mathematics

Article

Sliding Surface-Based Path Planning for Unmanned Aerial Vehicle Aerobatics

Oleg Cravioto [1], Belem Saldivar [2], Manuel Jiménez-Lizárraga [3,*], Juan Carlos Ávila-Vilchis [1] and Carlos Aguilar-Ibañez [4]

1 Facultad de Ingeniería, Universidad Autónoma del Estado de México, Toluca de Lerdo 50130, Mexico; ocraviotog001@alumno.uaemex.mx (O.C.)
2 Departamento de Control Automático, Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional (CINVESTAV), Ciudad de México 07360, Mexico
3 Facultad de Ciencias Físico Matemáticas, Universidad Autónoma de Nuevo Léon, Nuevo León 66451, Mexico
4 Centro de Investigación en Computación, Instituto Politécnico Nacional, Ciudad de México 07738, Mexico; carlosaguilari@cic.ipn.mx
* Correspondence: manuel.jimenezlzr@uanl.edu.mx; Tel.: +52-81-8329-4030 (ext. 6136)

Abstract: This paper exploits the concept of nonlinear sliding surfaces to be used as a basis in the development of aerial path planning projects involving aerobatic three-dimensional path curves in the presence of disturbances. This approach can be used for any kind of unmanned aerial vehicle aimed at performing aerobatic maneuvers. Each maneuver is associated with a nonlinear surface on which an aerial vehicle could be driven to slide. The surface design exploits the properties of Viviani's curve and the Hopf bifurcation. A vector form of the super twisting algorithm steers the vehicle to the prescribed surfaces. A suitable switching control law is proposed to shift between surfaces at different time instants. A practical stability analysis that involves the descriptor approach allows for determining the controller gains. Numerical simulations are developed to illustrate the accomplishment of the suggested aerobatic flight.

Keywords: 3D path planning; sliding mode surface; multivariable super twisting algorithm; LMI

MSC: 93C10; 93C35

Citation: Cravioto, O.; Saldivar, B.; Jiménez-Lizárraga, M.; Ávila-Vilchis, J.C.; Aguilar-Ibañez, C. Sliding Surface-Based Path Planning for Unmanned Aerial Vehicle Aerobatics. *Mathematics* **2024**, *12*, 1047. https://doi.org/10.3390/math12071047

Academic Editor: Daniel-Ioan Curiac

Received: 2 March 2024
Revised: 27 March 2024
Accepted: 27 March 2024
Published: 30 March 2024

1. Introduction

The term aerobatics, which is a contraction of the words 'aerial' and 'acrobatics', refers to the practice of aerial maneuvers that involve unusual or abrupt changes of altitude or acceleration. Performing automated aerobatic maneuvers stands as a challenging problem within the unmanned aerial vehicle (UAV) research community. The development of such maneuvers allows for testing aerodynamic limits concerning maneuverability and control of an aerial vehicle [1]. Nowadays, flight mission requirements are becoming more complex due to the broad scope of civil and military applications, which include flying in uneven urban and natural environments, in many cases in the presence of obstacles requiring sudden and complex trajectory changes [2,3]. Under such scenarios, aerobatics becomes important, providing an alternative approach for navigating under the above-mentioned conditions. Training and recreation are additional applications of aerobatics.

The path planning task consists in determining a geometrical path that a vehicle, considering its dynamic characteristics and physical limitations, must follow in order to achieve a set goal [4–7]. In general, the path planning task involves the determination of a route that allows the vehicle to reach a goal. The simplest way to model a UAV path is by using segments that connect a number of waypoints [8,9].

Path planning is vital when different autonomous systems perform tasks in the same working environment since it aims at avoiding collisions. In [10], a bifurcation-based approach to generate different formation patterns is presented; it allows for generating a path for a UAV within the formation, avoiding obstacles and/or collisions with other UAVs.

Woo et al. [11] propose the use of the rapidly exploring random tree algorithm combined with the artificial potential field (P-RRT*) to generate an initial collision-free path. In addition, the line-of-sight path optimization (LoSPO) technique is applied to obtain the shortest path with reference to the initial one. The effectiveness of the combined path planning algorithms was demonstrated through the use of two- and three-dimensional digital terrain maps. This paper does not address the path planning for aerobatics; the method consists of connecting initial and final points using straight line trajectories. The algorithm becomes slow as it needs to generate different trajectories before determining the optimal one. In addition, it requires storing information on all the generated trajectories, which implies high computational consumption.

In [12], the trajectory optimization for a UAV carrying out a maritime radar surveillance mission is investigated. The proposed method aims at maximizing information obtained from the search area and minimizing fuel consumption. Quintic polynomials are used to generate UAV paths due to their ability to provide complete and complex solutions while requiring few inputs.

In [7], a comprehensive survey of UAV path planning techniques is presented; comparison tables showing path length, optimality, completeness, cost efficiency, time efficiency, energy efficiency, robustness, and collision avoidance are included. Furthermore, several open research problems on UAV path planning and UAV network communication are explored. Energy efficiency, security, and privacy are among the mentioned open research problems.

In [13], a comprehensive review of more than 150 articles from 2000 to 2022 concerning path planning methods using optimization approaches is presented. These methods are classified into five categories: classical methods, heuristics, metaheuristics, machine learning, and hybrid algorithms. A critical analysis is provided for each category considering targeted objectives, constraints, and environments.

As reported in the above-mentioned surveys and in the references therein, research efforts have been devoted to ensuring optimal and collision-free paths between two locations while meeting requirements related to the UAV characteristics and the serving area. For the case of UAV aerobatics, besides defining the start and end points, a predefined path involving the desired acrobatic maneuvers between those two points should be characterized. For instance, in [14], the development of Pugachev's cobra maneuver in a quadrotor is addressed. This famous maneuver is used to perform aerobatic shows or in combat for a sudden brake. It consists in turning the aircraft vertically to perform the maneuver along with sudden deceleration. The quadrotor is suspended for a few seconds without lifting force during that time. The above represents a challenge since, in $\theta = \pi/2$, there is no thrust to compensate the gravity. The proposal applies only for this specific maneuver, and the algorithm presents vulnerabilities when the pitch angle is such that $\theta > \pi/2$.

Other acrobatic maneuvers for UAV can be found in [15], where Knife-Edge and Rolling Harrier are studied. Knife-Edge maneuvers are useful for flying between obstacles when the passage is narrower than the aircraft's wingspan. The goal is to maintain 90° roll while tracking a straight line at a constant altitude. In a Rolling Harrier maneuver, the aircraft flies at a constant altitude while maintaining a constant roll rate. This maneuver has little practical utility, but it demonstrates the aircraft's flight capability. In [15], in addition to the Knife-Edge and Rolling Harrier maneuvers, the hovering and aggressive turnaround flights are discussed. Each maneuver is executed in isolation. As future work, the use of these stunts as an alternative for avoiding obstacles is mentioned.

In [16], the problem of performing quadrotor aggressive maneuvers that are attitude-constrained is tackled. The trajectory generation is formulated as a quadratic programming problem with linear constraints. The underactuation of the vehicle is explored to embed the attitude constraints into the trajectory generation via constraints on the desired quadrotor acceleration. Experimental and simulation results illustrate the application of the proposed method to quadrotor maneuvers involving a 360° flip. Among the disadvantages of the approach proposed in [16], it can be mentioned that it was only applied to the flip maneuver and that it does not consider the presence of disturbances.

On the other hand, due to the presence of external forces such as wind gusts, UAVs are susceptible to deviation from the original intended path. In this regard, the use of the sliding mode approach has been a successful technique to cope with this control problem; see, for instance, [17–19]. Sliding mode control has been the subject of extensive research for several decades, mainly due to its ability to operate in the presence of matched uncertainties [20]. One drawback of conventional (first-order) sliding mode control is the presence of 'chattering' caused by the discontinuous control action. However, higher-order sliding mode control (HOSMC) addresses this issue by attenuating the chattering effect. HOSMC retains the robustness observed in first-order sliding modes, while enhancing the accuracy of the control system [21]. One disadvantage of implementing an r-th order sliding mode is the requirement of having access to the $r-1$ time derivatives of the sliding surface. However, in a specific type of second-order sliding modes, known as 'super twisting', there is no need for such derivative information. Most research works on sliding mode control focus on a single-input control structure. For multi-input systems, a transformation allows for decoupling the structure to deal with m single-input control structures. However, in [22], a multivariable super twisting structure is proposed; this structure, also known as super twisting vector control, is also considered in [20,23,24].

For the path-following problem in UAVs, several sliding-mode-based controllers have been proposed. See, for instance, [25], where a fractional-order improved super twisting proportional-integral-derivative sliding mode controller (STPIDSMC) is proposed to ensure fast convergence, high precision, and good robustness against stochastic perturbations and uncertainties. Numerical simulations illustrate the effectiveness of the proposed strategy. This paper does not address the case of acrobatic maneuvers; it only considers straight line trajectories, so the proposed sliding surfaces are linear.

In [26], a robust backstepping-based approach combined with sliding mode control is proposed for trajectory tracking of a quadrotor UAV subject to external disturbances and parameter uncertainties. Numerical simulations and experimental tests have been developed to verify the validity of the proposed control approach. As in [25], this paper does not consider aerobatics, but only straight line paths. The designed controller induces chattering that could cause implementation problems.

As explained, most research works do not consider the execution of a set of acrobatic maneuvers. In general, the problem of simple straight-line trajectories is addressed, and in some cases, the execution of no more than one isolated acrobatic maneuver is addressed.

In this paper, the underpinning theory is the sliding mode control technique. The major thrust of this technique is to regulate the dynamics of a system by ensuring that the system's state trajectory 'slides' along a specific surface in the state space. As it is well known, the first step in designing a sliding mode controller is defining a sliding surface, which represents the desired behavior of the system. The goal is to force the system's state trajectory to converge to and slide along this surface. This is where the aim of this paper aligns with the sliding mode technique purpose; i.e., the underlying idea is that when the vehicle is forced to move in the sliding regime, it will inevitably perform, even in the presence of disturbances, the desired aerobatics represented by a suitable sliding surface. In that sense, the design of sliding surfaces based on different mathematical structures (Hopf bifurcation, Viviani's curve, and logarithm function) that allows for characterizing a set of geometrical paths to compose a circuit of acrobatic maneuvers is proposed.

This work is inspired in the results presented in [27], where the path planning problem for a fly mission, which depends on a circular surveillance motion around a mobile objective, is addressed. The proposed path-following method that allows the circular flight makes use of the Hopf bifurcation. Another proposal that exploits the Hopf bifurcation to generate a periodic orbit behavior in an aircraft can be found in [28], where bifurcations are used to define the aircraft departure recovery dynamics. Invariant set theory is used to prove the existence and stability of the elliptical orbits.

The objectives pursued in [27,28] differ from the problem addressed in this work as explained below. In [27], the Hopf bifurcation is used to generate only a planar circular trajectory for a UAV around a mobile objective on the ground. In [28], the Hopf bifurcation allows for inducing periodic orbits similar to those that pilots would perform when there are spatial restrictions on takeoff and landing. Unlike what is proposed in [27,28], in this paper, we develop a novel strategy to solve the problem of 3D path planning to generate a series of acrobatic maneuvers that include a loop, a descending spiral, and a curve defined on the surface of a sphere, using, in addition to the Hopf bifurcation, Viviani's curve and a logarithm function.

The main focus of this article is on the characterization of the paths that define acrobatic maneuvers. The complete complex dynamics of a UAV are not considered; hence, we leave full orientation control for future research. Here, as a benchmark, the UAV is seen as a punctual mass whose 3D evolution is defined by three double integrators. Then, additionally, we propose the use of a super twisting vector control to steer the punctual point to the 3D path. The path planning problem involves only the definition of the translational movement of the punctual mass; the rotational movement (orientation) is not considered here.

