Trajectory Estimation for Geo-Fencing Applications on Small-Size Fixed-Wing UAVs

Mirco Theile[1*], Simon Yu[2*], Or D. Dantsker[1] and Marco Caccamo[1]

Abstract— **The steadily increasing popularity of Unmanned Aerial Vehicles (UAVs) is creating new opportunities in diverse fields of technology and business. However, this increase of popularity also raises safety concerns. To tackle the primary concern of keeping the UAV inside a designated region, a novel trajectory estimation algorithm for geo-fencing applications is proposed. We derive the Beta-Trajectory that takes into account constraints in curvature as well as constraints in the change of curvature which is bounded by the maximum roll-rate of the aircraft. We incorporate the Beta-Trajectory into a geo-fencing algorithm. By using our open-source uavAP autopilot, the applicability and necessity of accurate trajectory estimation algorithms for geo-fencing applications are shown on small fixed-wing aircraft. The model and algorithm are validated in high-fidelity simulations as well as in real flight testing.**

I. INTRODUCTION

In recent years, we have seen an uptrend in the popularity of Unmanned Aerial Vehicles (UAVs) driven by the desire to apply these aircraft to areas such as precision farming, infrastructure and environment monitoring, surveillance, surveying, and mapping, search and rescue missions, weather forecasting, and much more. All the above application scenarios require the aircraft to safely interact with the surrounding humans, environments, and other aircraft. Therefore, unmanned aircraft should be constrained to a designated area or space defined by a geo-fence.

For rotary aircraft, such as quadcopters, the task of staying inside the geo-fence is relatively simple since those type of aircraft are capable of stopping in mid-air and turning around with zero translational velocity. For fixed-wing aircraft, on the other hand, such execution of maneuvers is impossible as they need to maintain a minimum velocity in order to stay airborne. Consequently, a proper kinematic model for fixed-wing aircraft is required to determine the feasibility of a trajectory as well as the exact time for the initiation of an evasion maneuver. Most analytical kinematic models only constrain the maximum curvature of a trajectory, namely a Dubin's Curve ([1], [2]). In the context of geo-fencing, a constraint in the change of curvature has been widely ignored in the literature. The main contributions of this work are:

1) A precise kinematic model for a fixed-wing aircraft with constrained roll rate.
2) A geo-fencing algorithm using the model to avoid boundaries and stay in a designated area.
3) An implementation of the model and algorithm using our open-source uavAP autopilot.
4) An evaluation of the model and algorithm using high-fidelity simulators as well as real flight data.

To the best of our knowledge, this is the first geo-fencing algorithm that takes into account the constraint in the change of curvature.

A. *Rationale and Related Work*

The related work on geo-fencing applications mainly focuses on multicopters ([3]–[8]). As previously mentioned, the kinematic model of a fixed-wing aircraft is fundamentally different from that of a multicopter. Therefore, geo-fencing methods derived for multicopters cannot directly be adopted for fixed-wing aircraft. The authors in [7] look at both types of aircraft. They argue that the trajectories for a fixed-wing aircraft form a symmetric fan pattern around the velocity vector. This fan pattern, however, is based on the instantaneous change in roll, which we show, is not applicable for fixed-wing aircraft.

In related work done by [8], the velocity control of the aircraft is assumed to be instantaneous, meaning that its roll rate can be as high as infinity, giving an instantaneous change of roll angle. The work done by [9] demonstrates the issue of such an assumption by presenting discontinuity in curvature when flying with a circle-line-circle Dubin's path. In real-life scenarios, on the other hand, the curvature, as well as the roll rate of the aircraft, are limited. Finally, a variety of geo-fencing strategies are introduced in the related work. For instance, [8] limits the control space of the aircraft instead of completely overriding its mission when close to the geo-fence. In this work, on the other hand, the geo-fencing technique is similar to the technique mentioned in [7], which defines a soft boundary at a maximum distance that the aircraft can travel once the threat of breaking the geo-fence is detected. The authors in [10] developed an algorithm that defines different levels of safe zones to determine the necessity of an evasive maneuver. They use the Dubin's curve as their kinematic model and use slack variables to account for the deviation from it.

The authors in [11] describe a kinematic model that takes into account a constraint in the change of curvature for the purposes of optimal path planning. The approximation in this approach is only accurate for up to 30° of roll angle, which is not applicable for a geo-fencing algorithm where the

---

*The first two authors contributed equally to this work.

