Design of a Strategy for Planning of Trajectories With Avoidance of Obstacles of a Quadrotor Drone Used in the Characterization of Geological Outcrops

R.L. Mogollón Mendoza
Department of Mechatronics Engineering
Universidad Autónoma de Bucaramanga
Bucaramanga, Colombia
rmogollon130@unab.edu.co

F.P. Montes Monterrey
Department of Mechatronics Engineering
Universidad Autónoma de Bucaramanga
Bucaramanga, Colombia
fmontes@unab.edu.co

C.E. Moncada Guayazán
Department of Mechatronics Engineering
Universidad Autónoma de Bucaramanga
Bucaramanga, Colombia
cmoncada@unab.edu.co

S. Roa Prada
Department of Mechatronics Engineering
Universidad Autónoma de Bucaramanga
Bucaramanga, Colombia
sroa@unab.edu.co

Abstract— **This article focuses on the development of an algorithm for a Quadrotor-type Drone that allows it to recognize the environment and move around in it by avoiding obstacles. This article covers the simulation phase that takes place in the MATLAB software, where algorithms for planting trajectories in known static environments were created in two stages: an artificial force field (APF) model is created for each element in the environment using sigmoid functions, the trajectories are subsequently calculated using gradient descent based strategies. Two strategies were developed, the first based on mobile points that interconnect the drone with the goal, where later each point moves towards areas free from the influence of obstacles, following the potential field, which makes paths free of obstacles. The second is based on the use of the concept of safe zones, which is used as a criterion to update the position of the points.**

Keywords— Trajectory, obstacles, quadrotor and environment.

I. INTRODUCTION

In the study of nature there are several branches such as geology which helps us to know the state of geological outcrops and other soil characteristics, which becomes a very tedious and complicated task when the terrain or is difficult to access.

In the aforementioned we can note that the problem is mainly the difficult access to the areas for the study of geological outcrops, for which the solution is proposed to develop a strategy for unmanned vehicles that allows us to access these sites also having as Obstacle avoidance priority since being an unknown terrain we do not know for sure what may be there, being the most optimal solution when it comes to thoroughly characterizing any geological outcrop thanks to its maneuverability and flight autonomy.

There are several methods for the development of a mechatronic system, however, the best option for software design is the V methodology.

The V methodology is a process that represents the sequence of steps in the development of the life cycle of a project. The activities and results that must be produced during product development are described. The "V" in the model name refers to how the model compares the development phases with the corresponding quality control phases. The left arm of the letter "V" contains the design and development tasks of the system, and the right arm the quality control measures of each phase. At the junction between the two arms, the implementation of the product is located [1].

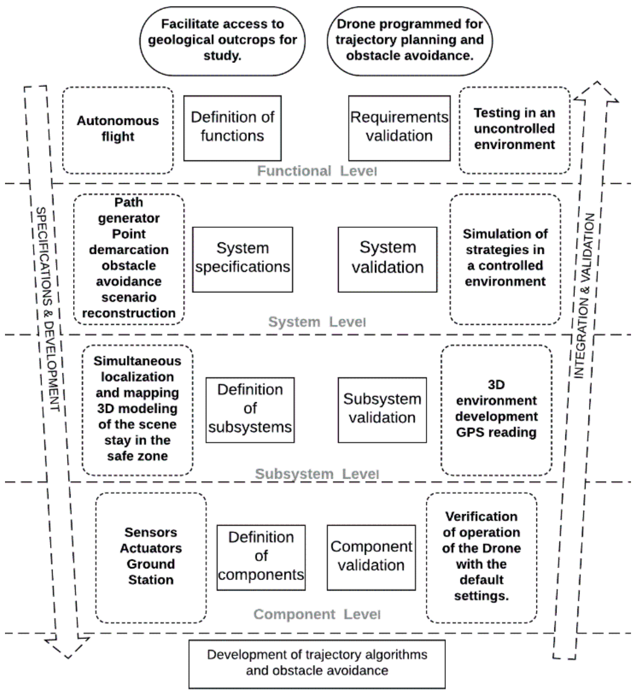

Fig. 1. V-Methodology diagram [2].

