

Simple Trajectory Smoothing for UAV Reference Path Planning Based on Decoupling, Spatial Modeling and Linear Programming

Mogens Plessen*

Abstract—A method for trajectory smoothing for UAV reference path planning is presented. It is derived based on the dynamics of a Dubins airplane model, and involves a decoupling step, spatial modeling and linear programming. The decoupling step enables algebraic control laws for flight-path angle and speed control. Only for roll angle control an optimization step is applied, involving the solution of a small linear program. Two variations are discussed. They differ by reference centerline tracking and the introduction of a path shaping constraint. The benefit of natural dimensionality reduction for spatial modeling is discussed. The simplicity of the overall method is highlighted. An extension to aerobatic flight is outlined, which comes at the cost of a model approximation, however at the gain of maintaining the general model structure. An extension of the method to tractor path planning along 3D terrain is discussed. The method is validated in simulations.

Index Terms—UAV path planning, trajectory smoothing, Dubins vehicle, spatial modeling, linear programming.

I. INTRODUCTION

In 2D, a Dubins path [1, 2] connects a start and end pose (position and heading) along the shortest possible path. This path is composed of 2 arcs of minimum turning radius of the vehicle and a straight-line segment. In 3D, for the equivalent problem path planning for Uncrewed Aerial Vehicles (UAVs) is more complex. Depending on the start and end pose there may be an infinite number of paths with minimum pathlength.

Connecting a start and end pose along the shortest possible path is useful in many scenarios. It comes, however, at two costs. First, motion is constrained to either a maximal sharp left turn, a maximal sharp right turn or straight driving. Second, the simplifying assumption is made that paths with instantaneous change from a maximal sharp left turn to a maximal sharp right turn and vice versa are possible.

In many scenarios paths are desired that do not only operate at steering limits and instead may satisfy other constraints. The development of a method for this purpose provided the motivation for this article. Starting from the *Dubins airplane model* proposed in [3], an extensive search of all its citing work revealed a research gap for an UAV reference path planning technique that exploits the special structure of the Dubins airplane model for controls decoupling, spatial modeling and linear programming. This is the contribution of this paper.

A literature review is provided. First, the term ‘Dubins airplane model’ is not unique. For example, the Dubins airplane models in [3–5] are all different. The latter reference relates heading angle to roll angle. Its model includes 3D position coordinates as well as airspeed, flight-path, heading and roll angle. It is therefore our preferred model.

Starting from a chosen Dubins airplane model a plethora of model-based control algorithms can be applied [6]. This paper focuses on an optimization-based approach, which can account for constraints in a structured manner [7]. For this purpose, the simplest class of convex optimization problems is linear programming.

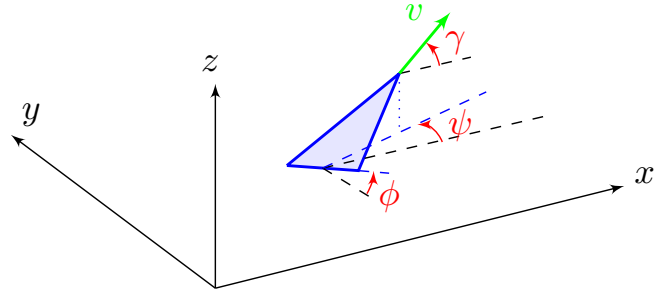


Fig. 1. Illustration of the Dubins airplane model. For simplicity it is depicted as a triangle. States and control variables are annotated. Different colors are used to differentiate positions, angles and velocity. See (1) for interpretation.

In [8] a method for 3D Dubins path planning with bounded curvature and pitch angle is presented. The problem is decoupled into horizontal and vertical problems, before both are solved separately using 2D Dubins paths. The focus is exclusively on Dubins paths between start and end pose. The method is not generalizable to trajectory smoothing tasks. Their Dubins airplane model does not include roll angle.

Transforming time-dependent to spatially-dependent dynamics, typically in combination with respect to a reference trajectory, is a technique for problem reformulations [9, 10]. It was found to often facilitate problem solutions to account for spatial constraints, and is also the approach taken here.

In 2D for planar path planning, the spatial problem reformulation is often referred to as planning in the *Frenet frame* [11, 12]. An extension to 3D is referred to as the *Frenet-Serret* frame, however, there exist more and modeling is much more complex [13–15].

Within a typical hierarchical framework, the presented method follows after an upstream planning step. For example, given a first high-level routing result [16, 17], a second trajectory smoothing step is required to prepare a reference path plan, before a subsequent third low-level path tracking step follows. Thus, the method can be classified as a trajectory smoothing step in-between routing and low-level tracking.

The remaining paper is organized as follows: problem formulation, problem solution, numerical examples, discussion and the conclusion are described in Sect. I–VI.