[1] Mirco Theile, Or D. Dantsker, and Marco Caccamo are with the TUM Department of Mechanical Engineering, Technical University of Munich, Germany {`mirco.theile, or.dantsker, mcaccamo`}`@tum.de`

[2] Simon Yu is with the Dept. of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, IL `jundayu2@illinois.edu`

Aircraft Model	**Wing-span (m)**	**Cruise Velocity (m/s)**	**Max Roll (°)**	**Max Roll Rate (°/s)**	**0° Roll Deviation (%)**	**Max Roll Deviation (%)**
Avistar	1.6	22.0	45.0	30.0	37.3	133.3
Pulsar	4.0	12.0	45.0	33.8	60.5	215.5
Cessna 177	10.8	53.6	45.0	54.5	8.4	30.2
Boeing 747	59.6	263.0	30.0	19.1	1.8	6.8

TABLE I: The deviation is given for two types of rolling maneuvers, from 0° to the maximum roll angle and from negative maximum roll to positive maximum roll angle. Data taken from: Cessna 177 [14], Boeing 747 [15], Pulsar sailplane from our flight measurements, and Avistar which is constrained to protect on-board equipment; the notion of the deviations is described in Section II.

maximum turn capability has to be exploited. We compare our approach to this approach as well as the Dubin's curve and show that we have higher accuracy than both of them while maintaining the computability as in [11].

Table I shows the deviation, the ratio of the distance between predicted orbit centers and the orbit radius (Figure 1), of the Dubin's Curve from our Beta-Trajectory given four example aircraft. The deviation shown is higher when the total velocity is lower, which is crucial for small fixed-wing UAVs such as the Pulsar by F5Models sailplane [12], [13] with a deviations of $60-215\%$. In addition, the deviation of middle-size (general aviation) aircraft like the Cessna 177 of $8-30\%$ is also significant and should be considered in trajectory planning. For large size aircraft like the Boeing 747, however, this effect can mostly be ignored.

This work is structured as follows: Section II contains the derivation of the Beta-Trajectory model and the percent deviation of the Dubin's Curve showed in Table I. Section III defines the geo-fencing algorithm using the derived Beta-Trajectory model. In Section IV, the model and the algorithms are validated using the simulator and real flight data. Finally, Section V concludes this work and gives an outline of future work.

II. DERIVATION OF KINEMATIC MODEL

This section shows the derivation of the Beta-Trajectory, describing the frames of reference and presenting the deviations of the Dubin's Curve from the Beta-Trajectory as contrast metrics. Additionally, we discuss the difference between the Beta-Trajectory and the approach in [11]. The underlying assumptions for the Beta-Trajectory are the following:

1) A rolling maneuver utilizes an approximately constant roll rate.
2) The aircraft's altitude and velocity stay constant during the maneuver.
3) The effects of wind are set to zero.

The argument for assumption 2 is that a change in altitude would add additional constraints to the system due to maximum and minimum altitude bounds. Furthermore, an autopilot is capable of keeping the altitude constant during a rolling maneuver by applying a corresponding pitching actuation. Regarding assumption 3, this algorithm is developed for small fixed-wing aircraft which often lack the necessary instrumentation to evaluate wind speed and direction. Therefore the presented approach utilizes a slack variable to account for the unknown wind effects. Modeling wind as a steady, uniform flow-field as done in [11] would only be a superposition between a translational motion and the Beta-Trajectory.

A. Beta-Trajectory

The derivation of the trajectory with constant roll rate is based on the relation between the aircraft's roll angle ϕ and its yaw rate $\dot{\psi}$. Their relation can be expressed as follows:

$$\dot{\psi}(t) = -\frac{g}{V}\tan\phi(t) \tag{1}$$

where g is the gravitational constant and V is the total velocity of the aircraft. The relation demonstrates the contribution of the rotated lift-force to the centripetal force acting on the aircraft. The constant roll angle rate $\dot{\phi}_c$ influences the roll angle ϕ by

$$\phi(t) = \dot{\phi}_c t + \phi_0 \tag{2}$$

where ϕ_0 is the initial roll angle of the aircraft. Since we assume constant altitude during flights, the velocity and position of the aircraft can be expressed in two dimensions. To simplify the derivations of the trajectory, we define the velocity, as well as the position, in the complex plane. Hence

$$\begin{aligned} \dot{c}(t) = v(t) &= v_x(t) + \mathrm{i}v_y(t) \\ c(t) &= x(t) + \mathrm{i}y(t) \end{aligned}$$

Using the complex form, we can then express the velocity using Euler's Equation as

$$v(t) = V\mathrm{e}^{\mathrm{i}\psi(t)} \tag{3}$$

where $\psi(t)$ can be found by integrating (1). The full derivation of the kinematic model can be found in our technical report [16]. The following results are obtained by integrating the yaw rate and velocity. The position of the aircraft flying with constant roll rate can be expressed using the complex Incomplete Beta Function defined as the following:

$$B(x;a,b) = \int_0^x y^{a-1}(1-y)^{b-1}dy \tag{4}$$

where $B : \mathbb{R} \times \mathbb{C} \times \mathbb{C} \mapsto \mathbb{C}$. By using the Incomplete Beta Function, we are able to describe every point on the trajectory as a function of the roll angle of the aircraft:

