The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

A Path Planning Algorithm for UAVs with Limited Climb Angle

Armando A. Neto and Mario F. M. Campos

Abstract— **In this paper we present a methodology based on a variation of the Spatial Pythagorean Hodograph curves to generate smooth feasible paths for autonomous vehicles in three-dimensional space under the restriction of limited climb angles. An fast iterative algorithm is used to calculate the curve. The generated path satisfy three main angular constraints given by the vehicle: (i) maximum curvature, (ii) maximum torsion and (iii) maximum climb (or dive). A path is considered feasible if these kinematic constraints are not violated. The smoothness vehicle's acceleration profile is indirectly guaranteed between two points. The proposed methodology is applicable to vehicles that move in three-dimensional environments, and that can be modeled by the constraints considered here. We show results for small aerial vehicle.**

I. INTRODUCTION

There are still several open issues in the task of making autonomous vehicles going through three-dimensional environments. If a robot wants to go from one location to another, there is a basic question that it should answer before starts its navigation: "How do I get there?". It might be possible for such robots to traverse environments in a reactive way, but the generation of viable paths is an important feature for a large number of robotics tasks.

Mobile robots always present some type of motion constraint that must be solved by the path planning algorithm. Fixed-wings aerial vehicles, for example, present dynamic behaviors where variables of spatial position and orientation are completely interdependent, which impose several holonomic and nonholonomic constraints to the system. These constraints are embedded, for example, in the maximum values of lateral acceleration that can be imposed by those vehicles, which can be translated by the minimum value of curvature radius that the vehicle can describe in space.

One of the main motion constraints of a vehicle moving in 3D space is the climb (or dive) angle. It basically refers to the rate of change in altitude, which may be severely limited for some types of vehicles. Some aircrafts, por example, have a very small angle of attack (the angle that the chord of the wing, viewed laterally, makes with respect to the wind) that is one way of describing the climb angle, which is often limited by the control action of the navigation. Some underwater vehicles exhibit climb angles which are restricted by the configuration of their actuators and control surfaces.

We propose a path planning algorithm that takes into account three major motion constraints in three-dimensional space: maximum curvature, maximum torsion and maximum climb angle, but with special emphasis on the cases where climb (or dive) rates are limited. For the special case of fixed-wing aircrafts, the goal is to guarantee stall-free maneuvers. The general idea behind the method is to model the path as a 5th order spatial Bézier curve which is iteratively computed. This path will satisfy the required constraints and, at the same time, will produce curve with a satisfactory length.

The authors are with the Vision and Robotics Laboratory (Verlab), Computer Science Department, Federal University of Minas Gerais, Brazil. e-mails: {aaneto,mario}@dcc.ufmg.br

II. RELATED WORKS

One of the most important factors for path planning is to produce paths that are feasible to be executed by the vehicle, which means that during path generation, the movement restrictions of the vehicle must be considered (e.g. nonholonomic constraints). This problem has been thoroughly studied, and the literature available in the area abounds specially for manipulators and two dimensional mobile robots [1]. However, there are fewer works dealing with vehicles that displace in the three-dimensional space. Furthermore, many new challenges are posed despite the fact that, in principle, it seems that the problem is less restrictive. Current problems, which include path planning for multiple UAVs (Unmanned Aerial Vehicles) and AUVs (Autonomous Underwater Vehicles), still demand better solutions.

In this paper we use a special technique to calcule feasible paths, called Pythagorean Hodograph Interpolation. The PH curves of fifth order were presented for the first time by Farouki and Sakkalis [2], for the two-dimensional case. A Hermite Interpolation algorithm was proposed in [3], where the author demonstrate that there exist four possible solutions for the curve in $\mathbb{R}^2$. The chosen solution is the one that minimizes the cost function (bending energy function) based in the integral of the modulus a curvature function (the torsion in this in case is null).

The three-dimensional case is presented by Farouki et al. in [4]. In [5], the quaternion representation is used to deal with the Hermite Interpolation issues, and the author claims that the infinite cardinality of the set of solutions for the problem is due to an underdetermined system of equations needed to compute the final curve. In order to significantly reduce the solution space, the author suggests assigning a small set of values to the unknown variables, reducing the number of possible solutions to some. The best solution is obtained from the minimization of the cost function defined in [4], that is based on the integral of the sum of the curvature and torsion functions modulus of each curve.