1.1. Contributions

The contributions of this paper that, to the best of the authors' knowledge, are not published before in its present form are fourfold:

1. The introduction of nonlinear 3D sliding mode surfaces, taking advantage of the geometry profile of paths defined by the Hopf bifurcation, Viviani's window, and the logarithm function to execute different acrobatic maneuvers;
2. The design of a discontinuous vector sliding mode control that forces the system to perform the aerobatics, ensuring robustness against matched perturbations with finite time convergence characteristics;
3. The development of a practical stability analysis to determine the controller gains that allows relaxing the condition on the term that encompasses disturbances and unmodeled dynamics;
4. A suitable switching or commutation control to shift between different sliding surfaces during the planning.

In summary, this paper proposes a simple and original strategy to solve the path planning problem that exploits one of the key features of the sliding mode approach, which is the sliding manifold. This work can be used as a base to develop a complete strategy (considering position and orientation) for the execution of aggressive acrobatic maneuvers of unmanned aerial vehicles. The above has an important impact in both the academic and social fields due to the wide range of applications that can be addressed, such as entertainment and air shows, aerospace research and development, pilot training and skill development, search and rescue operations, among others.

1.2. Motivation

The main motivation of this study is that, to the best of the authors' knowledge, a simple and intuitive strategy does not exist in the literature for solving the path planning problem for unmanned aerial vehicle aerobatics. The use of mathematical structures in combination with the sliding mode strategy had not been exploited to solve the path planning problem. In this sense, to highlight the simplicity and cleverness of the method, a series of steps that guide the reader in the implementation of the proposed strategy for any predefined aerobatics can be found in Section 5.

1.3. Organization of the Paper

The rest of the paper is organized as follows: In Section 2, the problem statement is presented. Four nonlinear sliding surfaces associated with different acrobatic maneuvers are introduced. In Section 3, a super twisting vector sliding mode control is designed to ensure the practical stability of the predefined surfaces; a Lyapunov analysis that makes use of the descriptor method is provided. A switching control that allows swapping between two predefined surfaces is introduced in Section 4. Section 5 presents numerical results illustrating the effectiveness of the proposed approach. Finally, Section 6 gives some concluding remarks.

2. Problem Statement

2.1. Motion Context

Consider a punctual unitary mass (that can be considered to be located at the center of mass of a UAV) as the element that will be following the trajectories of interest. In addition, consider that this punctual mass is affected by an unknown term (see, for instance, [29,30]). The behavior of the punctual mass in three-dimensional space is modeled by three double integrators as follows:

$$\frac{d^2 i}{dt^2} = u_i + \xi_i \tag{1}$$

where $i \in \{x, y, z\}$, in which x, y, z are the Cartesian coordinates defining the space where the particle evolves. The terms u_x, u_y, u_z are control inputs in the x, y, and z directions, respectively. Each term ξ_i captures the combined action of bounded unknown external disturbances and unmodeled dynamics on the respective direction x, y, or z.

For a state space representation of the system, the state vector is chosen as follows:

$$\mathbf{x} = [x, y, z, v_x, v_y, v_z]^\intercal \tag{2}$$

where $v_i = \frac{di}{dt}$ is the translational velocity of the punctual mass in the i direction. Defining the control vector $\mathbf{u}$ as follows:

$$\mathbf{u} = [u_x, u_y, u_z]^\intercal \tag{3}$$

and the vector that encompasses disturbances and unmodeled dynamics as follows:

$$\boldsymbol{\xi} = [\xi_x, \xi_y, \xi_z]^\intercal \tag{4}$$

the system can be expressed in matrix form as follows:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}(\mathbf{u} + \boldsymbol{\xi}) \tag{5}$$

where the state matrix $\mathbf{A} \in \mathbb{R}^{6\times 6}$ and the input matrix $\mathbf{B} \in \mathbb{R}^{6\times 3}$ are given by the following:

$$\mathbf{A} = \begin{bmatrix} 0_{3\times 3} & I_{3\times 3} \\ 0_{3\times 3} & 0_{3\times 3} \end{bmatrix} \tag{6}$$

$$\mathbf{B} = \begin{bmatrix} 0_{3\times 3} \\ I_{3\times 3} \end{bmatrix} \tag{7}$$

where $0_{3\times 3}$ is the 3×3 zero matrix and $I_{3\times 3}$ is the identity matrix of dimension 3.

Disturbances and unmodeled dynamics are matched Lebesgue-measurable input functions such that $\|\boldsymbol{\xi}\| < L_\xi \in \mathbb{R}$. Consider the following error vector:

$$\mathbf{e} = \mathbf{x} - \mathbf{x_r} \tag{8}$$

where $\mathbf{x_r} \in \mathbb{R}^6$ is the reference state vector defined as follows:

$$\mathbf{x_r} = [x_r, y_r, z_r, 0, 0, 0]^\intercal \tag{9}$$

The desired position of the particle is identified with the Cartesian coordinates (x_r, y_r, z_r). The target position is defined according to the considered maneuver, as will be explained in Section 2.2. The error vector is defined as follows:

$$\mathbf{e} = [e_x, e_y, e_z, \dot{e}_x, \dot{e}_y, \dot{e}_z]^\intercal \tag{10}$$

Therefore, assuming a constant reference position, the error dynamics is defined by the following:

$$\dot{\mathbf{e}} = \mathbf{A}\mathbf{e} + \mathbf{B}(\mathbf{u} + \boldsymbol{\xi}) + \mathbf{A}\mathbf{x_r} \tag{11}$$

As mentioned before, the path planning design stands as one of the most crucial problems in aerial navigation. The next section describes in detail the proposed approach, which is based on the concept of sliding mode surface, for the path planning design to perform a series of aerobatic maneuvers.

2.2. Path Planning Design

Executing complex aerial assignments is becoming more relevant in the world of agile miniature aircraft. Such flights are required, for example, when navigating in hostile environments where the vehicle needs to perform difficult maneuvers like inverted flight or sudden direction changes. Studying the development of such movements serves as a basis for testing the limits of the vehicle. Nevertheless, solving the problem of the 3D path planning is first required.

In this section, we propose an approach to build up the path planning for the development of three different types of aerobatics: looping, eight on a sphere, and descending spiral. The looping consists in following a circular trajectory in a vertical plane. The eight on a sphere consists of two turns in opposite directions; the vehicle climbs and descends following a symmetric pattern on a spherical surface. During the spiral, the vehicle follows a circular pattern while descending.

Figure 1 shows a circuit consisting of seven path segments; three of them define the aforementioned aerobatic maneuvers, and the remaining four correspond to straight-line paths. The trajectories are defined on the basis of the inertial reference frame shown in Figure 1.

The proposed approach to build up the path planning uses the concept of nonlinear sliding mode surface (SMS). An SMS is commonly used to stabilize the trajectories of a given system or drive them to some operating mode (see, for instance, [20,21,31]). However, in this case, SMSs are designed to define the movement of a punctual mass that represents a UAV; a different surface is designed for each path segment of the circuit shown in Figure 1.

1. Logarithm-based surface defining straight-line paths in the 3D space ($\mathbf{S}_1$, $\mathbf{S}_3$, $\mathbf{S}_5$, $\mathbf{S}_7$);
2. Hopf-bifurcation-based parameterized surface defining the looping ($\mathbf{S}_2$);
3. Viviani's window-based surface defining the eight on a sphere ($\mathbf{S}_4$);
4. Hopf-bifurcation-based parameterized surface defining the descending spiral ($\mathbf{S}_6$).

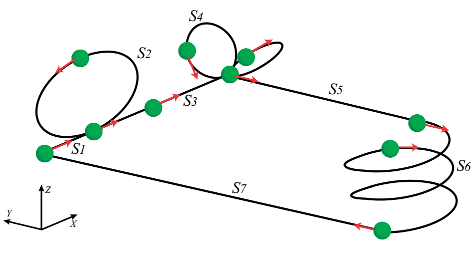

Figure 1. Three-dimensional planned circuit for aerobatics.

The key idea of the path planning proposal is to determine a suitable set of parameterized nonlinear SMSs that define the movement of the punctual mass representing a UAV for the execution of aerobatic maneuvers.

For the case of the looping and the spiral, the SMS exploits the properties of the Hopf bifurcation to generate a circular path. For the eight on a sphere, Viviani's window is used to define the movement on a spherical surface. Finally, to describe straight-line paths, logarithm functions that allow for reducing the convergence time are used.

In the remaining part of this section, the sliding surfaces that allow for generating the path planning corresponding to each acrobatic maneuver are presented. In Section 4, a suitable switching control is defined to shift between each pair of SMSs.

It is important to point out that this paper proposes four different types of nonlinear SMS for the path planning of the aforementioned maneuvers; however, following the design approach proposed in this work, other 3D parameterized curves can be converted to an SMS to develop a distinct path planning and execute different maneuvers.

2.3. Logarithm-Based Surface: Straight-Line Path Segment

As shown in Figure 1, the circuit starts and ends with straight-line path segments, and between two different aerobatics, there is also a straight-line path. In this section, an SMS proposal for generating the 3D straight-line paths is presented. Following [32], the considered sliding surface makes use of the natural logarithm function, which allows faster convergence properties to the surface. The 3D SMS is defined in vector form as follows:

$$\mathbf{S}_j = \begin{bmatrix} \dot{e}_{xj} + c'_{xj}e_{xj} + c_{xj}\ln(k_{l1}|e_{xj}|+1)\,\mathrm{sign}(e_{xj}) \\ \dot{e}_{yj} + c'_{yj}e_{yj} + c_{yj}\ln(k_{l2}|e_{yj}|+1)\,\mathrm{sign}(e_{yj}) \\ \dot{e}_{zj} + c'_{zj}e_{zj} + c_{zj}\ln(k_{l3}|e_{zj}|+1)\,\mathrm{sign}(e_{zj}) \end{bmatrix} \tag{12}$$

where $\mathbf{S}_j = [S_{jx}, S_{jy}, S_{jz}]^\intercal$, $j \in \{1,3,5,7\}$ corresponds to each straight-line path. The parameters c_{xj}, c'_{xj}, c_{yj}, c'_{yj}, c_{zj}, c'_{zj}, k_{l1}, k_{l2}, and k_{l3} are positive design constants. The position error is defined by the following:

$$\mathbf{e}_l^\intercal = [e_{xj}, e_{yj}, e_{zj}]$$

where $e_{xj} = x - x_{rj} \in \mathbb{R}$, $e_{yj} = y - y_{rj} \in \mathbb{R}$, and $e_{zj} = z - z_{rj} \in \mathbb{R}$. The current Cartesian coordinates of the vehicle's position are denoted by $x, y, z \in \mathbb{R}$. The coordinates x_{rj}, y_{rj}, z_{rj} define a target position; in this case, it corresponds to the end point of the straight-line path defined by $\mathbf{S}_j$. For example, x_{r1}, y_{r1}, z_{r1} denote the end point of the segment defined by $\mathbf{S}_1$. Notice that if $\mathbf{S}_j = 0$, a set of three differential equations describing the error dynamics is obtained. A stability proof of this system of equations can be found in [32]. The stable error dynamics allows the convergence of the vehicle to the target position.