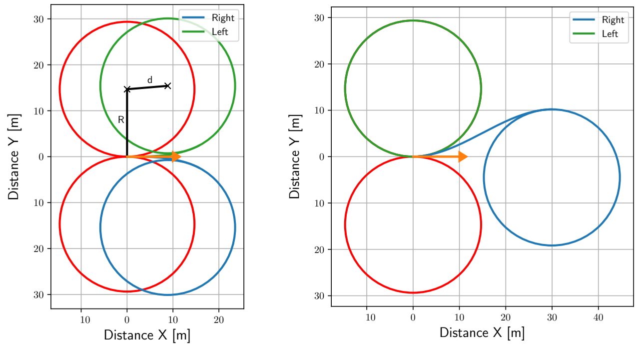
$$\beta(\phi) = \operatorname{sign}(\phi)\frac{V}{2\dot{\phi}_c}\left[B(1;a,b) - B(\cos^2\phi;a,b)\right] \tag{5}$$

where $\operatorname{sign}()$ is the sign function that returns the sign of its argument. The parameters a and b for the Incomplete Beta Function are defined as follows:

$$a = \frac{1}{2} + \mathrm{i}\frac{g}{2V\dot{\phi}_c}, \quad b = \frac{1}{2}$$

Additionally, the yaw angle of the aircraft at each roll angle can be calculated using

$$\psi(t) = \frac{g}{V\dot{\phi}_c}\ln\cos\phi(t) \tag{6}$$



(a) $\phi_0 = 0°$ (b) $\phi_0 = -45°$

Fig. 1: Contrast between Dubin's Path (red) and Beta-Trajectory (green and blue, left and right turn, respectively); Orange arrow indicates initial aircraft position and yaw angle. Data is taken from the Pulsar sailplane in Table I.

The equations for the position and yaw angle of the aircraft are relative to a trajectory frame. In this frame, the initial position and yaw angle of the aircraft are the following:

$$c_0 = \operatorname{sign}(\phi_0)\frac{V}{2\dot{\phi}_c}\left[B(1;a,b) - B(\cos^2\phi_0;a,b)\right] \tag{7}$$

$$\psi_0 = \frac{g}{V\dot{\phi}_c}\ln\cos\phi_0 \tag{8}$$

The full trajectory of an aircraft with a constrained roll rate as well as a constrained maximum roll angle ϕ_{max} can be described as a disjunction of the Beta-Curve B and an orbit $\mathcal{O}$. We define

$$B(\phi_0,\phi_{max}) = \{\beta(\phi) \mid \forall\phi \in [\phi_0,\phi_{max}]\} \tag{9}$$

depending on the initial and maximum roll angle. The orbit

$$\mathcal{O}(c_{center},R) = \{c \in \mathbb{C} \mid \ \|c - c_{center}\| = R\} \tag{10}$$

is defined by its radius R and center point c_{center}. The radius is calculated from the curvature in (1) as

$$R = \left\|\frac{V^2}{g\tan\phi}\right\| \tag{11}$$

The center point is geometrically calculated by adding R perpendicularly to the last point of the Beta-Curve in the roll direction, thus

$$c_{center}(\phi) = \beta(\phi) - \mathrm{i}\operatorname{sign}(\phi)R\mathrm{e}^{\mathrm{i}\psi(\phi)} \tag{12}$$

Consequently, the full Beta-Trajectory is expressed through the disjunction of the Beta-Curve and orbit as

$$T(\phi_0,\phi_{max}) = \mathcal{O}(c_{center}(\phi_{max}),R) \cup B(\phi_0,\phi_{max}) \tag{13}$$

Two example trajectories can be seen in Figure 1 using Pulsar sailplane data from Table I.

B. Frames of Reference

The definition of the Beta-Trajectory in (13) is based on the usage of different frames of reference. To apply the description of the trajectory to a geo-fencing context, a transformation needs to be established. In this work, we describe three frames of reference, illustrated in Figure 2. First is the Trajectory-Frame, described as c^T for position and ψ^T for yaw angle where each point on the trajectory can be described using (5). This frame is centered around the point at which the roll angle of the aircraft is zero, hence, $\beta^T(\phi = 0) = 0 + \mathrm{i}0$, as well as $\psi^T(\phi = 0) = 0$.

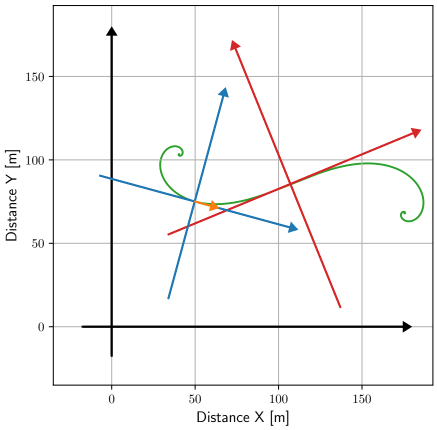


Fig. 2: Frames of reference: Red is the Trajectory-Frame, centered at $\beta(\phi = 0)$; Blue is the Aircraft-Frame, centered at the aircraft position c_0^E; Black is the Earth-Frame, which defines global positions (GPS); Green is an example Beta-Curve $B(-\frac{\pi}{2},\frac{\pi}{2})$ with positive roll rate.