II. PROBLEM FORMULATION

The addressed problem can be formulated as follows. Given an initial (rough) path trajectory in 3D, $\{x_i^{\text{ref}}, y_i^{\text{ref}}, z_i^{\text{ref}}\}_{i=0}^{N_{\text{ref}}}$, it is sought to *smooth* this trajectory subject to (i) UAV dynamics according to a Dubins airplane model, and (ii) additional trajectory shaping constraints. A desired result is a smoothed reference path plan that improves upon the initial trajectory.

The initial path trajectory may consist of a sequence of waypoints or an interpolated path connecting multiple waypoints. The path may initially not be feasible with respect to UAV dynamics.

*pmogens@proton.me, Findklein GmbH, Zug, Switzerland

The *Dubins airplane model* used for the remainder shall be described by the following set of differential equations [3],

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} v \cos(\gamma) \cos(\psi) \\ v \cos(\gamma) \sin(\psi) \\ v \sin(\gamma) \\ \frac{g}{v} \tan(\phi) \end{bmatrix}, \quad (1)$$

where (x, y, z) denote the inertial position of the UAV. The remaining variables are airspeed v , yaw (heading) angle ψ , flight-path angle γ , roll angle ϕ , and g the acceleration due to gravity. See Fig. 1 for illustration.

Several comments can be made. First, the Dubins airplane model is a simplified model. It does not account for effects such as aerodynamics, wind, sideslip and transients resulting from low-level control delays. However, for high-level path planning it is well suited [3].

Second, it is noted that there exist a variety of alternative models, that in their respective papers are also referred to as *Dubins airplane model* [4, 5, 18]. Thus, (1) does not represent *the* Dubins airplane model. However, its formulation appears favorable since it involves all 6 variables typical for UAV path planning: 3 inertial position coordinates, yaw, roll and flight-path angle. This serves well for fixed-wing path planning, but in general likewise also for quadrotor path planning [18].

Finally, it is explicitly stressed that the topic of this paper is *reference* path planning. Low-level control (such as, e.g., necessary for weather-dependent wind rejection) is not a topic of this paper. Generating smoothed nominal and deterministic reference path plans, that account for nonholonomic robot dynamics in 3D (Dubins airplane model) and spatial precision constraints is important for high-level mission planning. Agricultural area coverage is an example application [17].

III. PROBLEM SOLUTION

The proposed solution to the problem of Sect. II is centered around the Dubins airplane model (1) and exploiting its special structure. There are 4 state variables, (x, y, z, ψ) , and 3 control variables, (γ, ϕ, v) . Control methods for each of them are derived in the following.

A. Decoupling of dynamics

A first observation with respect to (1) is made. While (x, y) are coupled with respect to yaw angle ψ , this is not the case for z -channel dynamics. This motivates a decoupling.

Therefore, the following hierarchical control approach is taken: (i) z -dynamics are decoupled from the remaining system and treated isolatedly to determine a control law for γ , before (ii) feeding the resulting γ -trajectory as a reference to the remaining 3-state dynamics. These result in

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \bar{v} \cos(\psi) \\ \bar{v} \sin(\psi) \\ \frac{g}{v} \tan(\phi) \end{bmatrix}, \quad (2)$$

where $\bar{v} = v \cos(\bar{\gamma})$, and $\bar{\gamma}$ denotes the control trajectory determined from step (i) after decoupling.

B. Step 1: Control of flight-path angle γ

A control law for flight-path angle γ is derived. Integrating the decoupled z -dynamics and discretizing under a zero-order hold assumption, $\gamma(t) = \gamma_k$, $\forall t \in [kT_s, (k+1)T_s]$, and

sampling time T_s , one obtains $z_{k+1} = z_k + T_s v_k \sin(\gamma_k)$. Abbreviating sampling space,

$$D_{s,k} = T_s v_k, \quad (3)$$

accounting for an appropriately interpolated given initial reference trajectory, $\{z_k^{\text{ref}}\}_{k=0}^N$, along $N+1$ points, and after inversion, the basic control law becomes

$$\gamma_k^{\text{ref}} = \arcsin\left(\frac{z_{k+1}^{\text{ref}} - z_k^{\text{ref}}}{D_{s,k}^{\text{ref}}}\right). \quad (4)$$

Two refinement steps are additionally applied. First, rate and limit constraints are enforced. Thus, $\tilde{\gamma}_0 = \gamma_0^{\text{ref}}$ and for subsequence indices $k > 0$,

$$l_{\gamma,k} = \max(\gamma_{\min}, \tilde{\gamma}_{k-1} - \Delta\gamma_{\min}), \quad (5)$$

$$u_{\gamma,k} = \min(\gamma_{\max}, \tilde{\gamma}_{k-1} + \Delta\gamma_{\max}), \quad (6)$$

$$\tilde{\gamma}_k = \min(\max(\gamma_k^{\text{ref}}, l_{\gamma,k}), u_{\gamma,k}), \quad (7)$$

where $\Delta\gamma_{\min} = D_{s,k} |\dot{\gamma}_{\min}| / v_k$, $\Delta\gamma_{\max} = D_{s,k} \dot{\gamma}_{\max} / v_k$, and $\dot{\gamma}_{\min}$, $\dot{\gamma}_{\max}$, $\gamma_{\min}$ and $\gamma_{\max}$ describe rate and limit bounds, respectively. Second, a simple heuristic prediction compensation step is applied,

$$\tilde{\gamma}_k = (\tilde{\gamma}_k + \tilde{\gamma}_{k+1})/2, \quad \forall k = 0, \dots, N-1, \quad (8)$$

which improved reference tracking.