To understand the behavior of the surface $\mathbf{S}_j$, consider, for instance, the first element of (12). If we assume that $S_{jx} = 0$, then

$$\dot{e}_{xj} = -c'_{xj}e_{xj} - c_{xj}\ln(k_{l1}|e_{xj}|+1)\,\mathrm{sign}(e_{xj}) \tag{13}$$

Figure 2 shows the phase diagram of the dynamics defined by Equation (13). Figure 3 shows the behavior of the variable $\dot{e}_{xj}$ with respect to time, for $c'_{xj} = c_{xj} = 1$ and different values of k_{l1}. The curves shown in Figures 2 and 3 are compared with the ones corresponding to the linear equation:

$$\dot{e}_{xj} = -c'_{xj}e_{xj} \tag{14}$$

obtained by choosing $k_{l1} = 0$ (in yellow). Figure 2 shows that the curves corresponding to the nonlinear Equation (13) present an almost-linear behavior far from the equilibrium point (zero) and a nonlinear one when close to it. Evidently, this fact allows for accelerating the convergence to the equilibrium point. Notice also that the convergence to zero of the trajectories corresponding to (13) is faster compared with the one of the linear Equation (14) (see Figure 3).

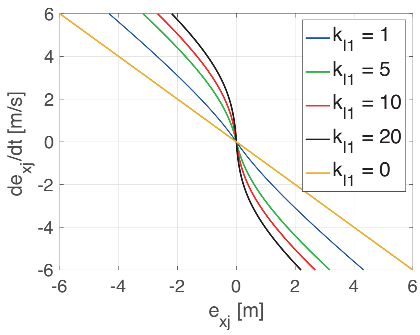

Figure 2. Phase diagram for $c_{xj} = c'_{xj} = 1$.

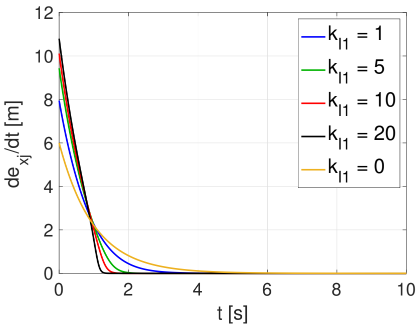

Figure 3. Evolution of $\dot{e}_{xj}$ for $cxj = c'_{xj} = 1$.

Following [32], the Lyapunov function $V(\mathbf{e}_l) = \frac{1}{2}\mathbf{e}_l^{\mathsf{T}} I_{3\times 3}\mathbf{e}_l$ is used to analyze the stability of the dynamics resulting from $\mathbf{S}_j = 0$.

The time derivative of $V(\mathbf{e}_l)$ is given by the following:

$$\dot{V}(\mathbf{e}_l) = \mathbf{e}_l^{\mathsf{T}} I_{3\times 3}\dot{\mathbf{e}}_l$$

In view of (12), for $\mathbf{S}_j = 0$, one obtains the following:

$$\begin{aligned} \dot{V}(\mathbf{e}_l) \quad = \quad & -c'_{xj}e_{xj}^2 - c_{xj}|e_{xj}|\ln(k_{l1}|e_{xj}|+1) \\ & -c'_{yj}e_{yj}^2 - c_{yj}|e_{yj}|\ln(k_{l2}|e_{yj}|+1) \\ & -c'_{zj}e_{zj}^2 - c_{zj}|e_{zj}|\ln(k_{l3}|e_{zj}|+1) \end{aligned} \tag{15}$$

The Lyapunov stability condition $\dot{V}(\mathbf{e}_l) \leq 0$ is satisfied for any positive constants c'_{xj}, c'_{yi}, c'_{zj}, c_{xi}, c_{yi}, c_{zj}, k_{l1}, k_{l2}, and k_{l3}.

2.4. Bifurcation Sliding Mode Surface: Looping and Spiral

In a differential equation, the local birth or death of a periodic solution (self-excited oscillation) from an equilibrium as a parameter crosses a critical value is referred to as Hopf bifurcation. This phenomenon occurs when a complex conjugate pair of eigenvalues of the linearized flow at a fixed point becomes purely imaginary [33,34]. In the case of a set of differential equations with a single equilibrium point at the origin that exhibits a Hopf bifurcation behavior, the solution behaves according to the parameter values: it circles the equilibrium point, it converges to the origin describing a spiral shape, or it presents a limit cycle whose radius grows from the equilibrium point.

The design of the sliding surface to execute the looping and spiral is based on the results presented in [27], where the path planning problem for a fly mission involving a circular motion is addressed. The proposed method that allows the circular flight makes use of the Hopf bifurcation.

The path planning to perform the loop and the spiral takes advantage of the *bifurcation sliding mode surface* introduced in [27], which is a parameterized nonlinear SMS including Hopf bifurcation dynamics. The key idea is to force the system trajectories to evolve according to the solution of a set of differential equations describing a circular path around the equilibrium point.

Notice in Figure 4 that the looping defines a movement in a plane parallel to the XZ plane (there is no displacement on the Y axis), while the spiral (Figure 5) is defined in the 3D space. The spiral consists in a circular path, defined in a plane parallel to the XY plane, that descends along the Z axis.

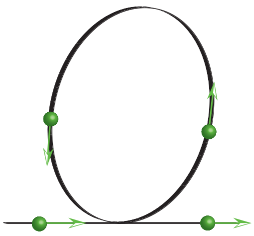

Figure 4. Acrobatic maneuvers: looping.

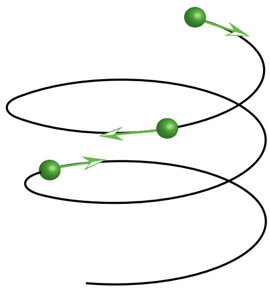

Figure 5. Acrobatic maneuvers: spiral.

The proposed sliding surface for generating the looping and the spiral is defined as follows:

$$\mathbf{S}_p = \begin{bmatrix} \dot{e}_{xp} - \mu_p e_{xp} - \gamma_p e_{kp} + e_{xp}(e_{xp}^2 + e_{kp}^2) \\ \dot{e}_{kp} + \gamma_p e_{xp} - \mu_p e_{kp} + e_{kp}(e_{xp}^2 + e_{kp}^2) \\ \dot{e}_{qp} + c'_{qp} e_{qp} + c_{qp} \ln(k_{lqp}|e_{qp}| + 1)\,\mathrm{sign}(e_{qp}) \end{bmatrix} \tag{16}$$

where $\mathbf{S}_p = [S_{px}, S_{pk}, S_{pq}]^\intercal$, μ_p, γ_p, c'_{qp}, c_{qp}, and k_{lqp} are design constants, $p \in \{2, 6\}$, $k, q \in \{y, z\}$, with $q \neq k$. For the spiral, $p = 6$, $k = y$, $q = z$, and for the looping, $p = 2$, $k = z$, $q = y$.

The position error is defined by the vector $[e_{xp}, e_{yp}, e_{zp}]^\intercal$, where $e_{xp} = x - x_{rp}$, $e_{yp} = y - y_{rp}$, and $e_{zp} = z - z_{rp}$, where the coordinates x_{rp}, y_{rp}, and z_{rp} define the center of the circular path related to each maneuver, and as before, the current Cartesian coordinates of the vehicle's position are denoted by $x, y, z \in \mathbb{R}$.

The first two components of Equation (16) exploit the Hopf bifurcation properties, while the third one is of the form of the logarithm-based surface defined in Section 2.3. For $\mathbf{S}_p = 0$, the first two elements correspond to a set of differential equations whose solution must present a stable limit cycle. This behavior allows for defining a circular path in a 2D space (in a plane parallel to the XZ plane for the looping or parallel to the XY plane for the spiral) around an appropriate equilibrium point.

The sliding surface defined in (16) is proposed based on the ideas explained below. A sliding-mode-based controller is designed in Section 3 to guarantee $\mathbf{S}_p = \dot{\mathbf{S}}_p = 0$. In this scenario, from Equation (16), one obtains the following:

$$\dot{e}_{xp} = \mu_p e_{xp} + \gamma_p e_{kp} - e_{xp}(e_{xp}^2 + e_{kp}^2) \tag{17}$$

$$\dot{e}_{kp} = -\gamma_p e_{xp} + \mu_p e_{kp} - e_{kp}(e_{xp}^2 + e_{kp}^2) \tag{18}$$

$$\dot{e}_{qp} = -c'_{qp} e_{qp} - c_{qp} \ln(k_{lqp}|e_{qp}| + 1)\,\mathrm{sign}(e_{qp}) \tag{19}$$

Let us consider the pair of Equations (17) and (18) that exhibiting Hopf bifurcation properties. To define the radius of the circular path and the rotational speed at which the vehicle must follow it, Equations (17) and (18) are transformed to the polar coordinates defined by $r_h = \sqrt{e_{xp}^2 + e_{kp}^2}$ and $\theta_h = \tan^{-1}(e_{kp}/e_{xp})$ as follows:

$$\begin{aligned} e_{xp} &= r_h \cos(\theta_h) \\ e_{kp} &= r_h \sin(\theta_h) \end{aligned} \tag{20}$$

Taking the time derivative of (20) yields the following:

$$\begin{aligned} \dot{e}_{xp} &= \dot{r}_h \cos(\theta_h) - r_h \dot{\theta}_h \sin(\theta_h) \\ \dot{e}_{kp} &= \dot{r}_h \sin(\theta_h) + r_h \dot{\theta}_h \cos(\theta_h) \end{aligned} \tag{21}$$

By substituting (20) into (17) and (18), one obtains the following:

$$\begin{aligned} \dot{e}_{xp} &= r_h(\mu_p - r_h^2) \cos(\theta_h) + \gamma_p r_h \sin(\theta_h) \\ \dot{e}_{kp} &= r_h(\mu_p - r_h^2) \sin(\theta_h) - \gamma_p r_h \cos(\theta_h) \end{aligned} \tag{22}$$

Equating (21) and (22) yields the following:

$$\dot{r}_h = r_h(\mu_p - r_h^2) \tag{23}$$

$$\dot{\theta}_h = -\gamma_p \tag{24}$$

Note that the first differential Equation (23) has two solutions: $r_h = 0$, $r_h = \sqrt{\mu_p}$, and that the solution of the second one is $\theta_h = -\gamma_p t$. This implies that the radius of the circular path is 0 or $\sqrt{\mu_p}$, and that the rotational speed at which the punctual mass representing a UAV will follow it is defined by γ_p. According to Jiménez and Jiménez-Lizárraga [27], for system (23) and (24), $r_h \to 0$ (asymptotically) for $\mu_p \leq 0$, and $r_h \to \sqrt{\mu_p}$ (asymptotically) for $\mu_p > 0$.

Therefore, the looping and the spiral are executed by forcing the existence of a stable limit cycle with the radii $\sqrt{\mu_2}$ and $\sqrt{\mu_6}$, respectively, where $\mu_2 > 0$ and $\mu_6 > 0$.

We can also identify this behavior by looking at the solution of the differential Equation (21), which is given by the following:

$$e_{xp} = \sqrt{\mu_p} \cos(\gamma_p t) \tag{25}$$

$$e_{kp} = -\sqrt{\mu_p} \sin(\gamma_p t) \tag{26}$$

By substituting (25) and (26) into the position errors defined as $e_{xp} = x - x_{rp}$, $e_{kp} = k - k_{rp}$, where $k = y$ for the spiral, and $k = z$ for the looping, one obtains the following:

$$x = x_{rp} + \sqrt{\mu_p} \cos(\gamma_p t) \tag{27}$$

$$k = k_{rp} - \sqrt{\mu_p} \sin(\gamma_p t) \tag{28}$$

Equations (27) and (28) describe a circular trajectory with the center at (x_{rp}, k_{rp}) and the radius $\sqrt{\mu_p}$.