This article will cover the development phase according to the methodology proposed in the following way, in section I the functional level, in section II the system level, in section III the subsystem level, in section IV the component level.

I. FUNCTIONAL LEVEL

Based on antecedents on autonomy in UAVs, the following needs are identified:

- It is an algorithm that allows the generation of obstacle-free trajectories between two points in space.
- A trajectory that does not have convergence problems caused by local minimums.
- An algorithm that allows you to have direct control over the distance at which you approach obstacles.
- A smooth trajectory with minimal oscillations.
- The data acquisition card with good processing speed to avoid system failure.

II. SYSTEM LEVEL

A. Environment representation

The drone can move over an environment in which it may encounter various obstacles, once the equations of the planes that constitute the environment of the environment over which the UAV is traveling have been found, they are modeled so that a mathematical representation that makes it possible to determine which are the zones through which it is possible to circulate freely, which zones would present a high risk of collision and at which points the collision is imminent. The strategy to generate a model of the environment selected in this case is based on potential fields.

Using sigmoid functions, a potential field can be generated throughout the space depending on the location of the elements that make up the environment. The use of sigmoid functions is considered because it allows to create a smooth force field model in the entire design environment and it is only necessary to adjust some parameters to change the effective action range of the field generated individually by each obstacle [3].

For the representation of the environment and obstacles in the two-dimensional case, Equation 1 must be taken into account, in which Xobs represents the value in X or Y of the obstacle and *gamma* is a coefficient that allows modifying the smoothness of the change of the function, the sign of *gamma* determines whether the monotonic function is increasing or decreasing.

$$f_{sig}(x) = \frac{beta}{1+e^{-gamma(Xini-Xobs)}} * \frac{1}{1+e^{-gamma(Yini-Xobs)}} * \frac{1}{1+e^{gamma(Xini-Xobs)}} * \frac{1}{1+e^{gamma(Yini-Yobs)}} \quad (1)$$

In the same way it is presented for the three-dimensional case, adding the coordinates in Z.

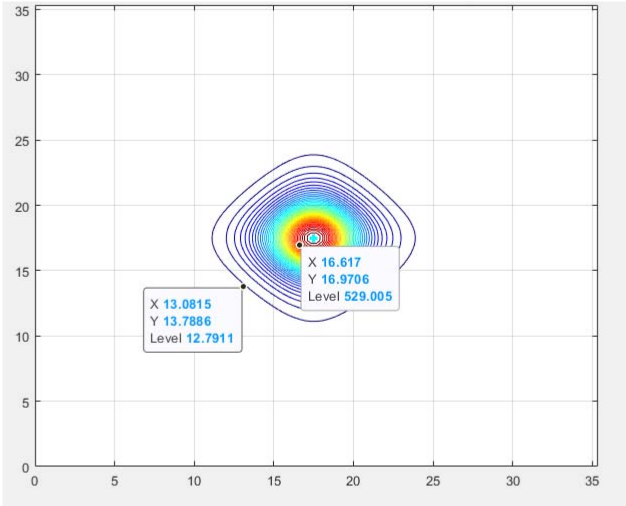

Fig. 2. Representation of an obstacle using sigmoid functions [2].

III. SUBSYSTEM LEVEL

Two techniques are developed, based on an extrapolation of the descent of the traditional 2D gradient to the three-dimensional case, the first is based on mobile points that interconnect the Quadrotor with the goal, where later each point moves towards areas free of the influence of obstacles following the potential field, which that makes pathways clear of obstacles. The second is based on the use of the concept of safe zones, which is used as a criterion to update the position of the points[2].

These two techniques are based on the descending gradient method that seeks to find the minimum of a function, be it local or global. The method is based on the use of the negative gradient, since in that direction we will obtain the maximum decrease in the values of the function.