The second frame is the Earth-Frame, described by c^E and ψ^E and based on the GPS coordinates. In this work, the Earth-Frame is the UTM-Frame in East-North-Up (ENU) notation where the zero yaw angle is in the east direction. The third frame is the Aircraft-Frame, described by c^A and ψ^A and centered around the current aircraft position where the current aircraft yaw angle is defined as zero.

To transform from the Trajectory-Frame to the Earth-Frame, the Aircraft-Frame serves as an intermediary for the conversion. With respect to the Earth-Frame, the Aircraft-Frame is offset by the current aircraft position c_0^E and rotated by the current aircraft yaw angle ψ_0^E. Additionally, the Aircraft-Frame is defined in the Trajectory-Frame with the offset c_0^T from (7) and the rotation ψ_0^T from (8). The transformation between the frames can be described in the complex plane. The transformation from the Trajectory-Frame to Aircraft-Frame can be expressed as

$$c^A = \mathrm{e}^{-\mathrm{i}\psi_0^T}(c^T - c_0^T) \tag{14}$$

$$\psi^A = \psi^T - \psi_0^T \tag{15}$$

Similarly the transformation from Earth-Frame to Aircraft-Frame is

$$c^A = \mathrm{e}^{-\mathrm{i}\psi_0^E}(c^E - c_0^E) \tag{16}$$

$$\psi^A = \psi^E - \psi_0^E \tag{17}$$

Combining (14) and (16) as well as (15) and (17) yields

$$c^E = \mathrm{e}^{\mathrm{i}\psi_0^E}\mathrm{e}^{-\mathrm{i}\psi_0^T}(c^T - c_0^T) + c_0^E \tag{18}$$

$$= \mathrm{e}^{\mathrm{i}(\psi_0^E - \psi_0^T)}(c^T - c_0^T) + c_0^E \tag{19}$$

$$\psi^E = \psi^T - \psi_0^T + \psi_0^E \tag{20}$$

as the transformation from the Trajectory-Frame to the Earth-Frame.

For simplification of the following equations, the frame notation is mostly avoided. An equation without frame indication assumes all the arguments are in the same frame. Since the frames represent linear and orthonormal transformations, an equation holding in one frame will hold in other frames.

C. Comparison to Other Approaches

To illustrate the necessity of the kinematic model, we compare it to the Dubin's-Curve model in [2], which is primarily used in the literature. For Dubin's-Curves, only the curvature is constrained, but not the change in curvature determined by the maximum roll rate. As seen in Figure 1, the trajectory deviates from the orbit predicted by the Dubin's-Curve. We define the percent deviation as the ratio between the distance of the center points d and the radius R. The radius R is calculated using (11) and the distance d is calculated using the two center points:

$$d = \|c_{center}(\phi_{max}) - c_{dubin}\| \tag{21}$$

Calculation for c_{dubin} is similar to (12) except that it is set next to the initial position. Hence

$$c_{dubin} = \beta(\phi_0) - \mathrm{i}\,\mathrm{sign}(\dot{\phi}_c) R \mathrm{e}^{\mathrm{i}\psi(\phi_0)} \tag{22}$$

Using the above equations, the deviation values in Table I are calculated.

The authors in [11] linearize the $\tan()$ in (1) to yield

$$\dot{\psi}(t) \approx -\frac{g}{V}\phi(t) \tag{23}$$

which they call Continuous-Curvature Convected Dubins-Curve (CCC-Dubin). This results in a representation of the position on the trajectory using Fresnel integrals, which can only be solved numerically, similar to the Beta Incomplete Function. The linearization introduces deviations at roll angles greater than 30°, which we show in Section IV.

III. GEO-FENCING

We define the geo-fence F as a convex[1] polygon described by N fence segments where each segment F_k is defined as

$$F_k = \{c \in \mathbb{C} \mid \mathrm{Re}\{\overline{n}_k c\} = b_k\} \tag{24}$$

where n_k is the unit normal vector of the line pointing into the polygon and b_k is the offset. $\overline{n}_k$ is the complex conjugate of n_k and $\overline{n}_k c$ represents the inner product of n_k and c in the complex plane. Using the definitions, a *safe* area with respect to F_k can be expressed as

$$S_k = \{c \in \mathbb{C} \mid \mathrm{Re}\{\overline{n}_k c\} > b_k\} \tag{25}$$

which describes the *safe* half-space in $\mathbb{C}$. We add a slack s as a safety margin to account for factors such as the wind, resulting in

$$S_k(s) = \{c \in \mathbb{C} \mid \mathrm{Re}\{\overline{n}_k c\} > b_k + s\} \tag{26}$$

The safe area of the full geo-fence F is then described by

$$S(s) = \bigcap_{\forall k} S_k(s) \tag{27}$$

A. Safety of Beta-Trajectory

We define the Beta-Trajectory T as *safe* if it satisfies

$$T(\phi_0, \phi_{max}) \subset S(s) \tag{28}$$

For the trajectory to fully lie in the safe area, both the orbit and the Beta-Curve need to lie in the safe area. For the orbit, the condition can be written as

$$\mathcal{O}(c_{center}(\phi_{max}), R) \subset S(s) \tag{29}$$

and trivially simplified to

$$c_{center}(\phi_{max}) \in S(s + R) \tag{30}$$

which can be immediately verified.