2.5. Viviani's Window-Based Surface: Eight on a Sphere

This section presents an approach for generating the path planning to develop the acrobatic maneuver called eight on a sphere. The path segment corresponding to this maneuver is generated through a specific Clelia curve (a Clelia curve or Clélie is a curve limited by the surface of the radius of a sphere (see [35])); if the sphere's diameter is intersected by a cylinder on a parallel axis, the Clélie is called Viviani's window (see, for instance, [36]).) called Viviani's curve (also known as Viviani's window), which is the space curve defined by the intersection of a cylinder and a sphere (see Figures 6 and 7).

The proposed SMS to perform the mentioned maneuver is given by the following:

$$\mathbf{S}_4 = \begin{bmatrix} S_{4x} \\ S_{4y} \\ S_{4z} \end{bmatrix} = \begin{bmatrix} \dot{e}_{x4} - \alpha r_v + 2\alpha e_{z4} \\ \dot{e}_{y4} - \alpha r_v \cos(\alpha t) \\ \dot{e}_{z4} - 2\alpha e_{x4} \end{bmatrix} \tag{29}$$

where α and r_v are design parameters described below. The position error is defined by the vector $\left[e_{x4}, e_{y4}, e_{z4}\right]^\intercal$, where

$$\begin{aligned} e_{x4} &= x - x_{r4} \\ e_{y4} &= y - y_{r4} \\ e_{z4} &= z - z_{r4} \end{aligned} \tag{30}$$

The coordinates x_{r4}, y_{r4}, and z_{r4} define the start and end points of this acrobatic maneuver that are the same, i.e., the point where the two turns of the eight shape intersect. As before, the current Cartesian coordinates of the vehicle's position are denoted by $x, y, z \in \mathbb{R}$.

The key idea for the design of the SMS (29) consists in defining a set of equations for the error dynamics whose solution evolves along the aforementioned curve.

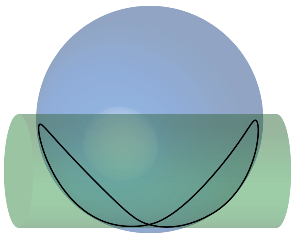

Figure 6. Geometry of Viviani's window.

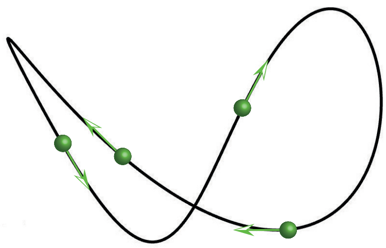

Figure 7. Viviani's window used to generate the acrobatic maneuver eight on a sphere.

Viviani's window is defined by the following:

$$\begin{aligned} x'^2 + y'^2 + z'^2 &= r_v^2 \\ z'^2 + y'^2 &= -r_v z' \end{aligned} \tag{31}$$

which defines the intersection of a sphere with the center $(0,0,0)$ and radius r_v and a cylinder of radius $r_v/2$, where x', y', and z' are Cartesian coordinates in the 3D space. The system of parametric coordinates of (31) is given by the following:

$$\begin{aligned} x' &= r_v \cos(\alpha t)\sin(\alpha t) \\ y' &= r_v \sin(\alpha t) \\ z' &= -r_v \cos^2(\alpha t) \end{aligned} \tag{32}$$

where α is a design constant. The system of Equation (32) defines the eight on a sphere curve where the intersection of the two turns is located at point $(0,0,-r_v)$. In order to move the start and end point of this maneuver to the origin $(0,0,0)$, the curve must be translated r_v units along the Z axis; i.e, the third equation of (32) must be modified as $z' = r_v - r_v \cos^2(\alpha t)$.

Now, to define the path planning for the eight on a sphere curve starting at the point (x_{r4}, y_{r4}, z_{r4}) that corresponds to the end point of the straight-line path defined by $\mathbf{S}_3$ (see the circuit described in Section 2.2), one has that the coordinates of the vehicle's position must be defined by the following:

$$\begin{aligned} x &= r_v \cos(\alpha t)\sin(\alpha t) + x_{r4} \\ y &= r_v \sin(\alpha t) + y_{r4} \\ z &= r_v - r_v \cos^2(\alpha t) + z_{r4} \end{aligned} \tag{33}$$

Then, in view of (30), one has the following:

$$\begin{aligned} e_{x4} &= r_v \cos(\alpha t)\sin(\alpha t) \\ e_{y4} &= r_v \sin(\alpha t) \\ e_{z4} &= r_v - r_v \cos^2(\alpha t) \end{aligned} \tag{34}$$

which can be rewritten as follows:

$$\begin{aligned} e_{x4} &= r_v \cos(\alpha t)\sin(\alpha t) \\ e_{y4} &= r_v \sin(\alpha t) \\ e_{z4} &= r_v \sin^2(\alpha t) \end{aligned} \tag{35}$$

The time derivative of (35) is given by the following:

$$\begin{aligned} \dot{e}_{x4} &= \alpha r_v [\cos^2(\alpha t) - \sin^2(\alpha t)] \\ \dot{e}_{y4} &= \alpha r_v \cos(\alpha t) \\ \dot{e}_{z4} &= 2\alpha r_v \sin(\alpha t)\cos(\alpha t) \end{aligned} \tag{36}$$

which can be written as follows:

$$\begin{aligned} \dot{e}_{x4} &= \alpha r_v [1 - 2\sin^2(\alpha t)] = \alpha r_v - 2\alpha e_{z_4} \\ \dot{e}_{y4} &= \alpha r_v \cos(\alpha t) \\ \dot{e}_{z4} &= 2\alpha r_v \sin(\alpha t)\cos(\alpha t) = 2\alpha e_{x4} \end{aligned} \tag{37}$$

Note that if $\mathbf{S}_4 = 0$, the system of Equation (37) is obtained.

The proposed path planning approach requires the design of a suitable control law that guarantees $\mathbf{S}_j = \mathbf{S}_p = \mathbf{S}_4 = 0$, $j \in \{1,3,5,7\}$, $p \in \{2,6\}$. Section 3 presents the design of a vector controller that allows for achieving this goal.

3. Super Twisting Vector Control

In this section, a super twisting vector sliding mode control (STVSMC) that guarantees the movement on the sliding surfaces defined in (12), (16), and (29) is proposed.

Notice that the seven nonlinear surfaces of interest $\mathbf{S}_j$, $\mathbf{S}_p$, $\mathbf{S}_4$, $j \in \{1,3,5,7\}$, and $p \in \{2,6\}$ can be represented in the general form (38), where subscripts are not needed anymore:

$$\mathbf{S} = \mathbf{S_f e} + \mathbf{f}(\mathbf{e}, t) \tag{38}$$

where $\mathbf{e}$ is defined as in (10). The linear terms are included in $\mathbf{S_f e}$ ($\mathbf{S_f} \in \mathbb{R}^{3\times 6}$), and $\mathbf{f}(\mathbf{e}, t) \in \mathbb{R}^3$ encompasses the nonlinear ones.

For example, the surface $\mathbf{S}_4$ defined in (29) admits the representation (38) with the following:

$$\begin{aligned} \mathbf{S_f} &= \begin{bmatrix} 0 & 0 & 2\alpha & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ -2\alpha & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \\ \mathbf{f}(\mathbf{e}, t) &= \begin{bmatrix} -\alpha r_v \\ -\alpha r_v \cos(\alpha t) \\ 0 \end{bmatrix} \\ \mathbf{e} &= [e_{x4}, e_{y4}, e_{z4}, \dot{e}_{x4}, \dot{e}_{y4}, \dot{e}_{z4}]^{\mathsf{T}} \end{aligned} \tag{39}$$

Taking the time derivative of (38) and substituting the error dynamics (11) yields the following:

$$\dot{\mathbf{S}} = \mathbf{S_f A e} + \mathbf{S_f B u} + \mathbf{S_f B}\boldsymbol{\xi} + \frac{d}{dt}\mathbf{f}(\mathbf{e}, t) + \mathbf{S_f A x_r} \tag{40}$$

By defining the variables $\mathbf{a}(\mathbf{e}, t) := \mathbf{S_f A e} + \frac{d}{dt}\mathbf{f}(\mathbf{e}, t) + \mathbf{S_f A x_r}$ and $\mathbf{b} := \mathbf{S_f B}$, Equation (40) can be rewritten as follows:

$$\dot{\mathbf{S}} = \mathbf{a}(\mathbf{e}, t) + \mathbf{b}(\mathbf{u} + \boldsymbol{\xi}) \tag{41}$$

Note that, for all surfaces, one has that $\mathbf{b} = I_{3\times 3}$.

The controller $\mathbf{u}$ is defined as follows:

$$\mathbf{u} = -\mathbf{a}(\mathbf{e}, t) + \mathbf{v} \tag{42}$$

Following [22], the vector form of the super twisting control is used, i.e.,

$$\begin{aligned} \mathbf{v} &= -k_1 \frac{\mathbf{S}}{\|\mathbf{S}\|^{1/2}} - k_2\mathbf{S} + \mathbf{Z} \\ \dot{\mathbf{Z}} &= -k_3 \frac{\mathbf{S}}{\|\mathbf{S}\|} - k_4\mathbf{S} \end{aligned} \tag{43}$$

where k_1, k_2, k_3, and $k_4 \in \mathbb{R}^+$ are design constants. Substituting (42), (43) into (41), one obtains the following:

$$\begin{aligned} \dot{\mathbf{S}} &= -k_1 \frac{\mathbf{S}}{\|\mathbf{S}\|^{1/2}} - k_2\mathbf{S} + \mathbf{Z} + \boldsymbol{\xi} \\ \dot{\mathbf{Z}} &= -k_3 \frac{\mathbf{S}}{\|\mathbf{S}\|} - k_4\mathbf{S} \end{aligned} \tag{44}$$

In [22], it was demonstrated that, for system (44), there exist a range of values for the gains k_1, k_2, k_3, and k_4, such that the variables $\mathbf{S}$ and $\dot{\mathbf{S}}$ are forced to zero in finite time and remain zero for all subsequent time when the term that encompasses unmodeled dynamics and disturbances $\boldsymbol{\xi}$ is bounded by a linear function of the norm of the variable $\mathbf{S}$, i.e., $\|\boldsymbol{\xi}\| < \delta_1\|\mathbf{S}\|$, where $\delta_1 > 0$ is a known scalar bound.

To relax the condition on ξ, in what follows, a practical stability analysis is developed for system (44) under the assumption that ξ is just bounded in norm, i.e., $\|\xi\| < L_{\xi}$, for a known constant L_{ξ}. The proposed analysis is based on the guidelines of the descriptor method (see [37]) to reduce the conservatism of the stability conditions stated in terms of matrix inequalities.