Several comments can be made. First, the control law is defined *spatially* along the reference trajectory. This resulted naturally, as a consequence of the specific dynamics, $\dot{z} = v \sin(\gamma)$, and above zero-order hold assumptions used for derivation.

Second, it is noted that rate and limit bounds are in general operating point varying, e.g., as a function of altitude. This can directly be accounted for in above formulation.

Finally, it is noted that the resulting control trajectory $\{\tilde{\gamma}_k\}_{k=0}^{N-1}$ according to above formulation is clearly strongly depending on the reference trajectory, $\{z_k^{\text{ref}}\}_{k=0}^N$. However, this is a natural result, compatible with the special dynamics structure of (1).

C. Step 2: Control of roll angle ϕ

Control of roll angle is based on the decoupled dynamics of equation (2) with 3 remaining states. Two observations can be made. First, (2) closely resembles the classic kinematic bicycle model for 2D path planning, $[\dot{x}_{\text{car}}, \dot{y}_{\text{car}}, \dot{\psi}_{\text{car}}] = [v_{\text{car}} \cos(\psi_{\text{car}}), v_{\text{car}} \sin(\psi_{\text{car}}), v_{\text{car}} \tan(\delta_{\text{car}})/l_{\text{car}}]$, where δ_{car} models steering angle, the center of gravity of the vehicle is assumed to be located at the rear acles, l_{car} denotes the vehicle's wheelbase, and subindex $\{\cdot\}_{\text{car}}$ is used to clearly distinguish the 2 models. This kinematic model served as the basis for a spatial control law for ground-based path smoothing for an agricultural application [19].

Second, it is noted that the control $\{\tilde{\gamma}_k\}_{k=0}^{N-1}$ according to derivation in Sect. III-B is naturally spatially dependent.

Both observations motivate to derive a spatially varying control law for ϕ . For notation, for an arbitrary variable, $p \in \mathbb{R}$, let time and spatial derivatives be denoted by $\dot{p} = \frac{dp}{dt}$ and $p' = \frac{dp}{ds}$, respectively, where s represents the spatial coordinate along a reference trajectory. Coordinates between the time and spatial domain can be transformed by projecting point masses (such as e.g. the vehicle's center of gravity) to the reference trajectory. Let heading deviation be $\dot{e}_{\psi} = \dot{\psi} - \dot{\psi}_s^{\text{ref}}$ where

ψ_s^{ref} is the reference heading, $\dot{e}_y = \bar{v} \sin(e_\psi)$ denoting lateral deviation, and $\dot{s} = \frac{\rho_s \bar{v} \cos(e_\psi)}{\rho_s - e_y}$ where ρ_s denotes the radius of curvature. With $e'_\psi = \frac{\dot{e}_\psi}{\dot{s}}$ and $e'_y = \frac{\dot{e}_y}{\dot{s}}$, the spatial equivalent to (2) is derived as

$$\begin{bmatrix} e'_\psi \\ e'_y \end{bmatrix} = \begin{bmatrix} \frac{(\rho_s - e_y)g \tan(\phi)}{\rho_s v \bar{v} \cos(e_\psi)} - \psi'_s \\ \frac{(\rho_s - e_y) \tan(e_\psi)}{\rho_s} \end{bmatrix}. \quad (9)$$

A key difference with respect to its ground-based equivalent is the velocity-dependence in the denominator of the e_ψ -dynamics in (9).

Note the dimensionality reduction due to the spatial transformation. In (9) there are only 2 states in comparison to 3 for its time-based equivalent in (2). Dimensionality reduction is an important characteristic and advantage of spatial modeling.

While the control trajectory $\{\bar{\gamma}_k\}_{k=0}^{N-1}$ from Sect. III-B can be seen as a passive local filtering of $\{z_k^{\text{ref}}\}_{k=0}^N$, for the control of ϕ an active optimization-based approach is proposed that determines an optimal solution accounting for the entire horizon of the reference trajectory ahead.

In line with the problem formulation in Sect. II and in reference to [19], two linear programs (LP) are proposed: one for reference centerline tracking, and one to illustrate how above spatial modeling permits to naturally formulate spatial constraints.

After linearization and discretization of (2), the proposed first LP for reference tracking, referred to as LP1 in the following, is:

$$\min_{\{\phi_k\}_{k=0}^{N-1}} \sum_{j=k}^N |e_{y,k} - e_{y,k}^{\text{ref}}| \quad (10a)$$

$$\text{s.t. } D_{s,k} \frac{\dot{\phi}_{\min}}{v_k^{\text{ref}}} \leq \phi_{k+1} - \phi_k \leq D_{s,k} \frac{\dot{\phi}_{\max}}{v_k^{\text{ref}}}, \quad k = 0, \dots, N-2, \quad (10b)$$

$$\phi_{\min} \leq \phi_k \leq \phi_{\max}, \quad k = 0, \dots, N-1, \quad (10c)$$

with $D_{s,k} = s_{k+1} - s_k, \forall k = 0, \dots, N-2$.

