An Algebraic Collision Avoidance Approach for Unmanned Aerial Vehicle

Adriana Dapena[1], María José Souto-Salorio[2], Ana Dorotea Tarrío-Tobar[3] and Paula M. Castro[1]
[1]*Department of Computer Engineering, Universidade da Coruña, Coruña, Spain*
[2]*Department of Computer Science, Universidade da Coruña, Coruña, Spain*
[3]*Department of Mathematics, Universidade da Coruña, Coruña, Spain*

Keywords: Collision Avoidance, Control and Supervision Systems, Characteristic Polynomial, Engineering Applications, Hyperboloid Structure, Modelling and Simulation, Unmanned Aerial Vehicle.

Abstract: In this paper we address the problem of avoidancing collisions between an Unmanned Aerial Vehicle (UAV) and a rigid surface. The UAV is modelled as an unique sphere envolving the UAV or as multiple spheres only envolving the motors. The UAV flies inside a rigid hyperboloid structure typically used in architecture. The collision is detected by means of the study of the characteristic polynomial associated to quadric surfaces. Computer simulation results included in this paper will show both the accuracy and the low computational cost exhibited by the proposed method.

1 INTRODUCTION

Collision detection between two or more rigid bodies is based on determining whether any of the solids makes contact at one or more points (Brozos-Vázquez et al., 2016), (Choi et al., 2006). This problem has a great interest in many fields such as robotics (Choi et al., 2006), (Steinbach et al., 2006), computer graphics (Jia et al., 2006), animation and computer simulated environments (C. O'Sullivan, 1999), etc.

From a mathematical point of view, the contact between rigid bodies can be determined by saying that the surfaces are in contact when their intersection is not an empty set of points. Then, given two surfaces, the question is to know whether they intersect or not.

Most work in the literature devoted to the study of collision detection only considers convex bodies (Choi et al., 2006), (Jia et al., 2006). Moreover, most of the proposed algorithms are indeed valid only under that condition of convexity. Recently, in (Brozos-Vázquez et al., 2016), the authors have presented a study in which one of the bodies was not convex (in fact, that was a hyperboloid). Since hyperboloid structures are doubly ruled surfaces, this type of architectural structures is very strong against buckling and cheaper than other singly ruled surfaces, like cylinders or cones. For this reason, the hyperboloid structure is used in architecture to design cooling towers, water towers, cathedrals, air traffic control towers, and so on.

Because of its potential for commercial, military, law-enforcement, research, and other purposes, Unmanned Aerial Vehicles (UAVs) have received considerable attention in recent years. The fast development of this field has required the design of robust collision avoidance systems which are based, in general, on the combination of information acquired from several sensors, which implies processing latency and complexity (Giancarmine et al., 2008). If the surface shape changes in height (like occurs with the hyperboloid structure) those sensors must be used for the registration of this information in all directions. Collision avoidance methods based on computing the distance from the UAV to the surface are adequate in most situations where that surface shape is similar for all height positions.

The method presented in this paper is oriented to predict collisions when an UAV is flying inside a hyperboloid structure. We propose to model the UAV using multiples spheres adapted to its shape. Using the results presented in (Brozos-Vázquez et al., 2016), we propose several algorithms for collision avoidance by means of the study of the roots of a characteristic polynomial associated to quadric surfaces. In those algorithms, the conditions for collision detection are evaluated before any UAV movement so that only valid movements are allowed thus avoiding such

Dapena, A., Souto-Salorio, M., Tarrío-Tobar, A. and Castro, P.
An Algebraic Collision Avoidance Approach for Unmanned Aerial Vehicle.
DOI: 10.5220/0006410002620269
In *Proceedings of the 14th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2017) - Volume 2*, pages 262-269
ISBN: Not Available

collisions. We will show that this collision detection can be performed using a reduced number of operations.

This paper is organized as follows. Section 2 briefly introduces some ideas about the collision avoidance systems in the literature similar to our proposal. Section 3 contains the mathematical formulation of the proposed method for collision avoidance, which will be applied in Section 4 when an UAV is flying inside a hyperboloid structure for tasks of monitoring, maintenance or control. Section 5 includes the simulation results and, finally, Section 6 contains the main conclusions about this work.