To evaluate

$$B(\phi_0, \phi_{max}) \subset S(s) \tag{31}$$

it is necessary to check each fence segment F_k individually. The Beta-Curve lies fully in $S_k(s)$ if its outer most point, a critical point, in the direction of a fence segment F_k lies in $S_k(s)$. This critical point is found through the following minimizer:

$$\phi_{crit,k} = \underset{\phi \in [\phi_0, \phi_{max}]}{\arg\min}\ \mathrm{Re}\{\overline{n}_k \beta(\phi)\} \tag{32}$$

This equation could be solved analytically. However, there is an intuitive solution simplifying the problem. Since $B(\phi_0, \phi_{max})$ is a smooth function describing the aircraft trajectory heading towards and away from the fence, potential critical points on the trajectory are the points where the aircraft is flying parallel to the fence. The start and end point of $B(\phi_0, \phi_{max})$, i.e. the current position and the start of the orbit can be evaluated individually.

The direction of the fence segment F_k is defined by its normal vector n_k as

$$\psi_k(l) = \arg\{\mathrm{i} n_k\} + l\pi, \quad l \in \mathbb{Z} \tag{33}$$

meaning that the direction is the 90-degree rotated normal vector and all of its rotationally coinciding vectors. Those coinciding vectors are significant since the Beta-Trajectory has the form of a spiral and, thus, can have multiple parallels. In the Trajectory-Frame, we can solve for the corresponding roll angle by inverting (6). The inversion yields

$$\phi = \pm\cos^{-1}\left(\mathrm{e}^{\frac{\dot{\phi}_c V}{g}\psi^T}\right) := \lambda(\psi^T) \tag{34}$$

Consequently, the critical roll angles on the $B(\phi_0, \phi_{max})$ with respect to fence segment F_k are

$$\Phi_k = \left\{\phi = \lambda(\psi_k^T(l)) \mid \forall l \in \mathbb{Z},\, \phi \in [\phi_0, \phi_{max}]\right\} \tag{35}$$

where the global minimum of (32) satisfies $\phi_{crit,k} \in \Phi_k$. Note that if $\lambda(\psi_k^T(l)) > \phi_{max}$, so is $\lambda(\psi_k^T(l+1))$, simplifying the search for critical roll angles. We can write that

$$B(\phi_0, \phi_{max}) \subset S_k(s) \tag{36}$$

$$\text{iff } \beta(\phi) \in S_k(s), \quad \forall \phi \in \Phi_k \tag{37}$$

---

[1] For concave polygons this approach can be used as well but requires additional steps.

To conclude, the trajectory $T(\phi_0, \phi_{max})$ is *safe* if

$$\beta(\phi) \in S_k(s), \forall \phi \in \Phi_k, \forall k \in \{1, .., N\}$$
$$\wedge$$
$$c_{center}(\phi_{max}) \in S(s + R)$$

This relation shows that for the geo-fencing algorithm only a few distinct points on the predicted trajectory have to be evaluated.

B. Implementation

Algorithm 1: Evaluate Safety

```
Input: $T$
Output: Safety of $T$
Data: aircraftState, $F$
orbitCenter = getOrbitCenter($T$);
forall $F_k \in F$ do
    if orbitCenter $\notin S_k(s+R)$ then
        return unsafe;
    $\Phi$ = getCriticalRolls($T, F_k$);
    for $\phi \in \Phi$ do
        if $\beta(\phi) \notin S_k(s)$ then
            return unsafe;
return safe;
```

The following geo-fencing algorithm is used to initiate an evasive maneuver if the aircraft's current position, attitude, and velocity indicate that a violation of the geo-fence is imminent. To determine if a violation is imminent, we consider two evasive maneuvers: turning fully to the left or right with the constant roll rate up to the maximum roll angle. We say that the aircraft will inevitably break the fence if the resulting trajectories of the evasive maneuvers violate the geo-fence.

At every time step, both trajectories are generated and evaluated. Algorithm 1 implements the derivation in the beginning of this section. It returns *unsafe* if a given trajectory violates any of the fence segments and *safe* otherwise. If both trajectories are classified as *unsafe*, an evasive maneuver is executed in the direction that was last classified as *safe*. The evasive maneuver ends when the aircraft is flying away from the violated fence segment.

The implementation of the algorithms in uavAP autopilot makes use of an overriding framework that enables modules to directly override the targets of the controller. When initiating an evasive maneuver, the geo-fencing module overrides the controller roll target to the maximum roll angle in the corresponding direction. To generate the trajectories and evaluate the β function in (5), the Arb library [17] is used. The Arb library uses ball arithmetic to solve real and complex functions such as the incomplete beta function.