To guarantee that the PH does note violate the kinematic constraints of a vehicle, Shanmugavel et al. [6] proposes a modification where the PH curve is computed iteratively. For every step of the algorithm, gain values are increased until $\vec{r}(t)$ becomes realizable. However, only the curvature and

torsion constraints are considered. As it will be shown later, for vehicles with bounded values of climb (or dive) angle, it is not possible to minimize the problem for the spatial cost functions.

In [7] the authors present an improved cost function based on [4] that consider all three constraints. In this paper, we propose a new cost function that will take into account just the climb angle limitation, indirectly ensuring the others constraints, and providing a decrease in the computational cost of the method. The θ_{max} constraint will be included in the PH computation, which will then generate feasible paths for robots in the three-dimensional space, under the aforementioned constraints.

III. PROBLEM STATEMENT

Our technique assumes an obstacle free environment, and that the only limitations for the navigation of the robot are imposed only by its own kinematic constraints. Two configurations, $\mathbf{P}_i$ and $\mathbf{P}_f$ mark the initial and final poses, respectively, which define the position and (partially) the orientation of the robot in the extreme points of the path.

A path may be defined mathematically as a parametric curve $\vec{r}(t)$ in three-dimensional space, where t is a parameter that continuously varies in $\mathbb{R}$. In this manner, the path planning problem can formally be described as:

$$\begin{array}{rcl} \mathbf{P}_i(x_i, y_i, z_i, \psi_i, \theta_i) & = & \vec{r}(t_i), \\ \mathbf{P}_f(x_f, y_f, z_f, \psi_f, \theta_f) & = & \vec{r}(t_f), \end{array} \quad (1)$$

where t_i and t_f are the initial and final values, respectively, for the curve parameter t.

Each waypoint is described by three position (x, y, z) and two orientation (ψ, θ) variables. The variable ψ is an angle that describes the way-point orientation parallel to the XY plane in relation to the $\mathbf{X}$ axis. Now θ corresponds to the way-point orientation measured parallel to XZ the plane, also in relation to the $\mathbf{X}$ axis.

The poses $\mathbf{P}_i$ and $\mathbf{P}_f$ can represent any pair of way-points in a set, which in turn, is determined by the missions planning modules in a high-level.

A. Constraints

In order for a path to be considered feasible for a given robot, the curve $\vec{r}(t)$ must simultaneously fulfill kinematic and dynamic constraints and their maximum numerical values. The three motion constraints mentioned before are the maximum curvature (κ_{max}), the maximum torsion (τ_{max}) and the maximum climb (or dive) angles (θ_{max}) realizable by the robot in 3D space. It is possible to completely define a curve in $\mathbb{R}^3$ only by means of its functions of curvature and torsion [8].

As far as the physics is concerned, the curvature may be defined as a quantity that is directly proportional to the lateral acceleration of the robot in space. The value of κ_{max} is inversely proportional to the minimum curvature radius (ρ_{min}) of the curve that the vehicle is able to execute, which is also proportional to the maximum lateral acceleration of the vehicle. The curvature function of a curve in the n-dimensional space is given by the following equation:

$$\kappa(t) = \frac{|\dot{\vec{r}}(t) \times \ddot{\vec{r}}(t)|}{|\dot{\vec{r}}(t)|^3}. \quad (2)$$

The torsion can be seen as being directly proportional to the angular moment (roll moment) of the robot, which is also physically limited. Thus, the value of τ_{max} is given in function of the minimum torsion radius (σ_{min}) that the robot describes in space. The torsion of a curve can be calculated through the following equation:

$$\tau(t) = \frac{[\dot{\vec{r}}(t) \times \ddot{\vec{r}}(t)] \cdot \dddot{\vec{r}}(t)}{|\dot{\vec{r}}(t) \times \ddot{\vec{r}}(t)|^2}. \quad (3)$$

Finally, the climb (or dive) angle is proportional to the ascent (or descent) rate of the robot in 3D space. In other words, it captures the variation of the altitude (z) throughout the path. For vehicles with limited values of climb angle, such as fixed-wing aircrafts, this is a fundamental variable. The value of θ_{max} may depend on many factors, as translation speed and spatial orientation of the robot. The climb angle function of a parametric curve in three-dimensions is given by:

$$\theta(t) = \tan^{-1}\left(\frac{\dot{z}(t)}{\sqrt{\dot{x}(t)^2 + \dot{y}(t)^2}}\right). \quad (4)$$

It is possible to show that this function is mathematically confined to the interval $\left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$. The same is valid for the value of θ_{max}.