The proposed second LP, referred to as LP2 in the following, is:

$$\min_{\{\phi_k\}_{k=0}^{N-1}} \sum_{k=1}^N |e_{y,k} - e_{y,k}^{\text{ref}}| \quad (11a)$$

$$\text{s.t. } e_{y,k} - e_{y,k}^{\text{ref}} \leq 0, \quad k = 1, \dots, N, \quad (11b)$$

$$D_{s,k} \frac{\dot{\phi}_{\min}}{v_k^{\text{ref}}} \leq \phi_{k+1} - \phi_k \leq D_{s,j} \frac{\dot{\phi}_{\max}}{v_k^{\text{ref}}}, \quad k = 0, \dots, N-2, \quad (11c)$$

$$\phi_{\min} \leq \phi_k \leq \phi_{\max}, \quad k = 0, \dots, N-1, \quad (11d)$$

with $D_{s,k} = s_{j+k} - s_k, \forall k = 0, \dots, N-2$. The key novelty with respect to (10) is the introduction of constraint (11b). This enforces the resulting smoothed trajectory to laterally rest to the right-hand side of the reference path within the projected (s, e_y) -plane. Many variations of this constraint are possible, for example, for obstacle avoidance or for other path shaping objectives.

Note that in both (10) and (11), speed v is not treated as a control variable. Instead, it is treated as a reference parameter, $\{v_k^{\text{ref}}\}_{k=0}^N$, that is in general spatially varying along the reference path. This procedure ensures compatibility with

$\{\bar{\gamma}_k\}_{k=0}^{N-1}$ -control generation according to Sect. III-B. Speed control, hierarchically following both the flight-path angle control and the LP-solution, is discussed below in Sect. III-D.

For linearization and in the objective functions, references $e_{\psi,k}^{\text{ref}}$, $e_{y,k}^{\text{ref}}$ and ϕ_k^{ref} are initialized at zero. For the absolute values in (10a) and (11a), N non-negative surrogate variables are introduced and the set of inequality constraints is extended accordingly. For feasibility guarantee, constraints in (11b) are softened to, $e_{y,k} - e_{y,k}^{\text{ref}} \leq \sigma$, by introducing a non-negative slack variable $\sigma \geq 0$ and adding it to the objective function (10a) with a high weight such that $\sum_{k=1}^N |e_{y,k} - e_{y,k}^{\text{ref}}|$ is replaced with $\sum_{k=1}^N |e_{y,k} - e_{y,k}^{\text{ref}}| + 10^{16} \sigma$. Thus, in total there are $n_u = 2N$ and $n_u = 2N+1$ scalar real-valued optimization variables for LP1 and LP2, respectively. This implies very small LPs, favorable for fast LP-solution times and enabling planning over large horizons. One additional benefit of proposed LP-formulations is that they are hyperparameter-free. There are no weighting parameters in the objective functions.

In general, *LP-iterations* or refinement steps can be conducted. These are useful for heuristic reduction of jaggedness [19]. For a first LP the original reference path is used as a reference. After that, the LP-solution itself is used as reference for a second LP-iteration, and so forth. In practice, one refinement step was found to be sufficient.

D. Step 3: Speed control v

The third and remaining control variable is speed v . According to above methodology there are 2 decouplings. First, flight-path angle γ is controlled decoupledly from the other 2 control variables according to Sect. III-B. It is determined as a function of $\{z_k^{\text{ref}}\}_{k=0}^N$, but implicitly also as a function of speed via (3), which determines the spatial interpolation spacing based on which the spatial control law for $\{\bar{\gamma}_k\}_{k=0}^{N-1}$ is derived. Second, $\{\phi_k\}_{k=0}^{N-1}$ is determined according to the LP-method from Sect. III-C, and based on the same spatial interpolation grid.

Now, an upper speed bound can be derived that is spatially varying as a function of curvature along the smoothed path trajectory from the LP-result. Using (1), the radius of curvature can be related to speed as $v = \rho_s \dot{\psi} = \rho_s \frac{g \tan(\phi)}{v}$. The radius of curvature along the smoothed path and maximum roll angle $\phi_{\max}$ is available. Thus, the upper speed bound can be computed as

$$v_{\max}(s) = \sqrt{\rho_s(s)g \tan(\phi_{\max})}, \quad (12)$$

where s represents the spatial coordinate along the smoothed path of length L , i.e., $s \in [0, L]$. This permits to formulate our basic control law for speed as,

$$v(s) = \min(v^{\text{ref}}(s), v_{\max}(s)), \quad \forall s \in [0, L]. \quad (13)$$

It implies following a desired reference velocity, and reducing speed as a function of curvature if necessary. Based on (13), two refinement steps analogous to (5)-(8) are applied to account for rate and limit bounds.