To develop the stability analysis, the following vectors are defined:

$$\chi := \begin{bmatrix} \frac{\mathbf{S}^{\mathsf{T}}}{\|\mathbf{S}\|^{1/2}} & \frac{\mathbf{S}^{\mathsf{T}}}{\|\mathbf{S}\|} & \mathbf{S}^{\mathsf{T}} & \mathbf{Z}^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}}, \tag{45}$$

$$\bar{\xi} := \begin{bmatrix} \xi^{\mathsf{T}} & \frac{\xi^{\mathsf{T}}}{\|\mathbf{S}\|^{1/2}} & \frac{\xi^{\mathsf{T}}}{\|\mathbf{S}\|} & \frac{\xi^{\mathsf{T}}\mathbf{S}\mathbf{S}^{\mathsf{T}}}{\|\mathbf{S}\|^{5/2}} & \frac{\xi^{\mathsf{T}}\mathbf{S}\mathbf{S}^{\mathsf{T}}}{\|\mathbf{S}\|^{3}} \end{bmatrix}^{\mathsf{T}} \tag{46}$$

In terms of the variables χ and $\bar{\xi}$, system (44) can be expressed as follows:

$$\begin{aligned} \dot{\chi} &= \mathbf{A_1}\chi + \frac{1}{\|\mathbf{S}\|^{1/2}}\mathbf{A_2}\chi + \mathbf{A_3}\frac{1}{\|\mathbf{S}\|}\chi \\ &\quad + \mathbf{A_4}\frac{\mathbf{S}\mathbf{S}^{\mathsf{T}}}{\|\mathbf{S}\|^{5/2}}\chi + \mathbf{A_5}\frac{\mathbf{S}\mathbf{S}^{\mathsf{T}}}{\|\mathbf{S}\|^{3}}\chi + \mathbf{B_S}\bar{\xi} \end{aligned} \tag{47}$$

where

$$\mathbf{A_1} = \begin{bmatrix} -\frac{k_2}{2}I_{3\times3} & -\frac{k_1}{2}I_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ -k_1 I_{3\times3} & 0_{3\times3} & -k_2 I_{3\times3} & I_{3\times3} \\ 0_{3\times3} & -k_3 I_{3\times3} & -k_4 I_{3\times3} & 0_{3\times3} \end{bmatrix} \tag{48}$$

$$\mathbf{A_2} = \begin{bmatrix} 0_{3\times9} & I_{3\times3} \\ 0_{9\times9} & 0_{9\times3} \end{bmatrix} \tag{49}$$

$$\mathbf{A_3} = \begin{bmatrix} 0_{3\times9} & 0_{3\times3} \\ 0_{3\times9} & I_{3\times3} \\ 0_{6\times9} & 0_{6\times3} \end{bmatrix} \tag{50}$$

$$\mathbf{A_4} = \begin{bmatrix} 0_{3\times9} & -\frac{1}{2}I_{3\times3} \\ 0_{9\times9} & 0_{9\times3} \end{bmatrix} \tag{51}$$

$$\mathbf{A_5} = \begin{bmatrix} 0_{3\times9} & 0_{3\times3} \\ 0_{3\times9} & -I_{3\times3} \\ 0_{6\times9} & 0_{6\times3} \end{bmatrix} \tag{52}$$

$$\mathbf{B_S} = \begin{bmatrix} 0_{3\times3} & I_{3\times3} & 0_{3\times3} & -\frac{1}{2}I_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & I_{3\times3} & 0_{3\times3} & -I_{3\times3} \\ I_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \end{bmatrix} \tag{53}$$

and $\bar{\xi}$ is such that

$$(\mathbf{B_S}\bar{\xi})^{\mathsf{T}}\mathbf{K}_{\bar{\xi}}(\mathbf{B_S}\bar{\xi}) \leq 1, \quad \forall t \geq 0 \tag{54}$$

the matrix $\mathbf{K}_{\bar{\xi}} \in \mathbb{R}^{12\times12}$ is strictly positive definite, i.e., $\mathbf{K}_{\bar{\xi}} > 0$.

Remark 1. *Note that a necessary condition for inequality (54) to be satisfied is* $\|\xi\| < L_{\xi}$.

Lemma 1 presented below is useful for the stability proof of Proposition 1 [38,39].

Lemma 1. *Consider a system of the following form:*

$$\dot{\mathbf{x}}(\mathbf{t}) = \mathbf{f}(\mathbf{x}(t)) + \xi, \qquad \mathbf{x}(0) = \mathbf{x}_0$$

where $\mathbf{x}(t) \in \mathbb{R}^n$ *is the state,* $\mathbf{x}_0$ *is the initial condition, and* $\boldsymbol{\xi}$ *denotes external perturbations such that* $\|\boldsymbol{\xi}\| < L_{\xi}$*.*

If there exists a Lyapunov function $V(\mathbf{x})$ *that satisfies the following:*

$$\alpha_1 \|\mathbf{x}(t)\|^2 \leq V(\mathbf{x}(t)) \leq \alpha_2 \|\mathbf{x}(t)\|^2 \;\; \forall t \geq 0 \tag{55}$$

and

$$\frac{d}{dt} V(\mathbf{x}(t)) + \gamma V(\mathbf{x}(t)) \leq \beta, \;\; \forall t \geq 0 \tag{56}$$

where α_1*,* α_2*,* γ*, and* β *are positive constants, then, for any initial condition* $\mathbf{x}_0$*, the following practical exponential estimate of the response holds:*

$$\|\mathbf{x}(t)\|^2 \leq e^{-\gamma t} \left(\frac{\alpha_2}{\alpha_1} \|\mathbf{x}_0\|^2 - \frac{\beta}{\alpha_1 \gamma} \right) + \frac{\beta}{\alpha_1 \gamma} \tag{57}$$

Furthermore, if the initial condition is such that $\|\mathbf{x}_0\|^2 \leq \frac{\beta}{\gamma \alpha_2}$*, one has that*

$$\|\mathbf{x}(t)\|^2 \leq \frac{\beta}{\gamma \alpha_1}, \quad \forall t \geq 0$$

For an initial condition that satisfies $\|\mathbf{x}_0\|^2 > \frac{\beta}{\gamma \alpha_2}$*,*

$$\|\mathbf{x}(t)\|^2 \leq \mu, \quad \forall t \geq T(\mu, \mathbf{x}_0)$$

where $\mu \geq \frac{\beta}{\gamma \alpha_1}$ *and* $T(\mu, \mathbf{x}_0)$ *satisfies the following:*

$$T(\mu, \mathbf{x}_0) \geq \frac{1}{\gamma} \ln \left(\frac{\alpha_2 \|\mathbf{x}_0\|^2 \gamma - \beta}{\mu \gamma \alpha_1 - \beta} \right)$$

Proof. Multiplying (56) by $e^{\gamma t}$ yields the following:

$$\frac{d}{dt} (e^{\gamma t} V(\mathbf{x})) \leq \beta e^{\gamma t} \tag{58}$$

Integrating the above expression from 0 to t yields the following:

$$V(\mathbf{x}) \leq \frac{\beta}{\gamma} (1 - e^{-\gamma t}) + e^{-\gamma t} V(\mathbf{x}_0) \tag{59}$$

From (55), one has that $V(\mathbf{x}_0) \leq \alpha_2 \|\mathbf{x}_0\|^2$, which yields the following:

$$V(\mathbf{x}) \leq \frac{\beta}{\gamma} (1 - e^{-\gamma t}) + \alpha_2 e^{-\gamma t} \|\mathbf{x}_0\|^2 \tag{60}$$

Then,

$$\|\mathbf{x}(t)\|^2 \leq \frac{\beta}{\gamma \alpha_1} + e^{-\gamma t} \left(\frac{\alpha_2}{\alpha_1} \|\mathbf{x}_0\|^2 - \frac{\beta}{\gamma \alpha_1} \right) \tag{61}$$

If the initial condition is such that $\|\mathbf{x}_0\|^2 \leq \frac{\beta}{\gamma \alpha_2}$, one has that

$$\|\mathbf{x}(t)\|^2 \leq \frac{\beta}{\gamma \alpha_1}, \quad \forall t \geq 0$$

For an initial condition that satisfies $\|\mathbf{x}_0\|^2 > \frac{\beta}{\gamma\alpha_2}$, it yields the following:

$$\|\mathbf{x}(t)\|^2 \leq \mu, \quad \forall t \geq T(\mu, \mathbf{x}_0)$$

where $\mu \geq \frac{\beta}{\gamma\alpha_1}$ and $T(\mu, \mathbf{x}_0)$ is obtained as follows. From (61), one has that

$$\|\mathbf{x}(t)\|^2 \leq \frac{\beta}{\gamma\alpha_1} + e^{-\gamma t}\left(\frac{\alpha_2}{\alpha_1}\|\mathbf{x}_0\|^2 - \frac{\beta}{\gamma\alpha_1}\right) \leq \mu$$

then,

$$e^{-\gamma t}\left(\frac{\alpha_2}{\alpha_1}\|\mathbf{x}_0\|^2 - \frac{\beta}{\gamma\alpha_1}\right) \leq \mu - \frac{\beta}{\gamma\alpha_1}$$

from the above inequality, one can see that $T(\mu, \mathbf{x}_0)$ satisfies the following:

$$T(\mu, \mathbf{x}_0) \geq \frac{1}{\gamma}\ln\left(\frac{\alpha_2\|\mathbf{x}_0\|^2\gamma - \beta}{\mu\gamma\alpha_1 - \beta}\right)$$

Observe that for $\|\mathbf{x}_0\|^2 > \frac{\beta}{\gamma\alpha_2}$ and $\mu \geq \frac{\beta}{\gamma\alpha_1}$, one has that $\alpha_2\|\mathbf{x}_0\|^2\gamma - \beta > 0$, $\mu\gamma\alpha_1 - \beta > 0$, and $\alpha_2\|\mathbf{x}_0\|^2\gamma - \beta > \mu\gamma\alpha_1 - \beta$; then, $T(\mu, \mathbf{x}_0)$ exists. $\square$

Proposition 1. *Consider system (47), which corresponds to the dynamics of the sliding surface given in (41) in a closed loop with the controller defined in (42). This controller guarantees the practical stability of the surfaces if matrices*

$$\Psi_1 = \begin{bmatrix} \gamma\mathbf{P_1} + \mathbf{P_2A_1} + \mathbf{A_1}^\top\mathbf{P_2}^\top & \mathbf{P_1} - \mathbf{P_2} + \mathbf{A_1}^\top\mathbf{P_3}^\top & \mathbf{P_2} + \mathbf{A_1}^\top\mathbf{P_4}^\top \\ * & -\mathbf{P_3} - \mathbf{P_3}^\top & \mathbf{P_3} - \mathbf{P_4}^\top \\ * & * & \mathbf{P_4} + \mathbf{P_4}^\top - \beta\mathbf{K}_\xi \end{bmatrix} \tag{62}$$

$$\Psi_2 = \begin{bmatrix} \mathbf{P_2A_2} + \mathbf{A_2}^\top\mathbf{P_2}^\top & \mathbf{A_2}^\top\mathbf{P_3}^\top & \mathbf{A_2}^\top\mathbf{P_4}^\top \\ * & 0 & 0 \\ * & * & 0 \end{bmatrix} \tag{63}$$

$$\Psi_3 = \begin{bmatrix} \mathbf{P_2A_3} + \mathbf{A_3}^\top\mathbf{P_2}^\top & \mathbf{A_3}^\top\mathbf{P_3}^\top & \mathbf{A_3}^\top\mathbf{P_4}^\top \\ * & 0 & 0 \\ * & * & 0 \end{bmatrix} \tag{64}$$

$$\Psi_4 = \begin{bmatrix} \mathbf{P_2A_4} + \mathbf{A_4}^\top\mathbf{P_2}^\top & \mathbf{A_4}^\top\mathbf{P_3}^\top & \mathbf{A_4}^\top\mathbf{P_4}^\top \\ * & 0 & 0 \\ * & * & 0 \end{bmatrix} \tag{65}$$

$$\Psi_5 = \begin{bmatrix} \mathbf{P_2A_5} + \mathbf{A_5}^\top\mathbf{P_2}^\top & \mathbf{A_5}^\top\mathbf{P_3}^\top & \mathbf{A_5}^\top\mathbf{P_4}^\top \\ * & 0 & 0 \\ * & * & 0 \end{bmatrix} \tag{66}$$

are such that $\Psi_1 \leq 0$, $\Psi_2 \leq 0$, $\Psi_3 \leq 0$, $\Psi_4 \leq 0$, *and* $\Psi_5 \leq 0$ *for any* k_1, k_2, k_3, *and* k_4, $\gamma > 0$, $\beta > 0$, $\mathbf{P_1} > 0$ *(symmetrically positive definite matrix).* $\mathbf{P_2}$, $\mathbf{P_3}$, *and* $\mathbf{P_4}$ *are slack matrices of appropriate dimension. Symmetric elements of symmetric matrices are denoted by* $*$.