Finally, the path $\vec{r}(t)$ is valid for a vehicle if the modulus of curvature, torsion and climb angle functions, are smaller than the the vehicles absolute maximum values, as described below:

$$\vec{r}(t), \quad |\kappa(t)| < \kappa_{max}, \ |\tau(t)| < \tau_{max}, \ |\theta(t)| < \theta_{max}. \quad (5)$$

With regard to the dynamic, it is important to consider the form by which such constraints vary in the time. The continuity of the curvature, torsion and climb angle functions is another fundamental characteristic in the path planning for real vehicles. Discontinuities in the curvature function, for example, can induce infinite variations of lateral acceleration, which of course, are not realizable. The same reasoning is valid for the torsion. In the case of the climb angle function, the lack of continuity implies in the tangential discontinuity of the curve itself.

Finally, the curve produced by the path planning algorithm is continuously derivable, should be third order differentiable, according to the Equation 3.

B. Spatial Pythagorean Hodograph Curves

Spatial Pythagorean Hodograph curves are a special kind of parametric polynomial curves defined in the three-dimensional space. They are represented, in general, as $\vec{r}(t) = [x(t), y(t), z(t)]$ and their derivatives (hodograph components) satisfy the Pythagorean condition:

$$\dot{x}(t)^2 + \dot{y}(t)^2 + \dot{z}(t)^2 = h(t)^2 \quad (6)$$

for some polinomial $h(t)$. This means that the parametric "speed", $\dot{s}(t)$, of the curve can be described by means of a polynomial, making it possible to compute the length of the path, s, exactly as:

$$s = \int_{t_i}^{t_f} \dot{\vec{r}}(t)\, dt = \int_0^1 |h(t)|\, dt \tag{7}$$

The spatial PH curves are still shaped as fifth order Bézier curves:

$$\vec{r}(t) = \sum_{k=0}^{5} \mathbf{p}_k \binom{5}{k} (1-t)^{5-k} t^k; \quad t \in [0,1] \tag{8}$$

where $\mathbf{p}_k = [x_k,\, y_k,\, z_k]$ is the k-th control point of the Bézier curve. The path planning problem is then reduced to find a solution to the Hermite Interpolation problem. One important advantage of using this model is that the resulting curve is infinitely continuous, so that the curvature, torsion and inclination functions are always smooth.

IV. REALIZABLE PATH CALCULATION

In this work the principles of spatial PH curves are used to generate paths that are attainable by a robot in three-dimensional space. These paths comply with the curvature, torsion and climb angle constraints imposed by this robot. Thus, assuming the model described by the Equation 8, the path planning problem is to determine the six control points of the Bézier curve.

A. First-order Hermite Interpolation Problem

To find the control points of the Bézier curve we should solve the following Hermite Interpolation system:

$$\begin{aligned}
\mathbf{p}_0 &= [x_i,\, y_i,\, z_i],\\
\mathbf{p}_1 &= \mathbf{p}_0 + \frac{c_0}{5}\mathcal{A}_0\mathbf{i}\mathcal{A}_0^*,\\
\mathbf{p}_2 &= \mathbf{p}_1 + \frac{c_0}{10}(\mathcal{A}_0\mathbf{i}\mathcal{A}_1^* + \mathcal{A}_1\mathbf{i}\mathcal{A}_0^*),\\
\mathbf{p}_3 &= \mathbf{p}_4 - \frac{c_5}{10}(\mathcal{A}_2\mathbf{i}\mathcal{A}_1^* + \mathcal{A}_1\mathbf{i}\mathcal{A}_2^*),\\
\mathbf{p}_4 &= \mathbf{p}_5 - \frac{c_5}{5}\mathcal{A}_2\mathbf{i}\mathcal{A}_2^*,\\
\mathbf{p}_5 &= [x_f,\, y_f,\, z_f].
\end{aligned} \tag{9}$$

where c_0 and c_5 are gain factors that has unit values for a PH with no constraints. The above set of equations was derived from [5], and according to authors, it represents an underdetermined system. In other words, there are infinity spatial PH curves that interpolate the initial and final waypoints.