To summarize, the simplicity and compactness of the proposed decoupled approach is highlighted. Both the control of γ according to Sect. III-B and the control of v follow algebraic equations. Only the control of roll angel ϕ involves an optimization step. However, aided by the spatial modeling approach this is only a small-scale LP and with a hyperparameter-free objective function.

E. Extension for aerobatic flight

Above method fails for excessive z -coordinate variation when the flight-path angle γ approaches 90° and a singularity in (1) occurs.

Without changing the fundamental structure, above method can be adapted to aerobatic flight maneuvers. This can be achieved by 2 steps: a coordinate system transformation and an adaptation of rate and limit bounds.

It is explicitly stressed that this adaptation is *approximative* as will be detailed below. However, it permits to extend planning capabilities to loopings and similar aerobatic maneuvers while maintaining the simple methods from Sect. III-B-III-D.

First, a coordinate system transformation is applied,

$$\begin{bmatrix} x_r \\ y_r \\ z_r \end{bmatrix} = \begin{bmatrix} x \\ z \\ -y \end{bmatrix}, \quad (14)$$

which implies a 90° -rotation around the original x -axis, and where the x_r , y_r and z_r axes define the new 3D Cartesian coordinate system.

Second, rate and limit bounds are adapted to mimic *in the new coordinate system the dynamics in the old coordinate system*. Therefore, we assign:

$$\psi_r = \gamma, \quad (15a)$$

$$\gamma_r = -\psi, \quad (15b)$$

and $\phi_r = \phi - \pi/2$ (since above rotation was carried out around the x -axis), and $v_r = v$. The derivative of (15a) yields $g \tan(\phi_r)/v_r = \dot{\gamma}$, from which limit and rate bounds can be derived as

$$\arctan\left(\frac{-|\dot{\gamma}_{\min}|v_{\min}}{g}\right) \leq \phi_r \leq \arctan\left(\frac{\dot{\gamma}_{\max}v_{\max}}{g}\right), \quad (16)$$

$$-\dot{\phi}_{\max} \leq \dot{\phi}_r \leq \dot{\phi}_{\max}, \quad (17)$$

whereby the rate bounds hold since $\phi_r = \phi - \pi/2$. Similarly, the derivative of (15b) yields $\dot{\gamma}_r = -g \tan(\phi)/v$, from which rate bounds can be derived as

$$\frac{g \tan(\phi_{\min})}{v_{\max}} \leq \dot{\gamma}_r \leq \frac{g \tan(\phi_{\max})}{v_{\min}}. \quad (18)$$

Note that γ_r remains unconstrained to permit aerobatic flight such as looping maneuvers.

Lastly, above mimicking step implies a model approximation. Dynamics along the x -axis are exact according to (14) and assignment (15), i.e., $v_r \cos(\gamma_r) \cos(\psi_r) = v \cos(\gamma) \cos(\psi)$. In contrast, the mimicking approximation becomes apparent through the dynamics along the y -axis (and similarly along the z -axis). It holds $v_r \cos(\gamma_r) \sin(\psi_r) \approx v \sin(\gamma)$ only as long as $\cos(\gamma_r) \approx 1$, i.e., for small γ_r -angles.

This implies the model approximation is valid for small γ_r -angles. For planning it encourages (i) to select a dominating plane in the original coordinate system, i.e., either the (x, y) -plane or the (x, z) -plane, (ii) to plan in that plane, before (iii) re-rotating the computed trajectory back to the original coordinate system if needed. To give an example, for a looping the (x, z) -plane would be dominating. In contrast, for a terrain following application the (x, y) -plane is dominating. Determining the dominating plane is a heuristic choice, typically relating variation of the reference path along the critical z -axis to variation within the (x, y) -plane.

F. Extension for tractor path planning along 3D terrain

In Sect. III-C a comment was made how (2) closely resembles the classic kinematic bicycle model for 2D path planning. In fact, the Dubins airplane model (1) can be modified into

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} v \cos(\gamma) \cos(\psi) \\ v \cos(\gamma) \sin(\psi) \\ v \sin(\gamma) \\ \frac{v}{l} \tan(\delta) \end{bmatrix}, \quad (19)$$

to approximate ground-based path planning accounting for topographically varying terrain, where δ indicates the steering angle and l the vehicle's wheelbase.

Crucially and in contrast to UAV path planning, γ is now not a control variable anymore but instead directly determined by the topography of the terrain where the vehicle is driving.

Furthermore, speed v only occurs in the nominators of (19). This is in contrast to UAV-modeling. As a result, the analogous derivation of (9) yields a natural elimination of velocity-dependence. Thus, after spatial transformation, steering control and velocity control are naturally decoupled [19].

Thus, while the method of Sect. III-B-III-D was developed for UAVs, it can be directly adopted for terrain-following 3D path planning applications such as in [20].