IV. EVALUATION

In order to evaluate the proposed kinematic model and the geo-fencing algorithm, we conducted evaluations in simulations as well as in real flights. In this section, we describe the experimental setup, followed by an evaluation and comparison of the Beta-Trajectory, and concluding with an experimental geo-fenced flight.

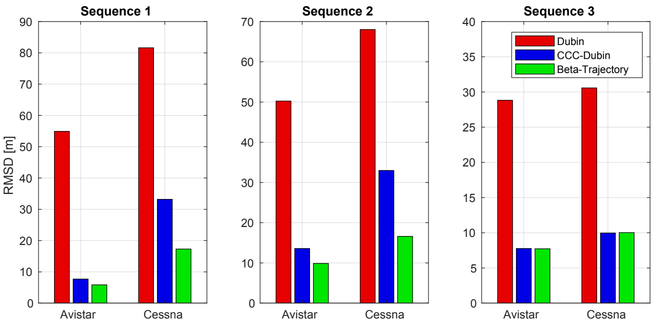


Fig. 3: Root-mean-square deviation of the trajectory predictions of the three approaches for three 30 second maneuvering sequences, comparing in two flight simulators, Trainer in FS One® and Cessna in X-Plane® 11; note the different scale of the y-axis.

A. Experimental Setup

To evaluate the Beta-Trajectory, we compare it to the Dubin's-Curve in [9] as well as the Continuous-Curvature Convected Dubin's-Curve (CCC-Dubin) in [11]. The uavAP autopilot is instructed to fly sequences of roll maneuvers in two different flight simulators, namely the high-fidelity FS One® Flight Simulator ([18], [19]) as well as the X-Plane® 11 ([20]). The aircraft used are a Trainer in FS One® similar to the Avistar Elite and the Cessna 172 in X-Plane®11 similar to the Cessna 177. The simulators are connected through the uavEE emulation environment described in [21].

The geo-fencing algorithm is evaluated by flying an aircraft inside a defined geo-fence. The hardware used for the actual flight is composed of an aircraft and computational hardware. A fixed-wing trainer-type radio control aircraft, the Great Planes Avistar Elite built for previous avionics development [21]–[24], is used for the evaluation. The Avistar has a 1.59 m wingspan and a mass of 3.92 kg; it has the following control surfaces: two ailerons (roll), two flaps, one elevator (pitch), and one rudder (yaw). The specifications of the aircraft can be found in [23]. The aircraft was instrumented with an Al Volo FC+DAQ 400 Hz flight computer and data acquisition system [25], which integrates the open-source uavAP autopilot [13]. The uavAP autopilot is based on a modular and configurable framework that allows for easy integration of different planning and control algorithms. For detailed information about uavAP, the interested reader is directed to the GitHub page[2].

B. Trajectory Prediction

To evaluate the prediction accuracy of the Beta-Trajectory, the autopilot is instructed to fly a sequence of roll maneuvers, alternating from right to left, for 30 seconds in total. The

---

[2] https://github.com/theilem/uavAP.git

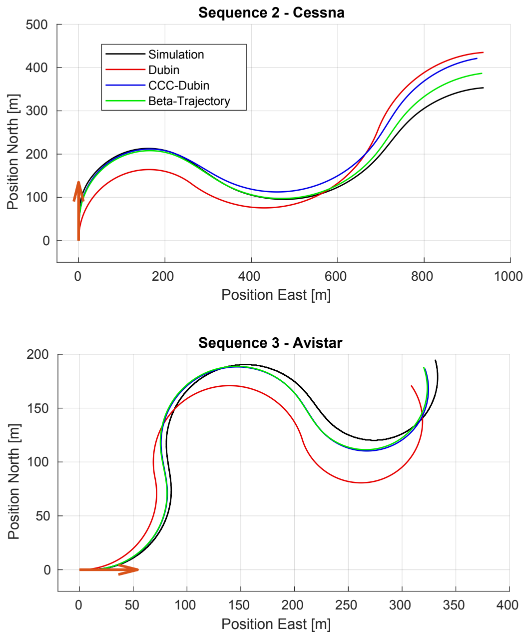


Fig. 4: Two simulated roll maneuver sequences recorded and compared to the three prediction approaches; the orange arrow shows the initial position and heading.

trajectory is recorded and compared to a prediction using the three approaches. The predictions are based on the roll sequences, the aircraft velocity, and the estimated roll rate. The performance of the respective approaches is evaluated based on the deviation to the simulated trajectory. The deviation is calculated as the root-mean-squared deviation/distance (RMSD) of the trajectories. Figure 3 shows the deviation for three different sequences executed with the Avistar and the Cessna. While sequence 1 and sequence 2 used roll targets from the whole spectrum of roll angles (-45° to 45°), sequence 3 is constrained to angles from only -30° to 30°, the linearization limits of the CCC-Dubin's curve. This is to show that the deviation in CCC-Dubin's curves arise from the linearization of the $\tan()$. It can be seen that both the CCC-Dubin approach as well as the Beta-Trajectory outperform the Dubin's approach in the sequences. In sequences 1 and 2, the Beta-Trajectory performs better than the CCC-Dubin approach while they show equal performance in sequence 3.

