n sub-intervals as $0 = \tau_0 < \tau_1 < ... < \tau_i < ... < \tau_n = 1$. Let, P_{τ_i} be the approximate path length of Bézier curve obtained at i^{th} instant as,

$$P_{\tau_i} = P_{\tau_{i-1}} + \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2} \tag{24}$$

Similarly, let the parametric domain of kinematic equations given in Eqs. (1) and (2) , $t = [0, t_f]$ be divided arbitrarily during numerical integration into sub-intervals $0 = t_0 < t_1 < ... < t_i < ... < t_n = t_f$. Let, L_{τ_i} be the path length obtained by quadrotor motion at i^{th} instant of numerical integration as,

$$L_{t_i} = V \times t_i \tag{25}$$

TABLE I: Parameters for Case 1

Control points	Coordinates
P_0	(10 m, 10 m)
P_4	(0 m, 0 m)
P_2	(0 m, 10 m)
P_1, P_3	(5 m, 10 m),(0 m, 5 m)

We interpolate P_{τ_i} to obtain that instantaneous τ_{ins} that equals $P_{\tau_i} = L_{t_i}$, re-parameterizing Bézier curve with $\tau_{\text{new}} = [0, 1]$. We now have one-to-one correspondence map, $t = [0, t_f] \rightarrow \tau_{\text{new}} = [0, 1]$. The guidance command can then be derived as,

$$\chi(t) = \gamma(\tau_{\text{new}}) = \tan^{-1}\left(\frac{dy(\tau_{\text{new}})/d\tau_{\text{new}}}{dx(\tau_{\text{new}})/d\tau_{\text{new}}}\right) \tag{26}$$

It can be readily seen from Eq. (26) that $\chi(t = 0) = -180$ deg. and $\chi(t = t_f) = -90$ deg. which is our guidance objective, that is, to satisfy Eqs. (3) and (4). This attributes to the generation of Bézier curve, which involves carefully selecting the appropriate control points as discussed in the preceding section. After considering the analysis that has been elucidated thus far, the process for determining control points P_i and subsequently generating the guidance command using Eq. (26) is explicated in Algorithm 1.

Algorithm 1 Guidance logic

```
Input: $P_0, P_4$.
Compute intermediate control points $P_2, P_1, P_3$ using Eqs. (21), (22) and (23).
Using control points $P_i$, generate Bézier curve using Eq. (14).
Compute guidance command $\chi_d$ using Eq. (26).
```

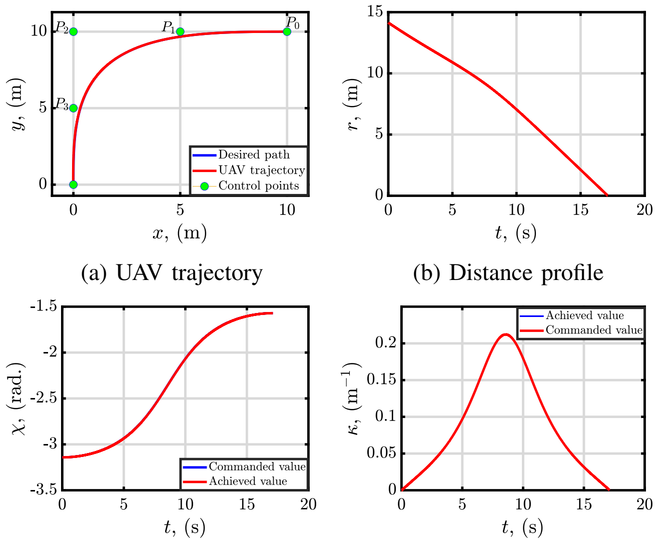
V. Simulation Results

This section discusses numerical simulation results of the proposed method . Simulation results consider variety of scenarios to generate landing trajectory while satisfying design constraints discussed in Section I. The simulation studies are carried out using first order heading control described as

$$\dot{\chi} = k (\chi_d - \chi) \tag{27}$$

where, commanded course angle χ_d is obtained using Eq. (26) and k is the controller gain. Unless specified, speed of UAV is kept constant at 1 m/s and $\alpha = 0.5$.