IV. NUMERICAL EXPERIMENTS

Results of numerical experiments are visualized in Fig. 2, 3, 4 and 5.

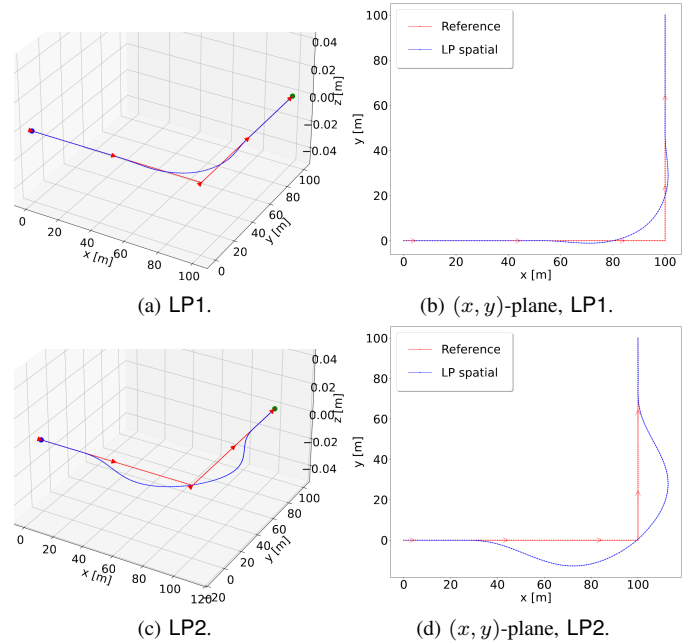


Fig. 2. Illustration of the effects of LP1 and LP2 for an edgy reference smoothing example.

V. DISCUSSION, LIMITATIONS AND OUTLOOK

The *Dubins airplane model* is a simplifying model [3]. Limitations include (i) not accounting for wind influences, (ii) time-delays in inner low-level control loops that regulate flight-path angle, roll angle and speed, and (iii) aerodynamic effects such as, for example, a positive flight-path angle reducing airspeed.

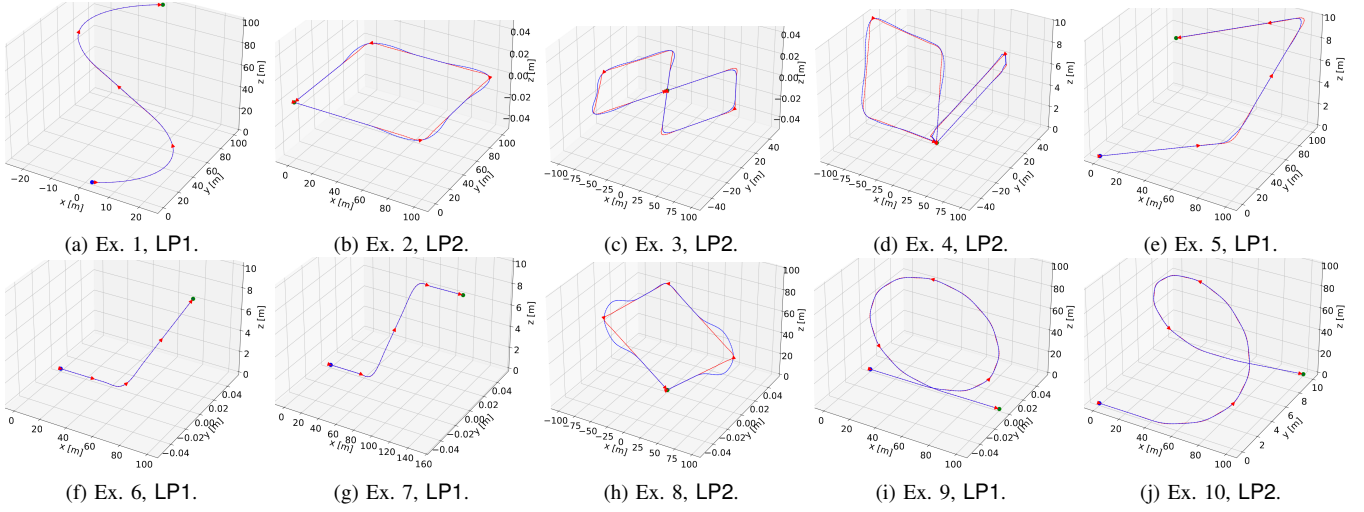
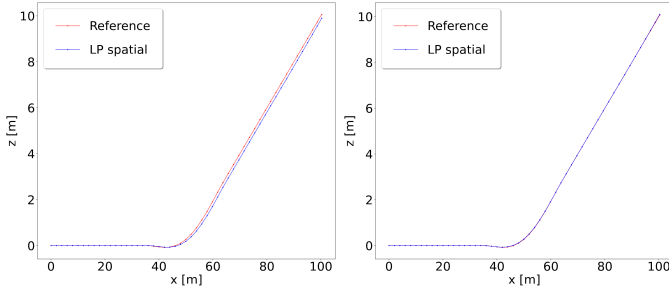


Fig. 3. Illustration of 10 numerical experiments. For Ex. 1 a path generated by the method from [3] served as reference. For the other examples waypoints were defined, then connected by uniform interpolation, before the method from Sect. III-B-III-E was applied. For example, for Ex. 3 there were 9 waypoints.