The problem is initially approached by assuming that the extreme points of the curve ($\mathbf{p}_0$ and $\mathbf{p}_5$) are directly determined by the initial and final poses $\mathbf{P}_i$ and $\mathbf{P}_f$, respectively. All the remaining points will depend on these pose vectors and on the c_0 and c_5 gains.

We model the problem using quaternions, which allows for a more compact and elegant representation of the system, as well as providing a clearer geometric view of the solution. The conjugate of $\mathcal{A}_k$ is given by the notation $\mathcal{A}_k^*$. As shown in [4], a set of equations of the type

$$\mathcal{A}_k\mathbf{i}\mathcal{A}_k^* = \mathbf{c}\,, \tag{10}$$

where $\mathbf{i}$ is a pure vector quaternion since its scalar part is null, such that $\mathbf{i}^2 = -1$; presents a solution $\mathbf{c}$ which also is a pure vector quaternion.

Hence, a system is formed that is composed of three equations and four unknowns, leaving a degree of freedom still to be determined. With this, the solution for every $\mathcal{A}_k$ is parameterized by an angle variable ϕ_k (for a detailed explanation the reader is referred to [4]). Deriving the Equation 10 with respect to $\mathcal{A}_0$, one obtains:

$$\mathcal{A}_0(\phi_0) = \sqrt{\frac{|\mathbf{d}_i|}{2}(1+\lambda_i)} \begin{bmatrix} -\sin(\phi_0) \\ \cos(\phi_0) \\ \dfrac{\mu_i\cos(\phi_0) + \nu_i\sin(\phi_0)}{1+\lambda_i} \\ \dfrac{\nu_i\cos(\phi_0) - \mu_i\sin(\phi_0)}{1+\lambda_i} \end{bmatrix}, \tag{11}$$

where $\mathbf{d}_i$ is the vetor direction of the pose $\mathbf{P}_i$, computed as

$$\mathbf{d}_i = c_0\ [\cos(\psi_i)\cos(\theta_i),\, \sin(\psi_i)\cos(\theta_i),\, \sin(\theta_i)]$$

and

$$[\lambda_i,\, \mu_i,\, \nu_i] = \mathbf{d}_i/\|\mathbf{d}_i\|\,.$$

Equivalently, $\mathcal{A}_2$ can be expressed in the form:

$$\mathcal{A}_2(\phi_2) = \sqrt{\frac{|\mathbf{d}_f|}{2}(1+\lambda_f)} \begin{bmatrix} -\sin(\phi_2) \\ \cos(\phi_2) \\ \dfrac{\mu_f\cos(\phi_2) + \nu_f\sin(\phi_2)}{1+\lambda_f} \\ \dfrac{\nu_f\cos(\phi_2) - \mu_f\sin(\phi_2)}{1+\lambda_f} \end{bmatrix}, \tag{12}$$

where $\mathbf{d}_f$ is the vetor direction of the pose $\mathbf{P}_f$, given by

$$\mathbf{d}_f = c_5\ [\cos(\psi_f)\cos(\theta_f),\, \sin(\psi_f)\cos(\theta_f),\, \sin(\theta_f)]$$

and

$$[\lambda_f,\, \mu_f,\, \nu_f] = \mathbf{d}_f/\|\mathbf{d}_f\|\,.$$

Once the values of $\mathcal{A}_0$ and $\mathcal{A}_2$ are determinated by setting the values of ϕ_0 and ϕ_2, respectively, $\mathcal{A}_1$ can be defined as function of a third angle variable ϕ_1:

$$\mathcal{A}_1(\phi_1) = -\frac{3}{4}(\mathcal{A}_0 + \mathcal{A}_2) + \frac{\sqrt{\dfrac{|\mathbf{c}|}{2}(1+\lambda)}}{4}\mathcal{A}_c(\phi_1), \tag{13}$$

where

$$\mathcal{A}_c(\phi_1) = \begin{bmatrix} -\sin(\phi_1) \\ \cos(\phi_1) \\ \dfrac{\mu\cos(\phi_1) + \nu\sin(\phi_1)}{1+\lambda} \\ \dfrac{\nu\cos(\phi_1) - \mu\sin(\phi_1)}{1+\lambda} \end{bmatrix},$$