1) Case 1: Sample landing scenario: The landing trajectory of quadrotor is generated using parameters listed in Table I. Eq. (21) gives control point P_2. Control points P_1



(a) UAV trajectory (b) Distance profile

(c) Heading angle profile (d) Curvature profile

Fig. 2: Results for landing scenario

TABLE II: Parameters for Case 2

α	$P_1(x_1\ \mathrm{m}, y_1\ \mathrm{m})$	$P_3(x_3\ \mathrm{m}, y_3\ \mathrm{m})$
0.1	$(9,10)$	$(0,9)$
0.3	$(7,10)$	$(0,7)$
0.5	$(5,10)$	$(0,5)$
0.7	$(3,10)$	$(0,3)$
0.9	$(1,10)$	$(0,1)$

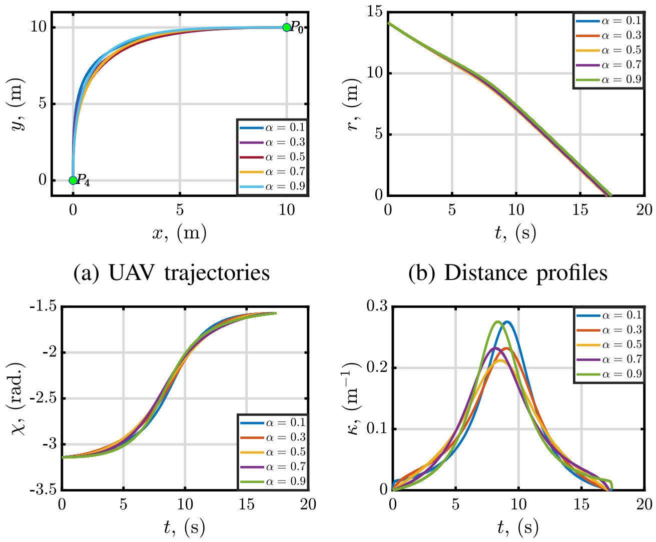
and P_3 are calculated using Eq. (22) and (23) with $\alpha = 1$ as

$$P_1 = (x_1, y_1) = \left(\frac{x_0 + x_2}{2}, \frac{y_0 + y_2}{2}\right) \tag{28}$$

$$P_3 = (x_3, y_3) = \left(\frac{x_2 + x_4}{2}, \frac{y_2 + y_4}{2}\right) \tag{29}$$

Figure 2 shows the simulation results. Figure 2a shows UAV trajectory tracks the desired path leading to the landing point. Distance variation which is reducing to zero is shown in Fig. 2b. Figure 2c shows the heading angle profile satisfying initial and final heading angle constraints, that is, $\chi(t=0) = -\pi$ rad. and $\chi(t=t_f) = -\pi/2$ rad., respectively. Figure 2d shows the corresponding curvature profile which is zero at $\kappa(t=0)$ and $\kappa(t=t_f)$.

2) Case 2: Different control points P_1, P_3: In this case, the control points P_1 and P_3 are computed by dividing tangents P_0P_2 and P_2P_4 at different ratios. The ratios at which the control points P_1 and P_3 divides the tangents P_0P_2 and P_2P_4 is denoted by α. The control points P_0, P_2 and P_4 are fixed as used in Case 1. The control points P_1 and P_3 corresponding to each $\alpha = \{0.1, 0.3, 0.5, 0.7, 0.9\}$ is computed using Eq. (22) and Eq. (23) and are tabulated in Table II. The simulation results are shown in Fig. 3. Multiple landing trajectories are generated in Fig. 3a. These trajectories have fixed initial and final points, but the positions of control points P_1 and P_3 are varied as listed in Table II. All of these landing trajectories lead to the same landing point, as shown in Fig. 3b, where the distance profile goes down to zero. Furthermore, Fig. 3c displays the variation of the heading angle profile, while Fig. 3d shows the corresponding curvature profiles that satisfy trajectory constraints. This simulations show high design flexibility of Bezier trajectory using varying α while satisfying terminal trajectory constraints. Choice of α affects the maximum curvature as evident from Fig. 3d. It can be readily seen in Fig. 3d that $\alpha = 0.5$ corresponds to the lowest maximum curvature whereas, $\alpha = 0.1$ and $\alpha = 0.9$ leads to the highest maximum curvature.



(a) UAV trajectories (b) Distance profiles

(c) Heading angle profiles (d) Curvature profiles

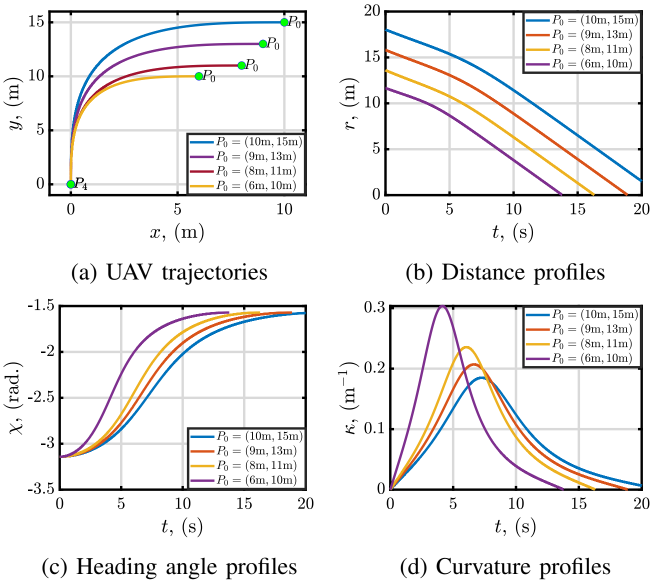
Fig. 3: Results for different control points