(a) Ex. 6, tracking if $\tilde{\gamma}_k = \tilde{\gamma}_k$ in (8). (b) Ex. 6, no tracking error for (8).

Fig. 4. Illustration of a detail discussed in Sect. III-B.

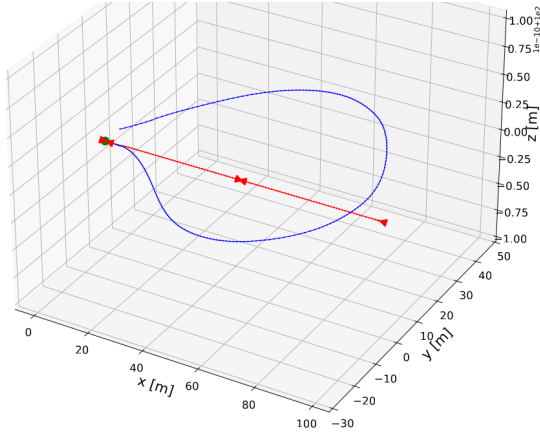


Fig. 5. A very unfavorable reference can still be tracked. See Sect. V.

While objective functions of the LPs in Sect. III-C are hyperparameter free, there are still heuristic choices needed. These are (i) path tangential interpolation spacing, and (ii) in case of aerobatic maneuvers a heuristic to determine the dominating plane and a splitting method to determine when to start planning in the respective coordinate systems.

The title of the paper deliberately includes the phrase trajectory *smoothing*. This is to underline that a reasonable

reference is required as input. Not any arbitrary references can be smoothed. For example, a 3-waypoint reference with $\{(x_i^{\text{ref}}, y_i^{\text{ref}}, z_i^{\text{ref}})\} = \{(0, 0, 100), (100, 0, 100), (0, 0, 100)\}$, that implies an instantaneous 180° -turn, which is infeasible for the nonholonomic dynamics of the Dubins airplane model, cannot directly be smoothed with the presented method. This is because spatial modeling implies a parametrization with respect to a reference trajectory, which makes it vulnerable to singularities [13, 14]. One practical simple solution is to (i) linearize and discretize (1) in the time domain and solve an analogous tracking problem as (10) for above 3-waypoint reference, before (ii) solving e.g. (11) on top to add spatial shaping constraints. This approach is efficient since optimization problems based on discretized time domain dynamics of (1) can be solved robustly (e.g., nilpotent state transition matrices). In contrast, formulating spatial constraints such as in (11) is not obvious for the time domain.

The speed control law (13) encourages following of a reference velocity as long as path curvature permits this. Alternatively, the method from [21] may be adopted for *time-scheduling of the reaching of waypoints* along the path. In that case, in contrast to the method of Sect. III-C, two alternative LPs would be formulated with now 2 controls, $u = [\frac{1}{v}, \phi]$, thereby replacing the speed control method from Sect. III-D.

A perceived benefit of Dubins airplane model-based planning is that 6 variables are involved that are also typical for quadrotor path planning. These are the 3 position coordinates (x, y, z) and 3 angles (ψ, γ, ϕ) . Dynamical quadrotor models typically have a much larger state space. Future work may analyze to what extent the fast (to a large part algebraic) planning methods presented may be used for reference trajectory generations for quadrotors [18]. It is interesting to note that the 4 flat outputs characteristic for *differential flatness*-based algorithms [22] are the exact same as the 4 states of the Dubins airplane model in (1).

Finally, spatial modeling seems well suited for drone formation flights [23, 24]. Setups can be imagined where a leading drone is followed by a cascade of follower drones. For such setups the spatial modeling of Sect. III-C could be extended with longitudinal (adaptive cruise control-like) dynamics [25].

VI. CONCLUSION

A method for trajectory smoothing for UAV reference path planning was presented. It was derived based on the dynamics of a Dubins airplane model, and involved a decoupling step, spatial modeling and linear programming. 3 control variables were considered. Proposed decoupling method resulted in algebraic control laws for flight-path angle and speed control. Only for roll angle control an optimization step was required, involving the solution of a small linear program. Two variations were discussed, differing by reference centerline tracking and the introduction of a path shaping constraint. An extension to aerobatic flight was discussed, which came at the cost of a model approximation, however at the gain of maintaining the general model structure. An extension to tractor path planning along 3D terrain was discussed.

It was found that despite its simplicity proposed method could be applied to a variety of 3D path planning setups.