$\mathbf{c}$ is a pure vector quaternion given by

$$\mathbf{c} = 120(\mathbf{p}_f - \mathbf{p}_i) - 15(\mathbf{d}_i + \mathbf{d}_f) + 5(\mathcal{A}_0\mathbf{i}\mathcal{A}_2^* + \mathcal{A}_2\mathbf{i}\mathcal{A}_0^*) \tag{14}$$

and

$$[\lambda,\, \mu,\, \nu] = \frac{\mathbf{c}}{|\mathbf{c}|}.$$

B. Iterative Algorithm

To compute the spatial PH, it is remains to determine the values of the angles ϕ_0, ϕ_1, ϕ_2 and the gains c_0 and c_5. The angular variables are defined for the range $\left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$. In [4], authors claim (empirically) that it is possible to represent the solution space in extensive form enough, by the quantization of the values of each of these angles according to:

$$\phi_k = \left[-\frac{\pi}{2}, -\frac{\pi}{4}, 0, \frac{\pi}{4}, \frac{\pi}{2}\right].$$

The combinatorial arrangement of these five values for each ϕ_k leads to a total of one hundred and twenty-five solutions to the PH curve. The authors also argue that for most cases, ϕ_1 may be equal to $-\frac{\pi}{2}$, without loss of generality, reducing the number of solutions to twenty-five. Each of these solutions has different curvature, torsion and climb angle functions. The best solution is the one that minimizes the cost function of the path, or the bending elastic energy function [9]:

$$\mathcal{E} = \int_0^1 \omega(t)^2 \left|\dot{\vec{r}}(t)\right| dt \tag{15}$$

where $\omega(t)$ is the total curvature of the spatial PH, which is given by

$$\omega(t) = \sqrt{\kappa(t)^2 + \tau(t)^2}.$$

This solution, however, is not satisfactory for the problem considered here, since it does not take into account the climb angle function in the energy computation. As it will be shown later, the PH curves that minimizes the cost function, may present climb (or dive) angles unattainable for a given robot.

To solve this problem, the following elastic bending energy function is proposed in [7]:

$$\omega(t) = \sqrt{\left(\frac{\kappa(t)}{\kappa_{max}}\right)^2 + \left(\frac{\tau(t)}{\tau_{max}}\right)^2 + \left(\frac{\theta(t)}{\theta_{max}}\right)^2}. \tag{16}$$

Besides minimizing the increase in the rate of climb of the vehicle in three-dimensional space, this new function still takes into account the three aforementioned constraints and their maximum values in a normalized form. The problem with this function is the increase in the computational complexity in relation to the previous equation. We show in the next section that a more satisfactory result can be obtained using the following cost function:

$$\omega(t) = \left|\theta(t)\right|. \tag{17}$$

Instead of considering the curvature and torsion profiles, this new function takes into account only the aforementioned climb constraint of the vehicle, which minimizes the cost of algorithm and contributes to the smoothing of the climb rate of the curve.

We verify, experimentally, that in most cases the lowest values of $\mathcal{E}$ were obtained when $\phi_1 = -\frac{\pi}{2}$ (as seen in the previous case), and when the difference between ϕ_0 and ϕ_2 was the largest possible. Therefore, taking ϕ_0 in the ϕ_k interval and $\phi_2 = -\phi_0$, the number of solutions is reduced to a set of five only.

Finally, the values of c_0 and c_5, the gains for which the spatial PH fulfills the requirements described in Equation 5 still remains to be determined. This naturally leads to an optimization approach, since there is no closed solution for this problem. Increasing these gains tends to minimize the function $\omega(t)$ as a whole, which is indirectly linked to the inverse of the parametric "speed" of the curve. But that brings as a consequence an increase in path length. Thus, the ideal values of c_0 and c_5 are those which produces a feasible curve for a given robot, but also minimizes s.

In the first step of the algorithm, the values of these gain factors are unit. In each iteration of the algorithm the conditions 5 are observed, and gains are increased by

$$c_0 = c_0 + \left(\frac{\mathcal{E}_0}{\mathcal{E}_0 + \mathcal{E}_5}\right)(\rho_{min} + \sigma_{min}), \tag{18}$$

$$c_5 = c_5 + \left(\frac{\mathcal{E}_5}{\mathcal{E}_0 + \mathcal{E}_5}\right)(\rho_{min} + \sigma_{min}), \tag{19}$$

where

$$\mathcal{E}_0 = \int_0^{0.5} \omega(t)^2 \left|\dot{\vec{r}}(t)\right| dt$$

and

$$\mathcal{E}_5 = \int_{0.5}^1 \omega(t)^2 \left|\dot{\vec{r}}(t)\right| dt.$$

This method promotes a convergence to the result with a very small number of iterations, producing a curve of reasonable length when compared to other techniques, such as RRT [10].