2 COLLISION AVOIDANCE SYSTEMS

Collision Avoidance Systems (CASs) for UAV consist of two main functions: firstly, sensing and detection, and secondly, collision avoidance/maneuver approach. The first one refers to the way of acquiring useful information about its surrounding environment, as for example, the UAV position or its speed/heading, environment conditions... This task can be done using active and/or passive sensors and communication equipments. The second function, i.e. the collision avoidance/maneuver approach, includes mechanisms for trajectory calculation and distance estimation (Albaker and Rahim, 2009), (Pham et al., 2015). In particular, the geometric approaches perform this second function using geometry, usually by means of the simulation of the UAV trajectories and the calculation of the shortest distance to the surrounding obstacles in a 2-D space (Park et al., 2018), (Strobel and Schwarzbach, 2014). For this purpose, such geometric approaches make use of information such as location, speed or heading of the UAV with respect to obstacles, which is frequently obtained by Automatic Dependent Surveillance-Broadcast (ADS-B). Geometric approaches are specially useful when the UAV is inside simple surfaces as cylinders or spheres, in which the surface shape remains unchanged with height. However, for hyperboloid surfaces, on which we focus in this work, this property can not be assumed.

In this paper, we will consider only the collision between the UAV and a rigid surface (the hyperboloid) which represents the structure walls. For collision avoidance, an UAV could employ contour or schematic maps specially useful for indoor environments (Krokowicz et al., 2010). For regular structures (like squares or cylinders), this map is quite simple since it remains unchanged for all height. However, the hyperboloid surfaces are more complex because they change with height. For this reason, we focus on collision avoidance in the real scenario of an UAV flying inside a hyperboloid structure (like a cooling tower in a power station). The information used by the proposed method is minimal, since only requires the knowledge of the UAV position, speed and turn angle. We hope that the ideas presented in this work can be extended to different surfaces for other real applications.

3 MATHEMATICAL FORMULATION

We consider a circular hyperboloid of one sheet centred at the origin and described by the following equation

$$\mathcal{H}: \frac{x^2}{a^2} + \frac{y^2}{a^2} - \frac{z^2}{c^2} = 1, \text{for } a, c > 0. \tag{1}$$

Using the standard notation for projective spaces, we can find the matrix $\mathbf{H}$ associated to the surface $\mathcal{H}$. Let $\mathbf{x} = (x, y, z, 1)^T$ be a generic vector in homogeneous coordinates not placed at infinity. Then, $\mathbf{x}^T\mathbf{H}\mathbf{x} = 0$ defines the surface $\mathcal{H}$, where $\mathbf{H}$ is given by

$$\mathbf{H} = \begin{pmatrix} a^{-2} & 0 & 0 & 0 \\ 0 & a^{-2} & 0 & 0 \\ 0 & 0 & -c^{-2} & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}. \tag{2}$$

Moreover, we also consider a spherical surface of radius $r > 0$ whose centre is at (x_0, y_0, z_0), defined by the following equation

$$\mathcal{S}: (x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2 = r^2. \tag{3}$$

As above, we can express this equation in matrix form using homogeneous coordinates in this way

$$\mathbf{S} = \begin{pmatrix} 1 & 0 & 0 & -x_0 \\ 0 & 1 & 0 & -y_0 \\ 0 & 0 & 1 & -z_0 \\ -x_0 & -y_0 & -z_0 & -r^2 + x_0^2 + y_0^2 + z_0^2 \end{pmatrix}. \tag{4}$$

The characteristic polynomial of both surfaces $\mathcal{H}$

and $\mathcal{S}$ is a polynomial of degree 4 given by

$$f(\lambda) = \det(\lambda\mathbf{H}+\mathbf{S}) = \frac{(a^2+\lambda)g(\lambda)}{a^4c^2}, \tag{5}$$

where

$$\begin{aligned} g(\lambda) &= -(a^2+\lambda)[(c^2-\lambda)(r^2+\lambda)+z_0^2\lambda] \\ &\quad +\lambda(c^2-\lambda)(x_0^2+y_0^2) \\ &= \lambda^3+a_2\lambda^2+a_1\lambda+a_0, \end{aligned} \tag{6}$$

where a_0, a_1, and a_2 are the real coefficients of $g(\lambda)$. Therefore, we can directly conclude that one root of $f(\lambda)=0$ is $\lambda_1=-a^2$ and that the other roots are also roots of the cubic equation $g(\lambda)=0$.