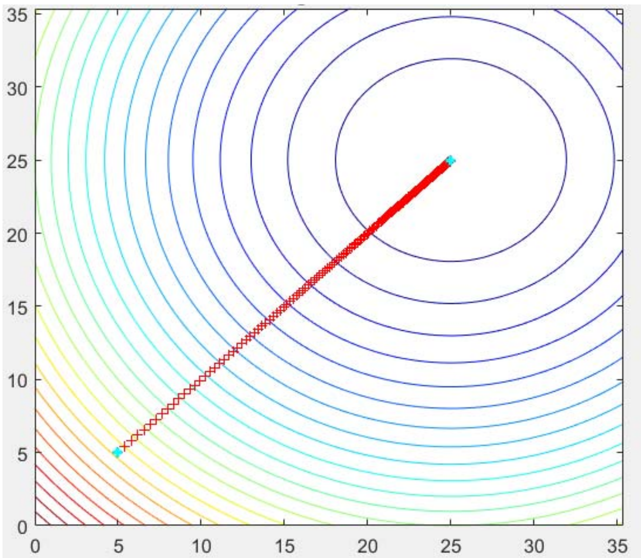

Fig. 3. Trajectory of gradient descent [2].

1. Algorithm based on moving independent points

As defined in [3] this algorithm allows the generation of obstacle-free trajectories between two points in space. This particular algorithm is very simple to program and consists of few steps, although its simplicity contrasts with the advantages it provides when implemented. It is based on the descent of the gradient, the principle of artificial potential fields and uses the concept of safe zones for its operation.

Initially, the parameters and locations of the starting point, the target point and the obstacles are defined. We also define a circumference of points that will later be evaluated to determine if they are free from the influence of any obstacles.

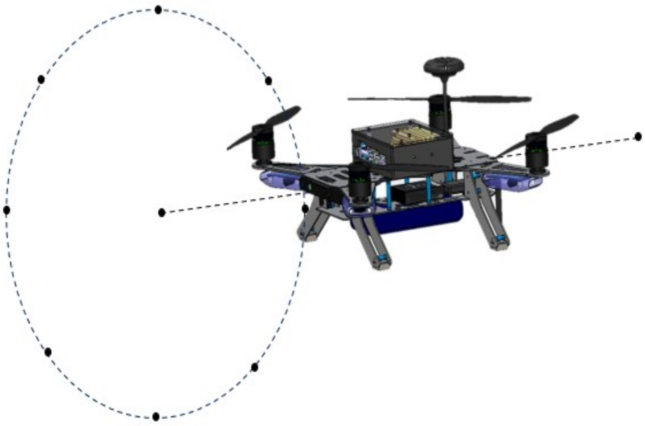

Fig. 4. Location of the praton of points. [2].

Then it makes the representation in two dimensions or three dimensions, as the case may be, of the obstacles using the sigmoid function, in this way the potential fields of each obstacle can be observed.

The starting points of the trajectory are located on the line segment that joins the starting position and the destination point. Next, it is necessary to find the APF gradient generated by the obstacles associated with each point, this will determine a vector with the direction towards which each point can move in space. Finally, except for the first and last point, each point moves in the direction obtained with the gradient until it finds a position in which the APF value is within the values of the safe zone.