V. RESULTS

The proposed methodology has been used to generate paths in three-dimensional space for an autonomous vehicle modeled by the following constraints:

- $\rho_{min} = 10$ meters,
- $\sigma_{min} = 100$ meters,
- $\theta_{max} = \frac{\pi}{6}$ radians.

These values were chosen arbitrarily, so that the vehicle presents a relatively small climb angle (30 degrees) and minimum torsion radius ten times larger than the minimum radius of curvature.

We have also established two spatial configurations representing the initial and final way-points of the path to be planned, where position variables units are in meters and the orientation in radians:

- $\mathbf{P}_i = \left[0, 0, 0, -\frac{\pi}{2}, \frac{\pi}{6}\right]$,
- $\mathbf{P}_f = \left[50, 20, 50, -\frac{\pi}{2}, 0\right]$.

A very important point is to ensure that the values of the climb angles θ_i and θ_f of these way-points are smaller in modulus than the maximum climb angle. Otherwise, it becomes impossible to find values of c_0 and c_5 that guarantees the condition 5 for $\vec{r}(t)$.

Next two types of paths for those configurations pairs were generated. The first use of the methodology of Spatial PH

curves, is shown in [4]. As seen, this technique only minimizes the curvature and torsion functions of $\vec{r}(t)$. The second method applies the proposed energy function (Equations 15 and 17) to minimize the three constraints considered.

Figures 1(a) and 1(b) show a comparison between the results produced by these two techniques, respectively. It is possible to see that the curve without climb constraint is incapable of being accomplished by the robot, even though it presents a length smaller than the second case.

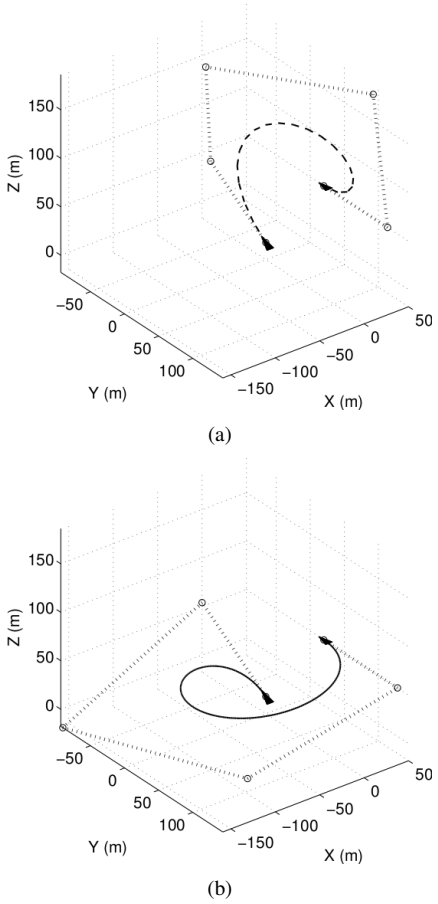

Fig. 1. Comparison between (a) Pythagorean Hodograph curve and (b) Pythagorean Hodograph curve with bounded climb angle for $\mathbf{P}_i = \left[0, 0, 0, -\frac{\pi}{2}, \frac{\pi}{6}\right]$, $\mathbf{P}_f = \left[50, 20, 50, -\frac{\pi}{2}, 0\right]$, $\rho_{min} = 10$ meters, $\sigma_{min} = 100$ meters and $\theta_{max} = \frac{\pi}{6}$ radians.

Besides the paths produced, it is possible to see the configuration of the control points calculated for each curve. These points were calculated from the values $\phi_0 = \frac{\pi}{2}$, $\phi_1 = 0$ and $\phi_2 = -\frac{\pi}{2}$ for the conventional PH and $\phi_0 = \frac{\pi}{2}$, $\phi_1 = -\frac{\pi}{2}$ and $\phi_2 = -\frac{\pi}{2}$ for the PH with bounded climb angle.

In the Figure 2 is possible to see the comparison of the two curves in relation to the constraint functions of the path. Both curvature (Figure 2(a)) and torsion (Figure 2(b)) functions for both methods are continuous and bounded by the maximum κ_{max} and τ_{max} values, respectively. The main difference between them is the climb angle function (Figure 2(c)), where it realizes that the first method which generates curves that pass over the value of θ_{max} in modulus. In certain respects, the climb comes close to extremes of the range $\left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$.