Figure 4 shows two examples from the sequences, first the sequence 2 using the Cessna and second the sequence 3 using the avistar. Due to the low roll angles in sequence 3, the CCC-Dubin and Beta-Trajectory give equal predictions. In sequence 2, however, a significant drift towards the end of the trajectory can be observed in the CCC-Dubin prediction. The Dubin approach deviates immediately and demonstrates that it is not suitable for short horizon predictions.

C. Geo-Fencing

Since it predicts the simulator path with high accuracy, the Beta-Trajectory is incorporated into the geo-fencing algorithm and deployed it onto a real aircraft. The autopilot on the aircraft was instructed to fly out of the designated area defined by the geo-fence. The higher level geo-fencing algorithm observes the flight and intervenes the current flight path when necessary. The result can be seen in Figure 5. The slack in Equation (26) was set low to emphasize the effect. The aircraft is kept inside the geo-fence with two prominent deviations. The first deviation is reflected by evasive maneuver 4 where the aircraft slightly overshoots the fence. The second deviation occurs in evasive maneuver 2, which has a greater safety margin than the others. These two deviations are attributed to the effects of wind. Since the initial assumption presumes no wind, the deviations due to winds and gusts are expected. However, these slight deviations show the algorithm's ability to decently perform in windy situations despite a low value for the slack. Future work regarding wind integration will improve the results even further.

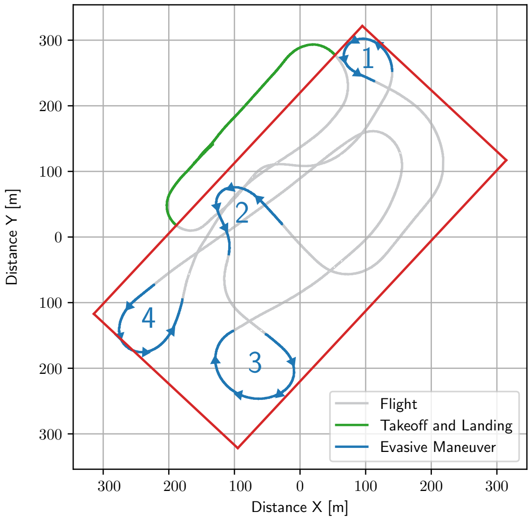


Fig. 5: Real flight path of the Avistar deployed with the uavAP autopilot and the geo-fencing algorithm; Red shows the geo-fence. Blue shows the evasive maneuvers labeled from 1-4. Slack value is 5 meters and velocity is 20 m/s.

V. Conclusion and Future Work

In this work, we show the applicability and the impact of the Beta-Trajectory as a new kinematic model for fixed-wing aircraft. The trajectory is derived from the maximum roll angle as well as the maximum roll rate constraints. Using the newly derived kinematic model, we develop a geo-fencing algorithm that aims to keep the aircraft in a designated area. The theoretical derivation of the kinematic model and the algorithms are validated in high-fidelity simulations using the uavEE emulation environment as well as in real flights. The source code related to this work is available in the open-source autopilot uavAP.

For future work, the Beta-Trajectory can be used for path and trajectory planning algorithms. The model can be updated to incorporate a change of altitude which would affect the relationship between the roll angle and yaw rate. In order to make use of instrumentation that estimates wind speed and direction, wind effects need to be added to the geo-fencing algorithm. As mentioned previously, wind effects can be incorporated as a superposition to the Beta-Trajectory. The geo-fencing algorithm also needs to be adapted to calculate critical roll angles based on the course angle of the aircraft, which represents the direction of the total aircraft velocity vector including the wind effects.

ACKNOWLEDGMENTS

The material presented in this paper is based upon work supported by the National Science Foundation (NSF) under grant number CNS-1646383. Marco Caccamo was also supported by an Alexander von Humboldt Professorship endowed by the German Federal Ministry of Education and Research. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the NSF.

REFERENCES

[1] L. E. Dubins, "On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents," *American Journal of Mathematics*, vol. 79, no. 3, pp. 497–516, 1957.

[2] I. Lugo-Crdenas, G. Flores, S. Salazar, and R. Lozano, "Dubins path generation for a fixed wing uav," in *2014 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 339–346, May 2014.

[3] M. N. Stevens and E. M. Atkins, "Multi-mode guidance for an independent multicopter geofencing system," in *16th AIAA Aviation Technology, Integration, and Operations Conference*, AIAA Paper No. 2016-3150, AIAA AVIATION Forum, Jun. 2016.

[4] S. Zhang, D. Wei, M. Q. Huynh, J. X. Quek, X. Ma, and L. Xie, "Model predictive control based dynamic geofence system for unmanned aerial vehicles," in *AIAA Paper No. 2017-0675, AIAA Infotech @ Aerospace*, Jan. 2017.