3.1 Calculation of $g(\lambda)$ Roots

In this section we will show how to obtain the discriminant Δ from the cubic polynomial $g(\lambda)$, which will be used for detecting collision between the UAV and the hyperboloid surface i.e., for determining if the UAV movement is valid or not, instead of directly calculating the roots of $g(\lambda)$.

Cardano's formulas can be applied to $g(\lambda)$ for detecting contacts between $\mathcal{H}$ and $\mathcal{S}$ employing simple calculations, as described in the following.

Thus, based on the roots of $g(\lambda)$, we can determine the position of the sphere with respect to the hyperboloid (Brozos-Vázquez et al., 2016). Considering $r<a$, Table 1 shows all the cases defining the relative position between $\mathcal{S}$ and $\mathcal{H}$.

In general, for a monotonic polynomial $\lambda^3+a_2\lambda^2+a_1\lambda+a_0$, the following real parameters can be defined

$$\begin{aligned} Q &= \frac{3a_1-a_2^2}{9}, \\ R &= \frac{9a_2a_1-27a_0-2a_2^3}{54}, \\ \Delta &= Q^3+R^2. \end{aligned} \tag{7}$$

From Table 1, you can see that this parameter Δ could be used as discriminant to cluster that relative position between $\mathcal{S}$ and $\mathcal{H}$ into two groups (contact, for $\Delta \geq 0$, and non-contact, for $\Delta<0$), although it does not provide enough information about the non-contact case since $\mathcal{S}$ can be interior or exterior to $\mathcal{H}$ i.e., about if the scenario corresponds to the first row of Table 1 or to the second one. For only this purpose, we will be forced to compute the roots of $g(\lambda)$ so that their sign will allow us to distinguish both scenarios. Since all the roots are real-valued when $\Delta<0$, the following expressions will be used for their computation

$$\begin{aligned} \phi &= \frac{1}{3}\text{acos}\left(\frac{R}{\sqrt{-Q^3}}\right), \\ \lambda_2 &= 2\sqrt{-Q}\cos(\phi)-\frac{a_2}{3}, \\ \lambda_3 &= 2\sqrt{-Q}\cos\left(\phi+\frac{2\pi}{3}\right)-\frac{a_2}{3}, \\ \lambda_4 &= 2\sqrt{-Q}\cos\left(\phi+\frac{4\pi}{3}\right)-\frac{a_2}{3}, \end{aligned} \tag{8}$$

where R and Q are the parameters defined in Equation (7).

4 COLLISION AVOIDANCE

In this paper we consider UAVs with four motors, like those shown in Figure 1, although this development could be applied for any number of motors. As first approach, the UAV is modelled as one sphere with its centre in the UAV centre (see Figure 1(a)). As second approach, the UAV is modelled as multiple spheres, four in our case, centred at each of its motors (see Figure 1(b)).

4.1 UAV Model of One Sphere

Firstly, we will model the UAV according to the representation of Figure 1(a). In this figure you can see that the UAV is represented by means of one sphere $\mathcal{S}$ with radius r centred at (x_0,y_0,z_0). Considering the definition of the characteristic polynomial given in Equation (5) and the matrices of Equations (2) and (4), we can obtain the coefficients of $g(\lambda)$ of Equation (6). Thus, the coefficients of this characteristic polynomial are given by

$$\begin{aligned} a_0 &= -a^2c^2r^2, \\ a_1 &= -(a^2c^2-a^2r^2+c^2r^2-c^2x_0^2-c^2y_0^2+a^2z_0^2), \\ a_2 &= a^2-c^2+r^2-z_0^2-x_0^2-y_0^2. \end{aligned} \tag{9}$$

Since we are interested in determining when the UAV moves to a valid position (i.e. if the UAV moves in

Table 1: General relative position between S and H.