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] X. Zhou, L. Li, X. Zhang, S. Jin, X. Xu, D. Shi, and H. Wang, "A seventy-year review of dubins robot path planning: basics, developments and opportunities," *Industrial Robot: The International Journal of Robotics Research and Application*, pp. 1–23, 2026.
- [3] T. McLain, R. W. Beard, and M. Owen, "Implementing dubins airplane paths on fixed-wing UAVs," 2014.
- [4] H. Chitsaz and S. M. LaValle, "Time-optimal paths for a dubins airplane," in *IEEE Conference on Decision and Control (CDC)*, 2007, pp. 2379–2384.
- [5] J. Lim, E. Hafner-Aeschbacher, F. Achermann, R. Girod, D. Rohr, N. Lawrance, Y. Bühler, and R. Siegwart, "Autonomous and efficient large-scale snow avalanche monitoring with an unmanned aerial system (uas)," *Natural Hazards and Earth System Sciences*, vol. 26, no. 1, pp. 411–431, 2026.
- [6] P. Kumar, K. Pal, and M. C. Govil, "Comprehensive review of path planning techniques for unmanned aerial vehicles (UAVs)," *ACM Computing Surveys*, vol. 58, no. 3, pp. 1–44, 2025.
- [7] M. Kamel, T. Stastny, K. Alexis, and R. Siegwart, "Model predictive control for trajectory tracking of unmanned aerial vehicles using robot operating system," in *Robot Operating System (ROS) The Complete Reference (Volume 2)*. Springer, 2017, pp. 3–39.
- [8] P. Váňa, A. A. Neto, J. Faigl, and D. G. Macharet, "Minimal 3D dubins path with bounded curvature and pitch angle," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 8497–8503.
- [9] M. Sampei, T. Tamura, T. Itoh, and M. Nakamichi, "Path tracking control of trailer-like mobile robot," in *Proceedings IROS'91: IEEE/RSJ International Workshop on Intelligent Robots and Systems*, 1991, pp. 193–198.
- [10] A. Micaelli and C. Samson, "Trajectory tracking for unicycle-type and two-steering-wheels mobile robots," Ph.D. dissertation, Inria, 1993.
- [11] M. P. Do Carmo, *Differential geometry of curves and surfaces: revised and updated second edition*. Courier Dover Publications, 2016.
- [12] M. Werling, J. Ziegler, S. Kammel, and S. Thrun, "Optimal trajectory generation for dynamic street scenarios in a frenet frame," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2010, pp. 987–993.
- [13] J. Arrizabalaga, Z. Šír, Z. Manchester, and M. Ryll, "A universal formulation for path-parametric planning and control," *arXiv preprint arXiv:2410.04664*, 2024.
- [14] T. Fork and F. Borrelli, "Euclidean and non-euclidean trajectory optimization approaches for quadrotor racing," *arXiv preprint arXiv:2309.07262*, 2023.
- [15] J. C. H. Ramírez and M. Nahon, "A gravity-referenced moving frame for vehicle path following applications in 3D," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4393–4400, 2021.
- [16] K. Yakunin, Y. Kuchin, E. Muhamedijeva, A. Symagulov, and R. I. Mukhamediev, "An algorithm for planning coverage of an area with obstacles with a heterogeneous group of drones using a genetic algorithm and parameterized polygon decomposition," *Drones*, vol. 9, no. 9, p. 658, 2025.
- [17] M. Plessen, "Path planning for spot spraying with UAVs combining tsp and area coverages," *Smart Agricultural Technology*, vol. 11, p. 100965, 2025.
- [18] Y. Lin and S. Saripalli, "Path planning using 3D dubins curve for unmanned aerial vehicles," in *IEEE International Conference on Unmanned Aircraft Systems (ICUAS)*, 2014, pp. 296–304.
- [19] M. Plessen, "Smoothing of headland path edges and headland-to-mainfield lane transitions based on a spatial domain transformation and linear programming," *Biosystems Engineering*, vol. 257, p. 104229, 2025.
- [20] —, "From 2D to 3D terrain-following area coverage path planning," in *IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, 2026, pp. 79–84.
- [21] M. G. Plessen, "Trajectory planning of automated vehicles in tube-like road segments," in *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, 2017, pp. 1–6.
- [22] D. Mellinger and V. Kumar, "Minimum snap trajectory generation and control for quadrotors," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 2520–2525.
- [23] Z. Wang, L. Liu, T. Long, and G. Xu, "Efficient unmanned aerial vehicle formation rendezvous trajectory planning using dubins path and sequential convex programming," *Engineering Optimization*, vol. 51, no. 8, pp. 1412–1429, 2019.
- [24] Z. Chao, S.-L. Zhou, L. Ming, and W.-G. Zhang, "UAV formation flight based on nonlinear model predictive control," *Mathematical Problems in Engineering*, vol. 2012, no. 1, p. 261367, 2012.
- [25] M. G. Plessen, D. Bernardini, H. Esen, and A. Bemporad, "Multi-automated vehicle coordination using decoupled prioritized path planning for multi-lane one- and bi-directional traffic flow control," in *IEEE Conference on Decision and Control (CDC)*, 2016, pp. 1582–1588.