[5] M. N. Stevens, B. Coloe, and E. M. Atkins, "Platform-independent geofencing for low altitude uas operations," in *AIAA Paper No. 2015-3329, 15th AIAA Aviation Technology, Integration, and Operations Conference, AIAA AVIATION Forum*, Jun. 2015.

[6] H. T. Dinh, M. H. C. Torres, and T. Holvoet, "Dancing uavs: Using linear programming to model movement behavior with safety requirements," in *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 326–335, June 2017.

[7] E. T. Dill, S. D. Young, and K. J. Hayhurst, "Safeguard: An assured safety net technology for uas," in *2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC)*, pp. 1–10, Sept 2016.

[8] T. Gurriet and L. Ciarletta, "Towards a generic and modular geofencing strategy for civilian uavs," in *2016 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 540–549, June 2016.

[9] M. Shanmugavel, A. Tsourdos, B. White, and R. bikowski, "Cooperative path planning of multiple uavs using dubins paths with clothoid arcs," *Control Engineering Practice*, vol. 18, no. 9, pp. 1084 – 1092, 2010.

[10] A. J. Bateman, W. Gressick, and N. Gandhi, "Application of run-time assurance architecture to robust geofencing of suas," in *AIAA Paper No. 2018-1985, AIAA Infotech@ Aerospace*, Jan. 2018.

[11] L. Techy, C. A. Woolsey, and K. A. Morgansen, "Planar path planning for flight vehicles in wind with turn rate and acceleration bounds," in *2010 IEEE International Conference on Robotics and Automation*, pp. 3240–3245, IEEE, 2010.

[12] O. D. Dantsker, M. Theile, M. Caccamo, and R. Mancuso, "Design, development, and initial testing of a computationally-intensive, long-endurance solar-powered," in *AIAA Paper No. 2018-4217, AIAA Applied Aerodynamics Conference, Atlanta, Georgia*, Jun. 2018.

[13] Real Time and Embedded System Laboratory, University of Illinois at Urbana-Champaign, "Solar-Powered Long-Endurance UAV for Real-Time Onboard Data Processing." http://rtsl-edge.cs.illinois.edu/UAV/, Accessed Sep. 2018.

[14] D. L. Kohlman, "Flight test data for a cessna cardinal. [steady state performance and fixed stick dynamic stability characteristics]," *NASA Contractor Report CR-2337*, Jan 1 1974.

[15] D. Geleyns, "Alternative flight control in civil aviation," Master's thesis, Delft University of Technology,, the Netherlands, 2016.

[16] M. Theile and S. Yu, "Kinematic Model for Fixed-Wing Aircraft with Constrained Roll-Rate," tech. rep., University of Illinois at Urbana-Champaign, Department of Computer Science, Sep. 2018.

[17] Fredrik Johansson, "Arb - a C library for arbitrary-precision ball arithmetic." http://arblib.org/, Accessed Sep. 2018.

[18] M. S. Selig, "Real-time flight simulation of highly maneuverable unmanned aerial vehicles," *Journal of Aircraft*, vol. 51, pp. 1705–1725, Nov.-Dec. 2014.

[19] M. Selig, "Modeling propeller aerodynamics and slipstream effects on small uavs in realtime," in *AIAA Paper No. 2010-7638, AIAA Atmospheric Flight Mechanics Conference, Toronto, Ontario, Canada*, Aug. 2010.

[20] Laminar Research, "X-Plane 11." http://www.x-plane.com/, Accessed Mar. 2019.

[21] M. Theile, O. D. Dantsker, R. Nai, and M. Caccamo, "uavee: A modular, power-aware emulation environment for rapid prototyping and testing of uavs," in *IEEE International Conference on Embedded and Real-Time Computing Systems and Applications, Hakodate, Japan*, Aug. 2018.

[22] R. Mancuso, O. D. Dantsker, M. Caccamo, and M. S. Selig, "A low-power architecture for high frequency sensor acquisition in many-DOF UAVs," in *ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS), Berlin, Germany, Apr. 2014*, pp. 103–114, 2014.

[23] O. D. Dantsker, R. Mancuso, M. S. Selig, and M. Caccamo, "High-frequency sensor data acquisition system (sdac) for flight control and aerodynamic data collection research on small to mid-sized uavs," in *AIAA Paper No. 2014-2565, AIAA Applied Aerodynamics Conference, Atlanta, Georgia*, Jun. 2014.

[24] O. D. Dantsker, M. Theile, and M. Caccamo, "A high-fidelity, low-order propulsion power model for fixed-wing electric unmanned aircraft," in *AIAA Paper No. 2018-5009, AIAA/IEEE Electric Aircraft Technologies Symposium, Cincinnati, Ohio*, Jul. 2018.

[25] Al Volo LLC, "Al Volo: Flight Systems." http://www.alvolo.us, Accessed Sep. 2018.