Description	Roots of $g(\lambda)$	Δ	UAV Movement
$\mathcal{S}$ interior to $\mathcal{H}$	All the roots are real and different: two negative, one positive	$\Delta < 0$	Valid
$\mathcal{S}$ exterior to $\mathcal{H}$	All the roots are positive real and different	$\Delta < 0$	Non-valid
Centre of $\mathcal{S}$ interior to $\mathcal{H}$. $\mathcal{S}$ tangent to $\mathcal{H}$.	All the roots are real, and two equal: two equal negative, one positive	$\Delta = 0$	Non-valid
Centre of $\mathcal{S}$ exterior to $\mathcal{H}$. $\mathcal{S}$ tangent to $\mathcal{H}$.	All the roots are positive real, and two equal	$\Delta = 0$	Non-valid
Intersection of $\mathcal{S}$ and $\mathcal{H}$.	One root is positive real and two are complex conjugate	$\Delta > 0$	Non-valid

the interior of $\mathcal{H}$), and taking into account the classification of Table 1, we could detect collisions from the study of the roots. Notice that the computational complexity associated to the calculation of these roots can be reduced considering that $\Delta \geq 0$ corresponds to *non-valid movements*. Moreover, if $\Delta < 0$ $\mathcal{S}$ could be interior or exterior to $\mathcal{H}$. However, with the root sign we can determine if $\mathcal{S}$ is interior to $\mathcal{H}$ or not.

The following algorithm, termed Algorithm 1, details the procedure.

Algorithm 1: One sphere.

Compute Δ.
If $\Delta < 0$
  Compute the root λ_2.
  If $\lambda_2 < 0$, **then** *valid movement (interior)*,
  otherwise
    Compute the root λ_3.
    If $\lambda_3 < 0$, **then** *valid movement (interior)*,
    otherwise *non-valid movement*.
otherwise *non-valid movement*.

4.2 UAV Model of Four Spheres

We consider in this subsection the spheres $\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3$, and $\mathcal{S}_4$, all of them with radius r_1, to model the four motors of our UAV, as shown in Figure 1(b). Considering that the UAV is centred at (x_0, y_0, z_0), the centre of each motor sphere, denoted by (x_c, y_c, z_c), can be mathematically expressed as follows

$$\begin{aligned} x_c &= x_0 + \frac{l\sqrt{2}}{2}(\cos(g) - \sin(g)) = x_0 + \frac{l\sqrt{2}}{2}T_1, \\ y_c &= y_0 + \frac{l\sqrt{2}}{2}(\cos(g) + \sin(g)) = y_0 + \frac{l\sqrt{2}}{2}T_2, \\ z_c &= z_0, \end{aligned} \tag{10}$$

where T_1 and T_2 are obviously given by

$$\begin{aligned} T_1 &= \cos(g) - \sin(g), \\ T_2 &= \cos(g) + \sin(g). \end{aligned} \tag{11}$$

Notice that for the initial position given in Figure 1, the spheres $\mathcal{S}_1$, $\mathcal{S}_2$, $\mathcal{S}_3$, and $\mathcal{S}_4$ respectively have angles of $-\pi/2$, 0, $\pi/2$, and $3\pi/2$. In general, the angle of $\mathcal{S}_2$, $\mathcal{S}_3$, and $\mathcal{S}_4$ can be computed from the angle of $\mathcal{S}_1$, denoted by g_1, by using $g_1 + (i-1)\pi/2$, with $i = 2, 3, 4$, respectively.

Considering Equation (10), we have that

$$\begin{aligned} x_c^2 + y_c^2 &= \left(x_0 + \frac{l\sqrt{2}}{2}T_1\right)^2 + \left(y_0 + \frac{l\sqrt{2}}{2}T_2\right)^2 \\ &= x_0^2 + \frac{l^2}{2}T_1^2 + l\sqrt{2}x_0T_1 + y_0^2 + \frac{l^2}{2}T_2^2 + l\sqrt{2}y_0T_2 \\ &= x_0^2 + y_0^2 + l^2 + l\sqrt{2}(T_1x_0 + T_2y_0), \end{aligned} \tag{12}$$