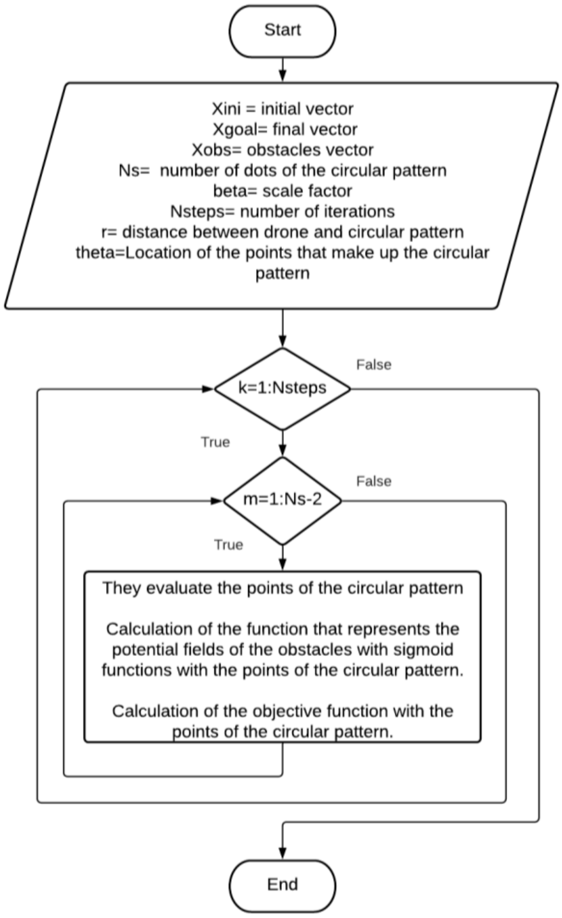

Fig. 5. Moving Independent Points Algorithm Flow Chart.[2].

2. *Results of the simulations of moving independent points*

In this section, two cases of convergence of the mobile independent points algorithm are presented, the first case in two dimensions. In Figure 6 a collision-free trajectory is shown moving along the side of obstacles.

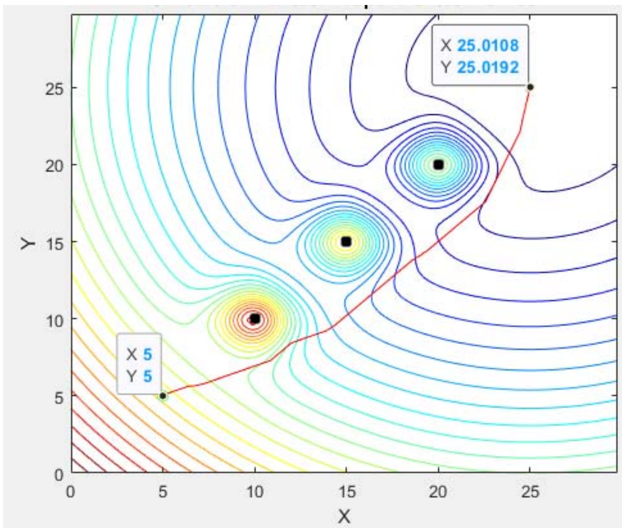

Fig. 6. Simulation of the trajectory of moving independent points in 2 dimensions. [2].

In the second case of convergence, it is presented in figure 7, where we can see that the trajectory crosses the field with the lowest value and avoids the others

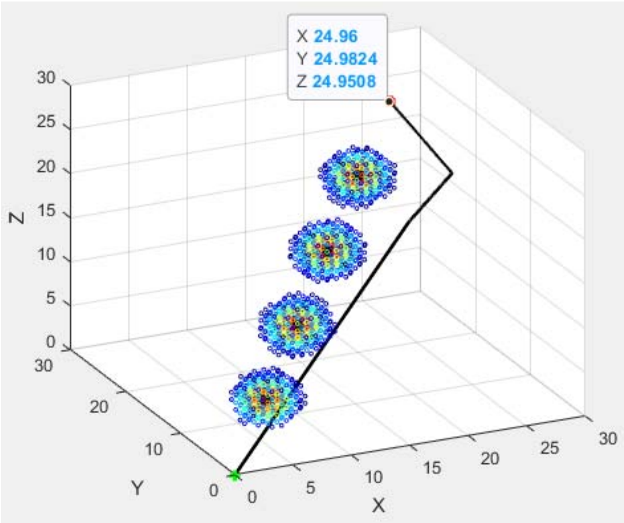

Fig.7. Simulation of the trajectory of moving independent points in 3 dimensions. [2].

3. *Algorithm based on staying in the safe zone*

An iterative algorithm is presented that allows the generation of obstacle-free trajectories in three dimensions, is based on the principle of artificial potential fields and uses the concept of safe zones for its operation.

When a point k is within the region in space called the safe zone, we proceed to find the position for the point of the trajectory k + 1, for this we calculate the gradient vector that will give the direction of movement, which is given by the direction of the line that interconnects point k with the target point.

On the contrary, when the point k is outside the region in the space called the safe zone, the APF values are used to determine a vector perpendicular to the field. This vector is calculated by finding the partial derivative in each direction on a selected point and then normalizing it. Now a search process begins in which, from the current position of point k, a movement begins following the direction of the gradient vector but with a movement perpendicular to this straight line until the point in which the point is within the safe zone , once there, only the gradient movement will be taken into account.