Proof. Consider the Lyapunov function:

$$V(\chi) = \chi^{\mathsf{T}} \mathbf{P_1} \chi \tag{67}$$

where $\mathbf{P_1} > 0$. By taking the time derivative of the Lyapunov function one obtains the following:

$$\dot{V}(\chi) = \chi^{\mathsf{T}} \mathbf{P_1} \dot{\chi} + \dot{\chi}^{\mathsf{T}} \mathbf{P_1} \chi \tag{68}$$

To guarantee the practical stability of the system dynamics defined in (47), it is required that the condition stated in (56) in Lemma 1 be fulfilled. Following the procedure described below, stability conditions in terms of matrix inequalities are obtained.

Using the descriptor method proposed in [37], the following null term is considered:

$$\begin{aligned} 0 \quad = \quad & \left[\chi^{\mathsf{T}} \mathbf{P_2} + \dot{\chi}^{\mathsf{T}} \mathbf{P_3} + (\mathbf{B_S} \bar{\xi})^{\mathsf{T}} \mathbf{P_4}\right] \left[- \dot{\chi} + \mathbf{A_1} \chi + \frac{1}{\|\mathbf{S}\|^{1/2}} \mathbf{A_2} \chi + \mathbf{A_3} \frac{1}{\|\mathbf{S}\|} \chi \right. \\ & \left. + \mathbf{A_4} \frac{\mathbf{S}\mathbf{S}^{\mathsf{T}}}{\|\mathbf{S}\|^{5/2}} \chi + \mathbf{A_5} \frac{\mathbf{S}\mathbf{S}^{\mathsf{T}}}{\|\mathbf{S}\|^{3}} \chi + \mathbf{B_S} \bar{\xi} \right] \end{aligned} \tag{69}$$

where $\mathbf{P_2}$, $\mathbf{P_3}$, and $\mathbf{P_4}$ are slack variables of appropriate dimension.

From Equation (54), one can see the following:

$$-\beta((\mathbf{B_S}\bar{\xi})^{\mathsf{T}} \mathbf{K}_{\xi} \mathbf{B_S} \bar{\xi} - 1) \geq 0 \tag{70}$$

for any $\beta > 0$.

From Equations (67), (68), and (70), one can see the following:

$$\begin{aligned} \dot{V}(\chi) \quad + \quad & \gamma V(\chi) - \beta \leq \chi^{\mathsf{T}} \mathbf{P_1} \dot{\chi} + \dot{\chi}^{\mathsf{T}} \mathbf{P_1} \chi \\ & + \gamma \chi^{\mathsf{T}} \mathbf{P_1} \chi - \beta (\mathbf{B_S} \bar{\xi})^{\mathsf{T}} \mathbf{K}_{\xi} \mathbf{B_S} \bar{\xi} \end{aligned} \tag{71}$$

By adding the null term (69) and its transpose to the right-hand side of (71), one obtains the following in matrix form:

$$\begin{aligned} \dot{V}(\chi) + \gamma V(\chi) - \beta \leq \varrho^{\mathsf{T}} \Bigg(& \Psi_1 + \frac{1}{\|\mathbf{S}\|^{1/2}} \Psi_2 \\ & + \frac{1}{\|\mathbf{S}\|} \Psi_3 + \Psi_4 \frac{\mathbf{S}\mathbf{S}^{\mathsf{T}}}{\|\mathbf{S}\|^{5/2}} + \Psi_5 \frac{\mathbf{S}\mathbf{S}^{\mathsf{T}}}{\|\mathbf{S}\|^{3}} \Bigg) \varrho \end{aligned} \tag{72}$$

where $\varrho^{\mathsf{T}} = \left[\chi^{\mathsf{T}} \;\; \dot{\chi}^{\mathsf{T}} \;\; (\mathbf{B_S}\bar{\xi})^{\mathsf{T}}\right]$, and Ψ_1, Ψ_2, Ψ_3, Ψ_4, and Ψ_5 are defined in (62)–(66).

From the above expression, it is easy to see that condition (56) is satisfied if the conditions stated in Proposition 1 are fulfilled. □

4. Switching Control

The path planning approach presented in Section 2.2 to develop the proposed aerobatics requires the implementation of STVSMC for each path segment.

By assuming that the vehicle starts at the ground level, two additional straight-line paths must be considered to raise the vehicle a certain distance from the ground (defined by the surface $\mathbf{S}_0$), and then, after developing the proposed maneuvers, return it to the ground (surface $\mathbf{S}_8$).

As explained in Section 2, the circuit includes three acrobatic maneuvers (looping, eight on a sphere, and descending spiral) and four straight-line paths. Additionally, two straight-line paths were considered for takeoff and landing. Therefore, nine STVSMCs must be synthesized. Each controller is activated at a different time instant.

Figure 8 shows the reference points corresponding to each path segment, and Table 1 summarizes the data. The red points represent the start and end points of each path segment, and the green ones represent the center of the circular trajectory generated through the

Hopf bifurcation. As explained before, the reference points to generate the straight-line paths are located at the end of the segment, the ones corresponding to the looping and the spiral are located at the center of the respective circular path, and the reference point for the eight on a sphere is given by the point where the two turns intersect.

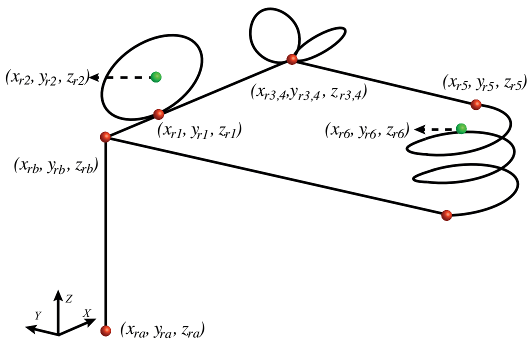

Figure 8. Path planning scheme.

Table 1. Aerobatics summary.

Aerobatic Maneuver	Surface	Initial Time	Final Time	Controller	Activation Function
Straight-line path	$\mathbf{S}_0$	t_0	t_1	u_0	$g_0(t)$
Straight-line path	$\mathbf{S}_1$	t_1	t_2	u_1	$g_1(t)$
Looping	$\mathbf{S}_2$	t_2	t_3	u_2	$g_2(t)$
Straight-line path	$\mathbf{S}_3$	t_3	t_4	u_3	$g_3(t)$
Eight on a sphere	$\mathbf{S}_4$	t_4	t_5	u_4	$g_4(t)$
Straight-line path	$\mathbf{S}_5$	t_5	t_6	u_5	$g_5(t)$
Spiral	$\mathbf{S}_6$	t_6	t_7	u_6	$g_6(t)$
Straight-line path	$\mathbf{S}_7$	t_7	t_8	u_7	$g_7(t)$
Straight-line path	$\mathbf{S}_8$	t_8	t_9	u_8	$g_8(t)$

The complete trajectory starts and ends at the same point, identified with the coordinates (x_{ra}, y_{ra}, z_{ra}). The controller u_0 allows for generating a straight-line path parallel to the Z axis to connect the initial point to the reference point (x_{rb}, y_{rb}, z_{rb}). Next, through the controller u_1, a new straight-line path parallel to the X axis is generated to reach the position (x_{r1}, y_{r1}, z_{r1}). Here, the first acrobatic maneuver (looping) takes place. To perform the looping, the controller u_2 allows the generation of a circular path defined by the limit cycle related to the Hopf bifurcation whose center is fixed at (x_{r2}, y_{r2}, z_{r2}). Note that since the radius of the circular trajectory is $\sqrt{\mu_2}$, then $(x_{r2}, y_{r2}, z_{r2}) = (x_{r1}, y_{r1}, z_{r1} + \sqrt{\mu_2})$.

Once the loop ends, the controller u_3 is used to generate a straight-line path to reach the position (x_{r3}, y_{r3}, z_{r3}), where the second acrobatic maneuver (eight on a sphere) takes place by applying the controller u_4. Note that the reference position related to the eight on a sphere (x_{r4}, y_{r4}, z_{r4}) coincides with (x_{r3}, y_{r3}, z_{r3}), denoted in Figure 8 as $(x_{r3,4}, y_{r3,4}, z_{r3,4})$.

The controller u_5 allows for generating the straight-line path required to reach the position (x_{r5}, y_{r5}, z_{r5}). At this point, through the controller u_6, the third acrobatic maneuver (spiral) takes place. As in the case of the looping, the trajectory is generated through a limit cycle, but in this case, the center of the circular path is located at (x_{r6}, y_{r6}, z_{r6}). The radius of the circular path is $\sqrt{\mu_6}$, then $(x_{r6}, y_{r6}, z_{r6}) = (x_{r5} - \sqrt{\mu_6}, y_{r5}, z_{r5})$.

After performing the spiral, the controller u_7 generates a straight-line path to reach the point (x_{rb}, y_{rb}, z_{rb}). Finally, a straight-line path is generated through the controller u_8 to descend and reach again the point (x_{ra}, y_{ra}, z_{ra}).

The switching law $\bar{\mathbf{u}}$ that allows for generating the whole trajectory is proposed as follows:

$$\bar{\mathbf{u}} = \sum_{m=0}^{8} (1 - g_m(t))\mathbf{u}_m \tag{73}$$

where $\mathbf{u}_m$, $m \in \{0,...,8\}$ are the controllers of the form (42); $\mathbf{u}_0$, $\mathbf{u}_1$, $\mathbf{u}_3$, $\mathbf{u}_5$, $\mathbf{u}_7$, and $\mathbf{u}_8$ allow for executing straight-line paths through the SMS defined in (12); $\mathbf{u}_2$ and $\mathbf{u}_6$ are the controllers designed to develop the looping and spiral, respectively, that use the SMS defined in (16); $\mathbf{u}_4$ is the controller that generates the eight on a sphere through the SMS (29); and g_m is the activation function defined by the following:

$$g_m(t) = \begin{cases} e^{-n(t_{m+1}-t)} & \text{if } t_m \leq t < t_{m+1} \\ 1 & \text{other case} \end{cases} \tag{74}$$

where $n \in \mathbb{R}^+$. Figure 9 shows the behavior of the activation function (74) for different values of n. As it can be observed, the activation function (74) allows a smoother transition between the different STVSMCs to generate each path segment. The transition speed is regulated through the parameter n.

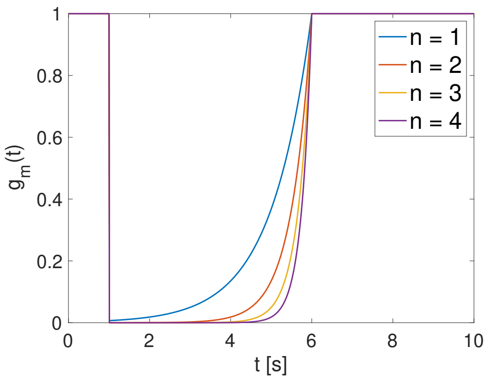

Figure 9. Behavior of the activation function.

Next, numerical simulations allow for illustrating the performance of the proposed path planning approach in the development of aerobatics.

5. Numerical Simulations

This section presents numerical simulation results to illustrate the performance of the controllers defined in (42) whose objective is driving the sliding surfaces (12), (16), and (29) to zero.