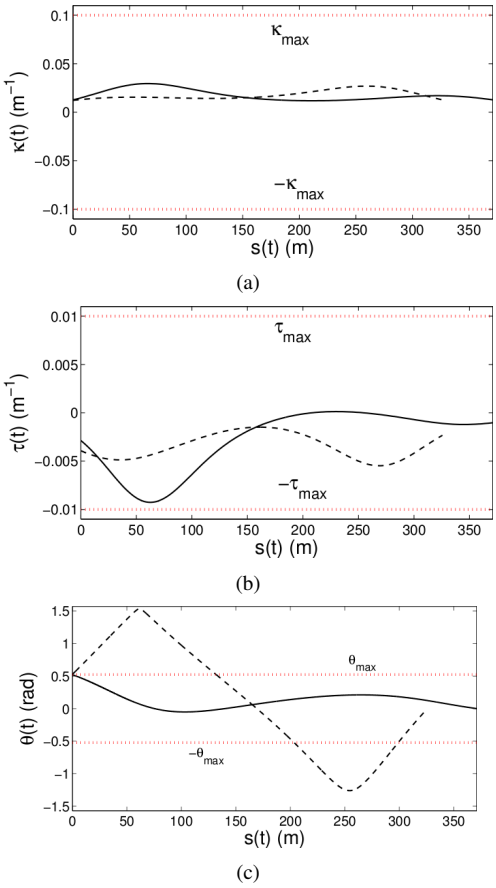

Fig. 2. Constraint functions of (a) curvature, (b) torsion and (c) climb angle of Spatial PH (dashed black line) and Spatial PH curve with bounded climb angle (continuous black line).

The proposed methodology limits values of the climb angle to the range of θ_{max} throughout the path. But that is at the expense of increasing the length of the path, as can be seen in Figure 2. In fact, higher values of c_0 and c_5 are necessary if the curve reach the three constraints as compared to the first technique, where only two constraints are used.

The technique was used to plan paths for a small unmanned fixed-wing aircraft called AqVS (Figure 3), developed at Universidade Federal de Minas Gerais. This is a small hand launched hybrid electric motor sail plane, equipped with gps receptor, barometric altimeter, infrared inclinometer, airspeed sensor and CCD camera, and controlled by a set of PID stabilizators for autonomous navigation (for more informations view [11]). The AqVS presents the following characteristics:

- $\rho_{min} = 50$ meters,
- $\sigma_{min} = 300$ meters,
- $\theta_{max} = \frac{\pi}{30}$ radians.

The above values were determined using data from actual flights of the SUAV, considering a speed of approximately 50

km/h. This vehicle has shown to be a good choice for testing our methodology because of its small climb angle (about 6 degrees).

Fig. 3. AqVS-SUAV from Universidade Federal de Minas Gerais/Brazil.

Figure 4 shows the result of two simulated flights of the AqVS/SUAV. In the first test we use the Spatial Pythagorean Hodograph methodology to generate paths to our vehicle. In the second test we use our methodology. The following five points were arbitrarily chosen and used to produce these paths:

- $\mathbf{P}_1 = [0, 0, 1000, 0, 0]$,
- $\mathbf{P}_2 = \left[1500, 0, 1050, -\frac{\pi}{4}, \frac{\pi}{30}\right]$,
- $\mathbf{P}_3 = \left[1500, 2000, 1100, \frac{\pi}{2}, 0\right]$,
- $\mathbf{P}_4 = \left[2500, 500, 1050, \pi, -\frac{\pi}{30}\right]$,
- $\mathbf{P}_5 = \left[0, 200, 1000, -\frac{\pi}{2}, 0\right]$.

The blue line in the graphs represents the actual trajectory executed by the SUAV model when tracking the planned paths. We can see in the Figure 4(a) that the aircraft have some difficulties to execute the paths, once its clearly violate the climb constraint of the robot. In the Figure 4(b) however, the fly is satisfactory. In spite of the noise, which is mainly due to the altitude sensor and the actions of the control system, the vehicle is able to closely track the path.

VI. CONCLUSIONS AND FUTURE WORKS

In this paper we presented a methodology for path planning for autonomous vehicles that move in three-dimensional environments. These vehicles typically present at least three motion constraints: maximum curvature, maximum torsion and maximum climb angles. These kinematic constrains represent a simplification to a more complicated dynamic model, in which dynamic constraints must to be consider.