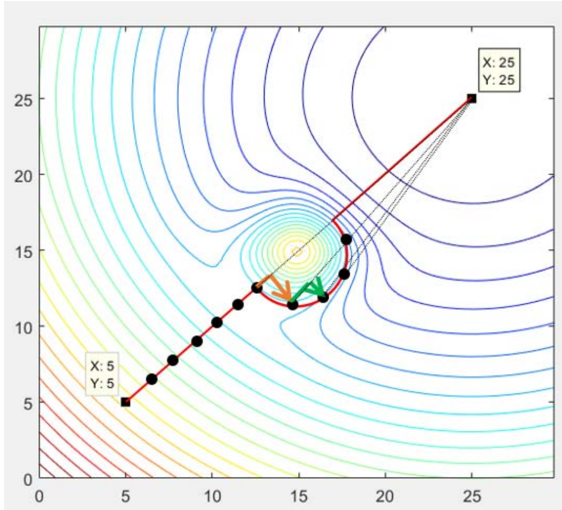

Fig. 8. Movement of each point until finding positions within the safe zone. [2].

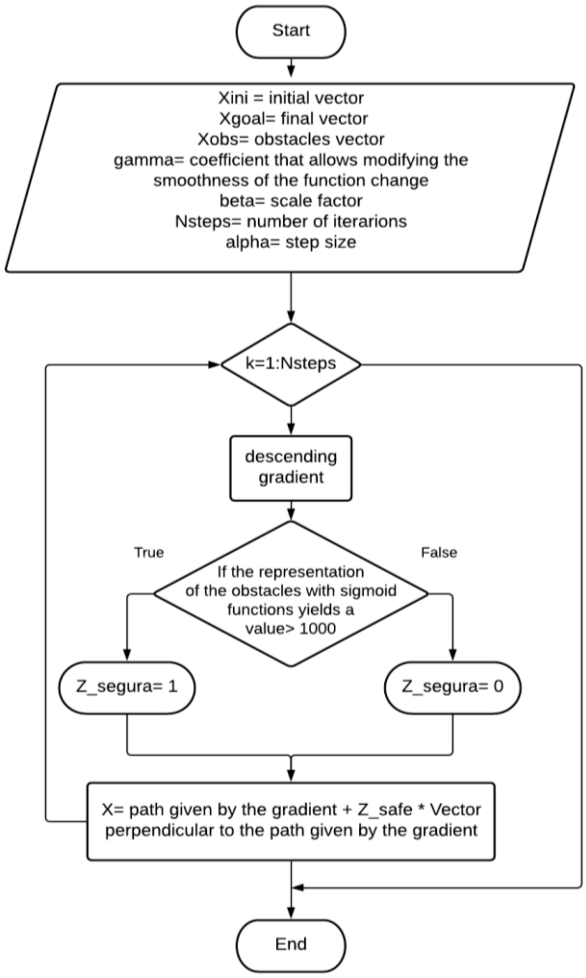

Fig. 9. Safe Zone Algorithm Flow Chart [2].

4. *Results of safe zone simulations*

This section presents the simulations of two possible scenarios for the algorithm based on the permanence in the safe zone on potential sigmoid fields. Figure 10 shows a two-dimensional trajectory that converges satisfactorily.

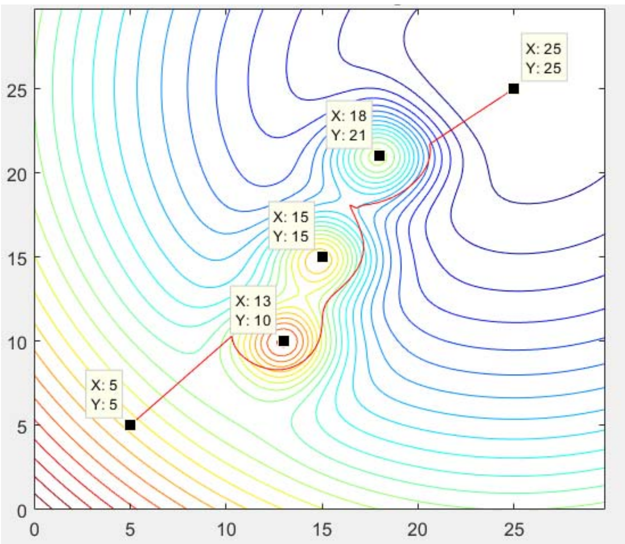

Fig. 10. Simulation of the trajectory of safe zones in 2 dimensions [2].