Perturbations of the form $\xi_i = 0.35\sin(3.5t)$, $i = \{x, y, z\}$, which clearly satisfy $\|\boldsymbol{\xi}\| < L_\xi$, are considered. The controller's gains are given by the following:

$$k_1 = 30 \quad k_2 = 1 \quad k_3 = 0.5 \quad k_4 = 0.001 \tag{75}$$

These controller gains ensure that the stability conditions stated in Proposition 1 are satisfied.

For the straight-line path segments whose SMSs are defined in (12), the parameters $c_j = [c_{xj}, c_{yj}, c_{zj}]$, $c'_j = [c'_{xj}, c'_{yj}, c'_{zj}]$ ($j \in \{1,3,5,7\}$), and $k_l = [k_{l1}, k_{l2}, k_{l3}]$ are chosen as follows:

$$c'_j = [2,2,2] \quad c_j = [2,2,2] \quad k_l = [1,1,1] \tag{76}$$

For the looping whose path planning uses the Hopf-bifurcation-based surface defined in (16), a unitary radius and a negative rotation direction with respect to the inertial reference frame shown in Figure 8 are assumed. The parameters of the surface are chosen as follows:

$$\mu_2 = 1 \quad \gamma_2 = -3 \quad c'_{y2} = 2 \quad c_{y2} = 2 \quad k_{ly2} = 1 \tag{77}$$

For the descending spiral, it is considered a unitary radius and a negative rotation direction with respect to the inertial reference frame shown in Figure 8. Then, the constants are proposed as follows:

$$\mu_6 = 1 \quad \gamma_6 = 2 \quad c'_{z6} = 0.2 \quad c_{z6} = 0.2 \quad k_{lz6} = 1 \tag{78}$$

For the eight on a sphere, which uses Viviani's window-based surface defined in (29), it is considered a unitary radius of the sphere; then, the parameters are chosen as follows:

$$\alpha = -3 \quad r_v = 1 \tag{79}$$

In the simulation, the whole trajectory is completed in 19.85 s. Figure 10 shows the evolution of the position of the punctual mass (representing an aerial vehicle) defined by the Cartesian coordinates x, y, and z. Table 2 shows the simulation data.

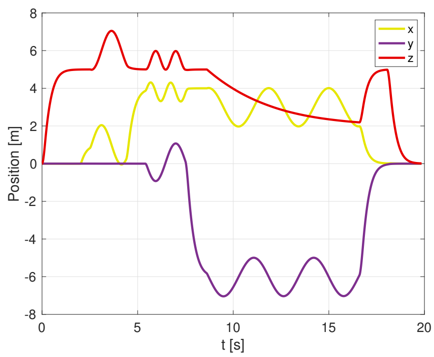

Figure 10. Position.

Table 2. Path planning simulation data.

	Path Segment	Initial Time (s)	Final Time (s)	Initial Position	Final Position
1.	Straight line	0	2	(0, 0, 0)	(0, 0, 5)
2.	Straight line	2	2.5	(0, 0, 5)	(0.8, 0, 5)
3.	Looping	2.5	4.4	(0.8, 0, 5)	(0.2, 0, 5.3)
4.	Straight line	4.4	5.4	(0.2, 0, 5.3)	(3.9, 0, 5)
5.	Eight on a sphere	5.4	7.5	(3.9, 0, 5)	(3.8, 0.1, 5)
6.	Straight line	7.5	8.6	(3.8, 0.1, 5)	(4, −5.8, 5)
7.	Spiral	8.6	16.6	(4, −5.8, 5)	(2, −5.9, 2.2)
8.	Straight line	16.6	18	(2, −5.9, 2.2)	(0, 0, 5)
9.	Straight line	18	19.85	(0, 0, 5)	(0, 0, 0)

In Figure 10, the nine path segments of the proposed circuit can be identified as follows.

1. Straight line parallel to the Z axis (take off). During the first 2 s, the violet and yellow curves are in zero.
2. Straight line parallel to the X axis. From 2 s to 2.5 s, the curves in violet and red take the zero value.
3. Looping executed on the XZ plane. The violet curve is at zero from 2.5 s to 4.4 s.
4. Straight line parallel to the X axis. From 4.4 s to 5.4 s, the violet and red curves do not present variations.
5. Eight on a sphere executed on the 3D space. From 5.4 s to 7.5 s, the three curves exhibit oscillations.
6. Straight line parallel to the Y axis. From 7.5 s to 8.6 s, the yellow and red curves remain constant.

7. Descending spiral. From 8.6 s to 16.6 s, the yellow and violet curves present the expected oscillations, while the red one is descendent.
8. Straight-line path connecting the end point of the spiral to the initial point of the second segment. From 16.6 s to 18 s, the violet, red, and yellow curves present variations.
9. Straight line parallel to the Z axis (landing). From 18 s to 19.85 s, the violet and yellow curves are in zero.

Figure 11 shows the error behavior corresponding to the three axes. Remember that, in the proposed path planning approach, the objective is not to drive the error towards zero; the approach aims at controlling the error dynamics in such a way that the evolution of the punctual mass's position allows for developing the predefined aerobatics. For this reason, the error curves have similar shapes to those corresponding to the position.

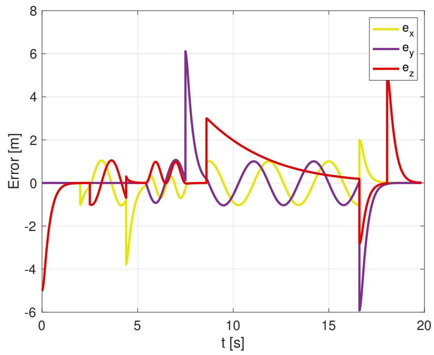

Figure 11. Error.

Figure 12 shows the whole trajectory of the punctual mass in the plane XY. The initial position is set at $(0, 0, 0)$; for takeoff, a straight line along the Z axis connects the initial point to the point $(0, 0, 5)$. Note that a straight-line path in the X axis allows for reaching the point $(3.9, 0, 5)$. In this figure, one can identify the eight on a sphere. During this maneuver, the maximum value in the Y axis is 1.07, and the minimum value is -0.92, while, in the X axis, the movement takes place between 3.5 and 4.5. After developing the eight on a sphere, a straight-line path is defined to reach the point $(4, -5.8, 5)$. At this point, the figure shows a circle representing the execution of the spiral with the center at $(3, -6, 5)$ and a unitary radius. Finally, a straight-line path allows for connecting the end point of the spiral to the initial point.

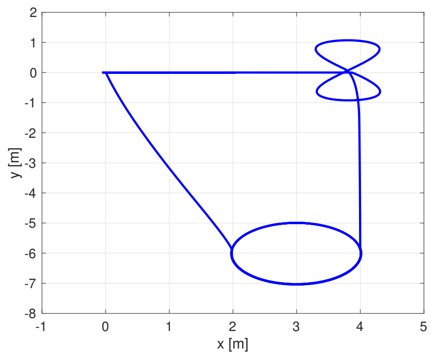

Figure 12. Position evolution (2D): XY plane.

Figure 13 shows the whole trajectory of the punctual mass in the XZ plane. For takeoff, a straight-line path segment is defined along the Z axis; it connects the point $(0, 0, 0)$ to the point $(0, 0, 5)$. Then, another straight-line path allows for reaching the point $(0.8, 0, 5)$ where a circle with the center at $(1, 0, 5)$ and a unitary radius representing the looping can be identified. A straight-line path connects the points $(0.2, 0, 5.3)$ and $(3.9, 0, 5)$. Here, another circle with the center at $(3.9, 0, 5.5)$ and a radius of 0.5 representing the eight on a sphere is shown. The path descends to the point $(2, -5.9, 2.2)$ through a wavy path segment corresponding to the spiral. A straight-line trajectory allows for connecting the point $(2, -5.9, 2.2)$ with the one fixed at $(0, 0, 5)$. Finally, a straight-line path is set to reach the initial point $(0, 0, 0)$. The above description can also be identified in Figures 14 and 15, where the whole trajectory in the 3D space is shown.

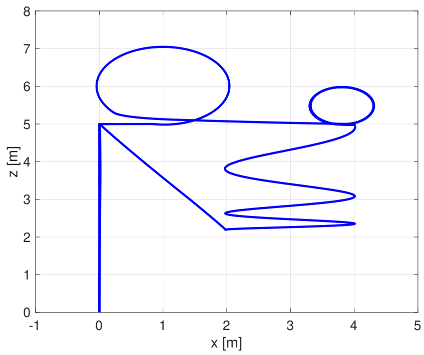

Figure 13. Position evolution (2D): XZ plane.

Animation videos of the aerobatics circuit can be found at https://youtu.be/o4ySj6CL1Sgandhttps://youtu.be/K8EM7LjIL2c (accessed on 20 February 2024).

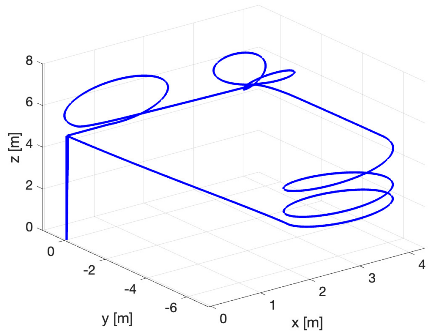

Figure 14. Position evolution (3D).

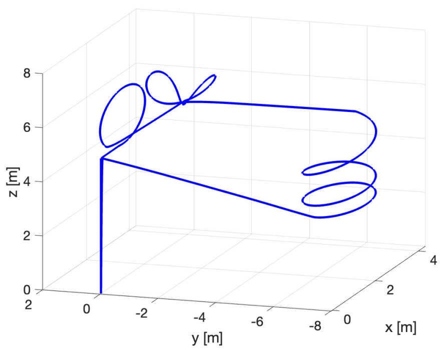

Figure 15. Position evolution (3D).

Figure 16 shows the behavior of the control signals. Note that the magnitude of the controllers is increased when the difference between the current position and the reference position is significant.

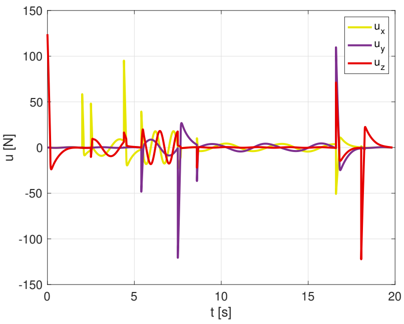

Figure 16. Sliding mode control signals.

6. Conclusions

This paper proposes a novel path planning approach to develop aerobatics that takes advantage of specific mathematical structures. The Hopf bifurcation properties are exploited to generate a stable limit cycle that allows for developing the looping and spiral. Viviani's curve is parameterized to generate the eight on a sphere maneuver. Straight-line path segments are generated by using the properties of the logarithm function.

The proposed path planning approach is based on the construction of different types of nonlinear SMSs. Following this design approach, different 3D parameterized curves can be transformed to an SMS to develop the path planning for executing other maneuvers.

To guarantee the execution of aerobatics, the proposed approach requires the convergence of the surfaces to zero. To this end, the use of STVSMC is proposed, and a stability analysis is developed. A Lyapunov-based practical stability analysis that allows for handling perturbations bounded in norm is proposed. The descriptor method is exploited here to reduce the inherent conservatism of the obtained conditions in terms of matrix inequalities. Once stability is guaranteed, STVSMC allows for reaching in finite time the predefined set of paths to perform the considered aerobatics. Numerical simulations were presented to illustrate the execution of aerobatics.

In order to apply the proposed method to execute any other predefined maneuver, the general process can be summarized as follows:

1. Define the circuit of 3D trajectories to be executed.
2. Characterize each path segment in terms of equations defining its geometry in the 3D space.
3. Based on these equations, define a sliding mode surface **S** representing the desired behavior of the system.
4. Write each surface in the general form given in Equation (38).
5. Use Proposition 1 to determine the controller gains.
6. Use a switching control of the form (73) and (74) to connect the path segments.