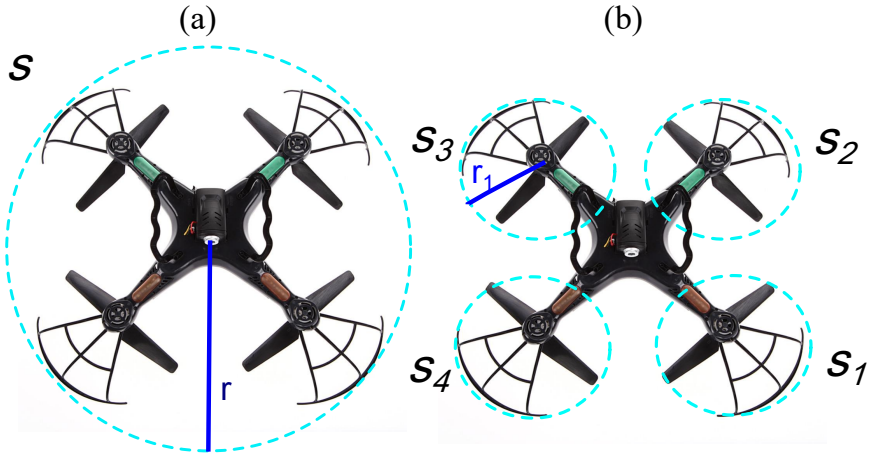


Figure 1: UAV models: (a) one sphere; (b) four spheres.

where we have used $T_1^2 + T_2^2 = 2$. Substituting Equation (12) into Equation (9), we obtain

$$\begin{aligned} a_1(g) &= A_1 + c^2 C(g), \\ a_2(g) &= A_2 - C(g), \end{aligned} \tag{13}$$

where

$$\begin{aligned} A_1 &= -(a^2c^2 - a^2r_1^2 + c^2r_1^2 - c^2x_0^2 - c^2y_0^2 \\ &\quad + a^2z_0^2) + l^2c^2, \\ A_2 &= a^2 - c^2 + r_1^2 - z_0^2 - x_0^2 - y_0^2 - l^2, \\ C(g) &= l\sqrt{2}(T_1x_0 + T_2y_0), \end{aligned} \tag{14}$$

with T_1 and T_2 as defined in Equation (11). From these expressions and considering the definition of a_0 of Equation (9), we can compute $\Delta(g)$ for each sphere by using the following expressions

$$\begin{aligned} Q(g) &= \frac{3a_1(g) - a_2(g)^2}{9}, \\ R(g) &= \frac{9a_2(g)a_1(g) - 27(\frac{r_1}{r})^2 a_0 - 2a_2(g)^3}{54}, \\ \Delta(g) &= Q(g)^3 + R(g)^2. \end{aligned} \tag{15}$$

For this UAV model, the collision detection can be performed applying the Algorithm 1 for each sphere. However, with the goal of reducing the computational complexity, we propose a two-step algorithm. For the first step, we will analyse the position of the total sphere $\mathcal{S}$. When $\Delta < 0$, we use the same procedure as done in Algorithm 1 to determine if the movement is valid or non-valid. Otherwise, when $\Delta \geq 0$, we refine the decision by computing the discriminant $\Delta(g)$ for the four spheres.

The resulting algorithm is summarized as detailed below, and we will referred to as Algorithm 2.

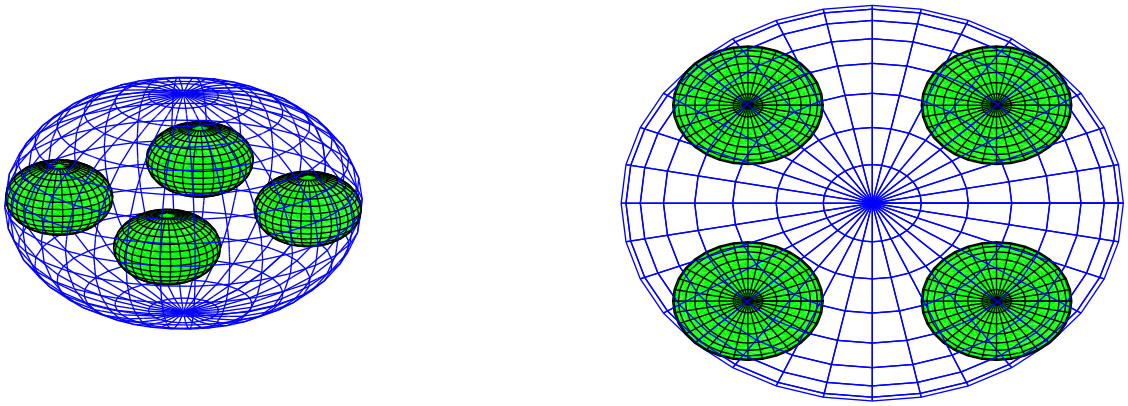
Algorithm 2: Four spheres.