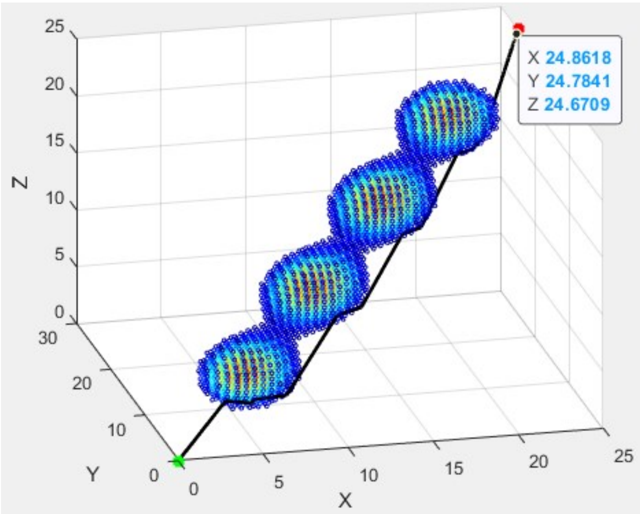

Fig. 11. Simulation of the trajectory of safe zones in 3 dimensions [2].

IV. Component level

The Quadrotor used in this work is aero ready to fly drone by Intel, which is composed of a carbon fiber support structure on which the motors are mounted, a graphics processing unit (GPU), the flight controller, power modules, camera and GPS antenna.

In figure 12 you can see the distribution of the quadrotor and its components.

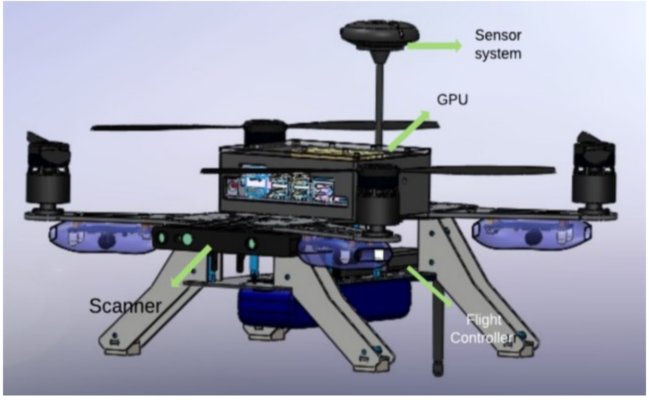

Fig. 12. CAD of drone made with SolidWorks[2].

The drone is in communication with the GCS (ground control station) through the radio transmission-reception component. The mission and the initial flight configuration are sent from the GCS. In addition, requests for flight information and images are sent which are attended through the same communication channel. Internally, the drone communicates its two parts, the flight controller and the CPU. This communication allows the GPU to send information and images to the User passing through the flight controller and also change the mission if applicable. The diagram of the modules and their communication can be seen in figure 13.