TABLE III: Parameters for Case 3

$P_0(x_0\ \mathrm{m}, y_0\ \mathrm{m})$	$P_1(x_1\ \mathrm{m}, y_1\ \mathrm{m})$	$P_2(x_2\ \mathrm{m}, y_2\ \mathrm{m})$	$P_3(x_3\ \mathrm{m}, y_3\ \mathrm{m})$
$(10,15)$	$(0,15)$	$(0,15)$	$(0,7.5)$
$(9,13)$	$(4.5,13)$	$(0,13)$	$(0,6.5)$
$(8,11)$	$(4,11)$	$(0,11)$	$(0,5.5)$
$(6,10)$	$(3,10)$	$(0,10)$	$(0,5)$

3) Case 3: Varying control point P_0: In this case, multiple initial control points P_0 with fixed final control point $P_4 = (0\ \mathrm{m}, 0\ \mathrm{m})$, is considered for simulation studies. The corresponding intermediate control points P_2, P_1, P_3 are computed accordingly, using Eqs. (21), (28) and (29), respectively and are listed in Table III. Fig. 4 illustrates the simulation results. Figure 4a shows multiple trajectory profiles. Distance profiles are shown in Fig. 4b, which reduces to zero. Figs. 4c and 4d shows heading and curvature plots satisfying the corresponding heading and curvature constraints, respectively.

4) Case 4: Varying initial heading angle χ_0: In this case, multiple initial heading angles χ_0 are considered, with fixed $P_0 = (10\ \mathrm{m}, 10\ \mathrm{m})$ and $P_4 = (0\ \mathrm{m}, 0\ \mathrm{m})$, for simulation studies. With $\alpha = 0.1$, the corresponding intermediate control points P_2, P_1, P_3 are computed accordingly, using Eqs. (21), (28) and (29), respectively and are listed in Table IV. The simulation results are presented in Fig 5. Figure 5a shows multiple trajectory profiles with the corresponding distance



(a) UAV trajectories

(b) Distance profiles

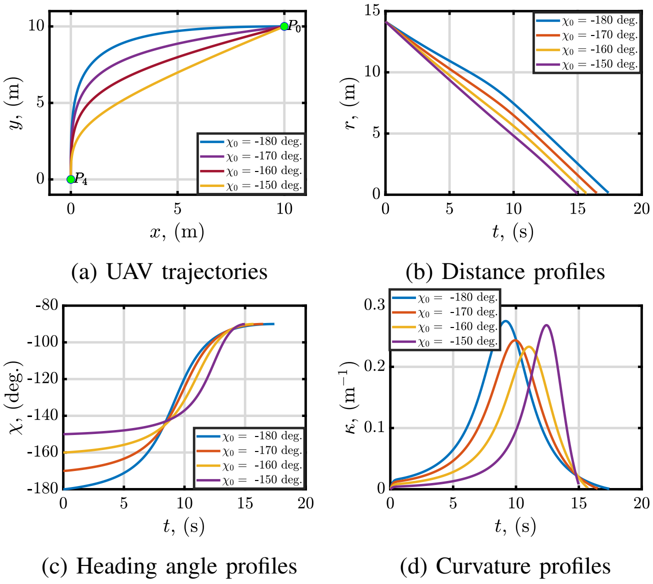
(c) Heading angle profiles

(d) Curvature profiles

Fig. 4: Results for multiple P_0 landing scenarios

TABLE IV: Parameters for Case 4

$\chi_0$deg.	$P_1(x_1$ m$, y_1$ m$)$	$P_2(x_2$ m$, y_2$ m$)$	$P_3(x_3$ m$, y_3$ m$)$
-180	$(5, 10)$	$(0, 10)$	$(0, 5)$
-170	$(5, 9.12)$	$(0, 8.24)$	$(0, 4.12)$
-160	$(5, 8.18)$	$(0, 6.36)$	$(0, 3.18)$
-150	$(5, 7.11)$	$(0, 4.23)$	$(0, 2.11)$



(a) UAV trajectories

(b) Distance profiles

(c) Heading angle profiles

(d) Curvature profiles

Fig. 5: Results for different initial heading angle χ_0

profiles attenuates to zero as indicated in Figure 5b. The heading angle profiles, which satisfy the desired heading constraints, are plotted in Figure 5c. Figure 5d shows the corresponding curvature profiles. This case study demonstrates that the final and the initial curvature constraint along with the final heading requirement is achieved even though the initial heading constraint is relaxed.