For the sphere $\mathcal{S}$ of radius r, **compute** Δ.
If $\Delta < 0$
  Compute the root λ_2.
  If $\lambda_2 < 0$, **then** *valid movement (interior)*,
  otherwise
    Compute the root λ_3.
    If $\lambda_3 < 0$, **then** *valid movement (interior)*,
    otherwise *non-valid movement*.
otherwise
  Compute $\Delta(g_1 + (i-1)\pi/2)$ for $i = 2,3,4$, with g_1 being the $\mathcal{S}_1$ angle,
  If all $\Delta(g_1 + (i-1)\pi/2) < 0$, **then** *valid movement (interior)*,
  otherwise *non-valid movement*.

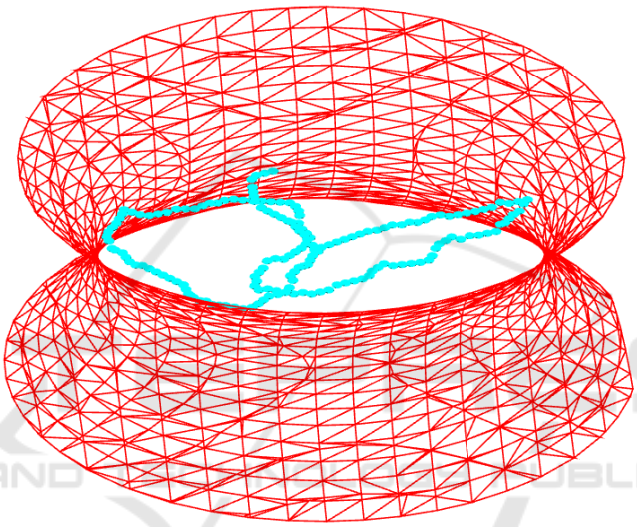
5 EXPERIMENTAL RESULTS

We consider a computer simulated environment where the hyperboloid models the cooling tower of a power station. This tower has a height of 35.6 m and a base diameter of 36.5 m. The UAV has a radius of 25 cm. Different motor radius have been considered in our simulations. Figure 2(a) shows the UAV model. The UAV flies inside the cooling tower following a random trajectory determined as follows:

- Straight UAV movement with probability 0.25.
- Down or up UAV movement with probability 0.25.



(a) UAV model: 3-D and 2-D views.



(b) Example of an UAV trajectory.

Figure 2: Computer simulations.

- UAV random turn with angle in $[-\pi/4, \pi/4]$ with probability 0.5.
- UAV turn of angle $-\pi/2$ or $\pi/2$ if a *non-valid movement* is detected with Algorithm 1.

We consider that the UAV follows a trajectory of $D = 500$ meters with a speed of v m/s. Before any movement, the UAV evaluates the Algorithm 1 or the Algorithm 2 to determine if the movement is valid. This prediction is done considering the position of the UAV in t seconds, i.e. its response time. For a non-valid movement, the UAV turns and tries a new movement. Figure 2(b) plots only one realization of our simulations.

We are interested in comparing the performances of both proposed algorithms for different motor radius. We have considered 5, 7.5, and 10 cm of radius. Table 2 shows the results obtained for this comparison between the Algorithm 1 (model of one sphere) and the Algorithm 2 (model of four spheres) considering different speeds and response times. The following expression has been evaluated

$$\varepsilon(\%) \quad = \quad 100 \times \frac{\text{non-valid movements with Alg. 2}}{\text{non-valid movements with Alg. 1}}. \tag{16}$$

For all response times, we can see that there exists a considerable difference between both algorithms for a radius of $r = 5$ cm. This difference reduces when the radius increases since the volume of the small spheres is similar to the volume of the big one. For instance, the accuracy achieved with the Algorithm 1 is higher than a 90% for all speeds with a radius of $r = 10$ cm.

Table 2: Comparison of the proposed algorithms in terms of $\varepsilon(\%)$ for three different UAV response times.