The proposed path planning method requires building up all of the fixed surfaces with their corresponding parameters, as well as the control gains, before the flight, that is, offline. Nevertheless, it could be possible to change certain parameters online, such as the shapes' size, speed, direction, and initial position of the aerobatics.

The proposed path planning strategy presents the following limitations: the vehicle orientation is not considered, the bound on the disturbance is required to be known, and the presence of in-flight failures and objects obstructing the path of the vehicle is not considered.

As a direction for future work, a comprehensive mathematical model of a UAV (quadrotor), including the position and orientation dynamics, will be considered. Then, the proposed approach will be extended to control also the vehicle's orientation using the geometric approach. Another research line that will also be explored is collision avoidance through the combination of the proposed path planning approach with other classical strategies, such as the conventional potential field method.

Author Contributions: Conceptualization, B.S. and M.J.-L.; methodology, O.C., B.S. and M.J.-L.; software, O.C.; validation, O.C., B.S. and M.J.-L.; formal analysis, O.C., B.S. and M.J.-L.; investigation, O.C., B.S., M.J.-L., J.C.Á.-V. and C.A.-I.; writing—original draft preparation, M.J.-L., O.C. and B.S.; writing—review and editing, M.J.-L., J.C.Á.-V. and C.A.-I.; supervision, B.S.; project administration, B.S. and M.J.-L.; funding acquisition, B.S. and C.A.-I. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by CONACYT grant number 1028331 and Asociación Universitaria Iberoamericana de Postgrado (AUIP). B.S. gratefully acknowledges financial support from the Consejo Nacional de Ciencia y Tecnología (CONACYT) provided in 2023 (CF-2023-I-722). C.A.-I. express his gratitude to the Instituto Politécnico Nacional of Mexico for the financial support provided in 2023 under the research grant 20231786.

Data Availability Statement: The data presented in this study are openly available in https://github.com/oleg19cg/Sliding_Surface-Based_Path_Planning_for_Unmanned_Aerial_Vehicle_Aerobatics.git at 10.5281/zenodo.7728226.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

UAVs	unmanned aerial vehicles
SMS	nonlinear sliding mode surface
SMC	sliding mode control
LMI	linear matrix inequality
STVSMC	super twisting vector sliding mode control
HOSMC	higher-order sliding mode control
STPIDSMC	super twisting proportional-integral-derivative sliding mode controller

References

1. FAA. *Airplane Flying Handbook: FAA-H-8083-3C*; Federal Aviation Administration: Washington, DC, USA, 2021.
2. Zhao, S.; Wang, X.; Zhang, D.; Shen, L. Curved path following control for fixed-wing unmanned aerial vehicles with control constraint. *J. Intell. Robot. Syst.* **2018**, *89*, 107–119. [CrossRef]
3. Yujie, S.; Shenmin, S. Three-dimensional adaptive finite-time guidance law for intercepting maneuvering targets. *Chin. J. Aeronaut.* **2017**, *30*, 1985–2003.
4. Rubí, B.; Pérez, R.; Morcego, B. A survey of path following control strategies for UAVs focused on quadrotors. *J. Intell. Robot. Syst.* **2020**, *98*, 241–265. [CrossRef]
5. Zhang, H.Y.; Lin, W.M.; Chen, A.X. Path planning for the mobile robot: A review. *Symmetry* **2018**, *10*, 450. [CrossRef]
6. Karur, K.; Sharma, N.; Dharmatti, C.; Siegel, J.E. A survey of path planning algorithms for mobile robots. *Vehicles* **2021**, *3*, 448–468. [CrossRef]
7. Aggarwal, S.; Kumar, N. Path planning techniques for unmanned aerial vehicles: A review, solutions, and challenges. *Comput. Commun.* **2020**, *149*, 270–299. [CrossRef]
8. Eslamiat, H.; Li, Y.; Wang, N.; Sanyal, A.K.; Qiu, Q. Autonomous waypoint planning, optimal trajectory generation and nonlinear tracking control for multi-rotor uavs. In Proceedings of the 2019 18th European Control Conference (ECC), Naples, Italy, 25–28 June 2019; pp. 2695–2700.
9. Li, Y.; Eslamiat, H.; Wang, N.; Zhao, Z.; Sanyal, A.K.; Qiu, Q. Autonomous waypoints planning and trajectory generation for multi-rotor UAVs. In Proceedings of the Workshop on Design Automation for CPS and IoT, Montreal, QC, Canada, 15 April 2019; pp. 31–40.
10. Rasche, C.; Stern, C.; Kleinjohann, L.; Kleinjohann, B. A 3d path planning approach extended by bifurcation theory for formation flights. In *Recent Advances in Robotics and Automation*; CRC Press: Boca Raton, FL, USA, 2013; pp. 103–113.
11. Woo, J.W.; An, J.Y.; Cho, M.G.; Kim, C.J. Integration of path planning, trajectory generation and trajectory tracking control for aircraft mission autonomy. *Aerosp. Sci. Technol.* **2021**, *118*, 107014. [CrossRef]
12. Brown, A.; Anderson, D. Trajectory Optimization for High-Altitude Long-Endurance UAV Maritime Radar Surveillance. *IEEE Trans. Aerosp. Electron. Syst.* **2020**, *56*, 2406–2421. [CrossRef]
13. Saadi, A.A.; Soukane, A.; Meraihi, Y.; Gabis, A.B.; Mirjalili, S.; Ramdane-Cherif, A. UAV Path Planning Using Optimization Approaches: A Survey. *Arch. Comput. Methods Eng.* **2022**, *29*, 4233–4284. [CrossRef]
14. Nekoo, S.R.; José Ángel, A.; Anibal, O. Quaternion-based state-dependent differential Riccati equation for quadrotor drones: Regulation control problem in aerobatic flight. In *Robotica*; Cambridge University Press: Cambridge, UK, **2022**; pp. 1–16.

15. Bulka, E.; Nahon, M. Automatic control for aerobatic maneuvering of agile fixed-wing UAVs. *J. Intell. Robot. Syst.* **2019**, *93*, 85–100. [CrossRef]
16. Yu, G.; Cabecinhas, D.; Cunha, R.; Silvestre, C. Quadrotor trajectory generation and tracking for aggressive maneuvers with attitude constraints. *IFAC-PapersOnLine* **2019**, *52*, 55–60. [CrossRef]
17. Ríos, H.; Falcón, R.; González, O.A.; Dzul, A. Continuous sliding-mode control strategies for quadrotor robust tracking: Real-time application. *IEEE Trans. Ind. Electron.* **2018**, *66*, 1264–1272. [CrossRef]
18. Wang, J.; Ma, X.; Li, H.; Tian, B. Self-triggered sliding mode control for distributed formation of multiple quadrotors. *J. Frankl. Inst.* **2020**, *357*, 12223–12240. [CrossRef]
19. González, J.A.C.; Salas-Peña, O.; De León-Morales, J. Observer-based super twisting design: A comparative study on quadrotor altitude control. *ISA Trans.* **2021**, *109*, 307–314. [CrossRef] [PubMed]
20. Utkin, V.; Poznyak, A.; Orlov, Y.V.; Polyakov, A. *Road Map for Sliding Mode Control Design*; Springer: Berlin/Heidelberg, Germany, 2020.
21. Shtessel, Y.; Edwards, C.; Fridman, L.; Levant, A. *Sliding Mode Control and Observation*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2014.
22. Nagesh, I.; Edwards, C. A multivariable super-twisting sliding mode approach. *Automatica* **2014**, *50*, 984–988. [CrossRef]
23. López-Caamal, F.; Moreno, J.A. Generalised multivariable supertwisting algorithm. *Int. J. Robust Nonlinear Control* **2019**, *29*, 634–660. [CrossRef]
24. Moreno, J.A.; Ríos, H.; Ovalle, L.; Fridman, L. Multivariable super-twisting algorithm for systems with uncertain input matrix and perturbations. *IEEE Trans. Autom. Control* **2021**, *67*, 6716–6722. [CrossRef]
25. Labbadi, M.; Boukal, Y.; Cherkaoui, M. Path following control of quadrotor UAV with continuous fractional-order super twisting sliding mode. *J. Intell. Robot. Syst.* **2020**, *100*, 1429–1451. [CrossRef]
26. Fethalla, N.; Saad, M.; Michalska, H.; Ghommam, J. Robust observer-based dynamic sliding mode controller for a quadrotor UAV. *IEEE Access* **2018**, *6*, 45846–45859. [CrossRef]
27. Jiménez, E.I.; Jiménez-Lizárraga, M. Robust tracking-surveillance and landing over a mobile target by quasi-integral-sliding mode and Hopf bifurcation. *J. Frankl. Inst.* **2022**, *359*, 2120–2155. [CrossRef]
28. Ramsey, J.A.; Ratliff, R.T.; Wise, K.A.; Lavretsky, E. Departure resilient control for autonomous air vehicles. In Proceedings of the 2009 American Control Conference, St. Louis, MI, USA, 10–12 June 2009; pp. 5386–5391.
29. Dong, X.; Zhou, Y.; Ren, Z.; Zhong, Y. Time-varying formation control for unmanned aerial vehicles with switching interaction topologies. *Control Eng. Pract.* **2016**, *46*, 26–36. [CrossRef]
30. Wang, X.; Yadav, V.; Balakrishnan, S. Cooperative UAV formation flying with obstacle/collision avoidance. *IEEE Trans. Control Syst. Technol.* **2007**, *15*, 672–679. [CrossRef]
31. Tan, S.C.; Lai, Y.M.; Tse, C.K. *Sliding Mode Control of Switching Power Converters: Techniques and Implementation*; CRC Press: Boca Raton, FL, USA, 2018.
32. Dong, H.; Yang, X.; Basin, M.V. Practical Tracking of Permanent Magnet Linear Motor Via Logarithmic Sliding Mode Control. *IEEE/Asme Trans. Mechatron.* **2022**, 27, 4112–4121. [CrossRef]
33. Marsden, J.E.; McCracken, M. *The Hopf Bifurcation and Its Applications*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2012; Volume 19.
34. Jordan, D.; Smith, P.; Smith, P. *Nonlinear Ordinary Differential Equations: An Introduction for Scientists and Engineers*; Oxford University Press on Demand: Oxford, UK, 2007; Volume 10.
35. Putnam, L.; Todd, S.; Latham, W. Abstract shape synthesis from linear combinations of clelia curves. In Proceedings of the 8th ACM/EG Expressive Symposium. The Eurographics Association, Genoa, Italy, 5–6 May 2019; pp. 87–99.
36. Castro, I.; Castro-Infantes, I.; Castro-Infantes, J. Spherical curves whose curvature depends on distance to a great circle. In *New Trends in Geometric Analysis: Spanish Network of Geometric Analysis 2007–2021*; Springer: Berlin/Heidelberg, Germany, 2023; pp. 19–42.
37. Fridman, E. *Introduction to Time-Delay Systems: Analysis and Control*; Springer: Berlin/Heidelberg, Germany, 2014.
38. Villafuerte, R.; Mondié, S.; Poznyak, A. Practical stability of time delay systems: LMI's approach. In Proceedings of the 2008 47th IEEE Conference on Decision and Control, Cancun, Mexico, 9–11 December 2008; pp. 4807–4812.
39. Benabdallah, A.; Ellouze, I.; Hammami, M. Practical stability of nonlinear time-varying cascade systems. *J. Dyn. Control Syst.* **2009**, *15*, 45–62. [CrossRef]