VI. Conclusions

In this study, we propose a new guidance approach for autonomous quadrotor landing using Bezier curves. The key contribution is the design of control points for the curve that adheres to the quadrotor's heading and curvature constraints along the curve. The efficacy of the proposed guidance approach is validated through numerical simulations.

References

[1] B. Prasad, and S. Pradeep, "Automatic Landing System Design Using Feedback Linearization Method ", *In AIAA Infotech@ Aerospace 2007 Conference and Exhibit.*

[2] F. B. Hsiao, W. L. Chan, Y. C. Lai, L.C. Tseng, S. Y. Hsieh, and H. K. Tenn, "Landing Longitudinal Control System Design for a Fixed-wing UAV ", *In 45th AIAA Aerospace 2007 Sciences Meeting and Exhibit.*

[3] D. B. Barber, S. R. Griffiths, T. W. McLain, and R. W. Beard., "Autonomous Landing of Miniature Aerial Vehicles ", *Journal of Aerospace Computing, Information, and Communication,* Vol. 4, No. 5, 2007, pp. 770-784.

[4] S. Thurrowgood, R. J. Moore, D. Soccol, M. Knight, and M. V. Srinivasan, "A Biologically Inspired Vision-Based Guidance System for Automatic Landing of a Fixed-Wing Aircraft ", *Journal of Field Robotics,* Vol. 31, No. 4, 2014, pp. 699-727.

[5] S. Suresh, and A. Ratnoo, "Fixed-Wing UAV Guidance for Autonomous Landing on a Translating Platform ", *In AIAA Scitech 2019 Forum.*

[6] M. T. Alkowatly, V. M. Becerra, and W. Holderbaum, "Bioinspired Autonomous Visual Vertical Control of a Quadrotor Unmanned Aerial Vehicle ", *Journal of Guidance, Control, and Dynamics*, Vol. 38, No. 2, 2015, pp. 249-262.

[7] J. W. Kwon, J. H. Kim, and J. Seo, "Vector Field Guided Auto-Landing Control of Airship With Wind Disturbance ", *IFAC Proceedings Volumes,* Vol. 47, No. 3, 2014, pp. 1114-1119.

[8] A. Gautam, A. Ratnoo, and P. B. Sujit, "Log polynomial Velocity Profile for Vertical Landing ", *Journal of Guidance, Control, and Dynamics,* Vol. 41, No. 7, 2018, pp.1617-1623.

[9] A. Gautam, P. B. Sujit, and S. Saripalli, "Autonomous Quadrotor Landing Using Vision and Pursuit Guidance ", *IFAC-Papers On Line*, Vol. 50, No. 1, 2017, pp. 10501-10506.

[10] V. M. Gonçalves, R. McLaughlin and G. A. S. Pereira, "Precise Landing of Autonomous Aerial Vehicles Using Vector Fields ", *In IEEE Robotics and Automation Letters,* Vol. 5, No. 3, 2020, pp. 4337-4344.

[11] B. M. Min, M. J. Tahk, H. C. D. Shim, and H. C. Bang, "Guidance Law for Vision-Based Automatic Landing of UAV ", *International Journal of Aeronautical and Space Sciences,* 2007, Vol. 8, No. 1, pp.46-53.

[12] B. Herissé, T. Hamel, R. Mahony, and F. X. Russotto, "Landing a VTOL Unmanned Aerial Vehicle on a Moving Platform Using Optical Flow ", *IEEE Transactions on robotics,* 2011, Vol. 28, No. 1, pp.77-89.

[13] J. Park, Y. Kim, and S. Kim, "Landing Site Searching and Selection Algorithm Development Using Vision System and its Application to Quadrotor ", *IEEE Transactions on Control Systems Technology,* 2014, Vol. 23, No. 2, pp.488-503.

[14] R. O. de Santana, L. A. Mozelli, and A. A. Neto, "Vision-Based Autonomous Landing for Micro Aerial Vehicles on Targets Moving in 3D Space ", *In IEEE International Conference on Advanced Robotics (ICAR)*, 2019, pp. 541-546.

[15] R. Balasubramaniam, and P. B. , Sujit, "A Cooperative Framework for Autonomous Landings of Quadrotors using Vision on a Moving UGV ", *In AIAA Scitech 2021 Forum.*

[16] H. Prautzsch, W. Boehm, and M. Paluszny, "Bezier and B-spline Techniques ", 1st ed., *Springer*, 2002, ch. 2, pp. 9-22.