(a) $t = 10\,\text{ms}$

Speed (m/s)		5	6	7	8	9
Precision (cm)		5	6	7	8	9
Radius	5	51.49	52.22	58.30	66.22	70.86
(cm)	7.5	79.20	82.92	88.09	91.05	92.15
	10	91.01	93.82	94.91	94.07	96.61

(b) $t = 50\,\text{ms}$

Speed (m/s)		5	6	7	8	9
Precision (cm)		25	30	35	40	45
Radius	5	68.18	68.29	69.42	70.49	81.81
(cm)	7.5	86.56	89.83	90.40	93.41	93.22
	10	92.82	93.54	94.40	98.10	99.12

(c) $t = 100\,\text{ms}$

Speed (m/s)		5	6	7	8	9
Precision (cm)		50	60	70	80	90
Radius	5	68.42	71.42	75.90	80.00	83.33
(cm)	7.5	89.00	91.81	99.20	99.50	95.71
	10	94.61	95.33	96.00	98.50	99.70

6 CONCLUSIONS

We have modelled the UAV as a spherical surface centred at the UAV centre or as four spherical surfaces respectively centred at each one of the four motor centres. We have shown that the roots of a characteristic polynomial can be used to decide if there is or not contact between the UAV and a rigid surface. We have considered a hyperboloid surface due its architectural properties and to its presence in real scenarios that could require control or maintenance, although the ideas here presented could be used for other surface types. Both proposed algorithms have a very low computational complexity since they are computing the roots of that characteristic polynomial only a very reduced number of times.

The results obtained from computer simulations show that the algorithm for collision detection based on a model of sphere per UAV motor is more adequate when a high or medium distance precision or equivalently, low response times, is needed. However, the method based on only one sphere could be used for requirements of low precision or, equivalently, high response times.

ACKNOWLEDGEMENTS

This work has been funded by the Xunta de Galicia (ED431C 2016-045, ED341D R2016/012), the Agencia Estatal de Investigación of Spain (TEC2016-75067-C4-1-R) and ERDF funds of the EU (AEI/FEDER, UE).

REFERENCES

Albaker, B. and Rahim, N. (2009). A survey of collision avoidance approaches for unmanned aerial vehicles. In *IEEE International Conference for Technical Postgraduates (TECHPOS)*, pages 1–7.

Brozos-Vázquez, M., Pereira-Sáez, M., Souto-Salorio, M., and Tarrío-Tobar, A. D. (2016). Contact detection between a sphere and a hyperboloid by means of the roots of a characteristic polynomial. *arXiv preprint arXiv:1602.06744*.

C. O'Sullivan, J. D. (1999). Real-time collision detection and response using sphere-trees. In *15th Spring Conference on Computer Graphics*, pages 83–92.

Choi, Y.-K., Wang, W., and Liu, Y. (2006). Continuous collision detection for two moving elliptic disks. *IEEE Transactions on Robotics*, 22:213–224.

Giancarmine, F., Accardo, D., Moccia, A., Carbone, C., Ciniglio, U., Corraro, F., and Salvatore, L. (2008). Multi-sensor-based fully autonomous non-cooperative collision avoidance system for unmanned

air vehicles. *Journal of aerospace computing, information, and communication*, 5(10):338–360.

Jia, X., Choi, Y.-K., Mourrain, B., and Wang, W. (2006). An algebraic approach to continuous collision detection for ellipsoids. *Computer Aided Geometric Design*, 28:164–176.

Krokowicz, T., Gasca, M., Voos, H., and Ucinski, D. (2010). Indoor navigation for quadrotor uavs using schematic environment maps. In *Robotics in Alpe-Adria-Danube Region (RAAD), 2010 IEEE 19th International Workshop on*, pages 457–462. IEEE.

Park, J.-W., Oh, H.-D., and Tahk, M.-J. (2018). UAV collision avoidance based on geometric approach. In *IEEE SICE Annual Conference*, pages 2122 – 2126.

Pham, H., Smolka, S. A., Stoller, S. D., Phan, D., and Yang, J. (2015). A survey on unmanned aerial vehicle collision avoidance systems. *arXiv preprint arXiv:1508.07723*.

Steinbach, K., Kuffner, J., Asfour, T., and Dillmann, R. (2006). Efficient collision and self-collision detection for humanoids based on sphere trees hierarchies. In *6th IEEE-RAS International Conference on Humanoid Robots*.

Strobel, A. and Schwarzbach, M. (2014). Cooperative sense and avoid: Implementation in simulation and real world for small unmanned aerial vehicles. In *IEEE International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1253–1258.