The methodology is an extension of Spatial Pythagorean Hodographs where such constraints are explicitly taken into account. The proposed methodology uses an elastic bending energy function for the resolution of the Spatial Pythagorean Hodograph that minimizes the climb angle function of the curve $\vec{r}(t)$, generating a solution that is adequate for vehicles with limited climb (or dive) angle capability.

Including other constraints to the cost function, such as maximum translation speed, is one of the next steps in this research. As further investigation we will consider environments with static and dynamic objects, which demand path replanning, and are of major relevance to the case of multiple and cooperating vehicles moving in three-dimensional space.

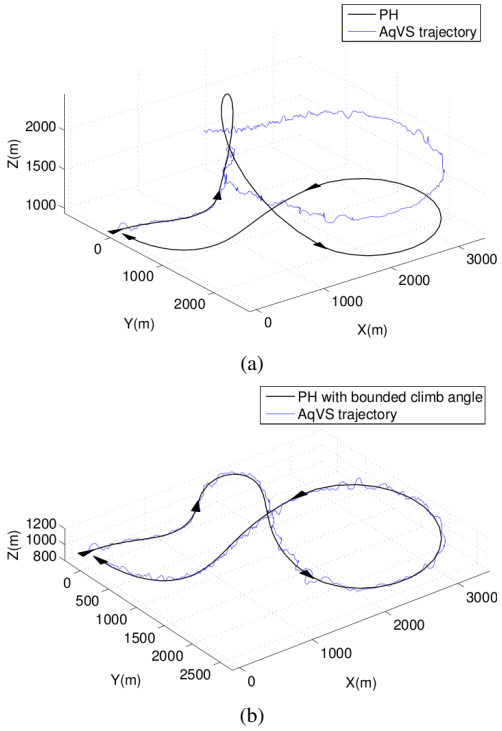

Fig. 4. Comparison between AqVS trajectory over the planned paths (a) using Spatial Pythagorean Hodograph curve and (b) using Spatial Pythagorean Hodograph curve with bounded climb angle.

VII. ACKNOWLEDGEMENTS

This work was developed with the support of CNPq, CAPES and FAPEMIG. The authors gratefully thank prof. Paulo Iscold for the flight data.

REFERENCES

[1] S. M. LaValle, *Planning Algorithms*. Urbana, Illinois, U.S.A.: Cambridge University Press, 2006.

[2] R. T. Farouki and T. Sakkalis, "Pythagorean Hodographs," *IBM Journal of Research and Development*, vol. 34, no. 5, 1990.

[3] R. T. Farouki and C. A. Neff, "Hermite Interpolation by Pythagorean Hodograph Quintics," *Mathematics of Computation*, vol. 64, pp. 1589–1609, 1995.

[4] R. T. Farouki, M. al Kandari, and T. Sakkalis, "Hermite Interpolation by Rotation-Invariant Spatial Pythagorean-Hodograph Curves," *Advances in Computational Mathematics*, vol. 17, pp. 369–383, 2002.

[5] R. T. Farouki and C. Y. Han, "Algorithms for Spatial Pythagorean-Hodograph Curves," *Geometric Properties for Incomplete Data*, pp. 43–58, 2006.

[6] M. Shanmugavel, A. Tsourdos, R. Zbikowski, and B. A. White, "3D Path Planning for Multiple UAVs using Pythagorean Hodograph Curves," Hilton Head, South Carolina, August 2007.

[7] A. Alves Neto and M. F. M. Campos, "On the Generation of Feasible Paths for Aerial Robots with Limited Climb Angle," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'09)*, Kobe, Japan, May 2009.

[8] E. Kreyszig, *Differential Geometry*. New York: Dover Publications, June 1991, vol. 1.

[9] R. T. Farouki, "The Elastic Bending Energy of Pythagorean Hodograph Curves," *Comput. Aided Geom. Design*, vol. 13, pp. 227–241, 1996.

[10] S. M. Lavalle, "Rapidly-exploring Random Trees: A New Tool For Path Planning," Computer Science Dept., Iowa State University, Tech. Rep., 1998.

[11] P. Iscold, "Development of a Small Unmanned Aerial Vehicle for Aerial Reconaiscence," in *International Congress of Mobility Engineering*, São Paulo, Brazil, 2007.