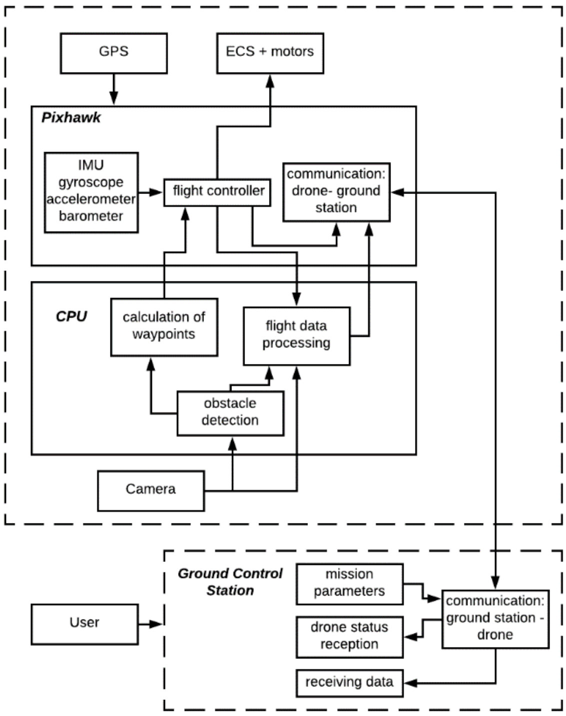

Fig. 13. Diagram of the Hardware modules that make up each unit, the functions they implement [2].

Pixhawk sensors are not calibrated at the factory and must be calibrated for operation. QGroundContol provides a way to do it following a procedure.

QGroundContol requires that the plate plus the external magnetometer be placed in certain positions and certain movements made to calibrate the sensors.

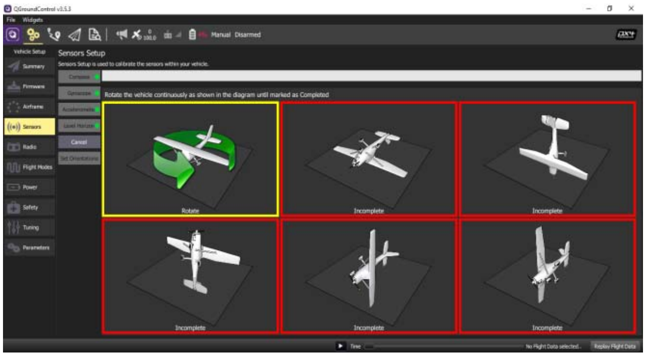

Fig. 14 GUI de QGC calibration *[2].*

With the calibration of the sensors we would have the drone ready to perform tests with its default configuration.

Previously, the simulation of a scenario is presented to explore the characteristics of the proposed techniques, a case is presented in which there are three obstacles that, due to their location, mean that there is no point of view between the starting point and the goal point.

It is also evident how they generate trajectory points that maintain a minimum distance to obstacles regardless of their dimensions, allowing the trajectory to be free of collisions and additionally avoiding obstacles having control over the proximity to them.

The independent mobile point technique highlights some of the drawbacks such as lack of control over the minimum distance that approaches objects, the distance traveled with this technique is much greater, which entails using a large number of points to be evaluated.

V. CONCLUSIONS

In the proposed configuration in 3D we can see that the safe zone technique converges satisfactorily compared to independent moving points in which it can be observed that I exactly reaches its goal point.

The smoothness of the final trajectory depends mainly on the number of points used, the more points the greater the smoothness in the result, although this leads to an increase in computational cost.

The total distance traveled using safe zones is less with respect to independent moving points, this is due to the heuristics handled by safe zones, where the points are updated in each iteration seeking to move towards the target point, while in independent moving points the points are updated according to their starting position.

The method of safe zones is better than that of independent mobile points since at the processing level due to its execution time, which was measured in MATLAB by the "tic toc" function, the safe zones algorithm gave a shorter time with a time of 0.82 seconds versus a time of 2.17 seconds that the independent moving points algorithm took, the method of independent mobile points requires greater processing capacity, in addition to being limited to the number of points desired by the user, which can be counterproductive. that the more points, the better the result may be but with a long data processing time.

In the proposed configuration in 2D, the two techniques converge generating trajectories. In the safe zone technique, it is possible to observe small inconveniences of the descent down the gradient such as oscillations near obstacles.

On the other hand, the technique of moving independent points presents disadvantages of descent through the gradient such as oscillations near obstacles and near the goal point.

As mentioned above, the safe zones technique presents a shorter route with good control of the distance between the obstacle and the drone.

REFERENCES

[1] IONOS Digital Guide, ¿Qué es el modelo V?, 2020. Recuperado de https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/modelo-v/

[2]Montes, F., Mogollon, R. (2020). Estudiantes Universidad Autónoma de Bucaramanga. Programa de Ingeniería Mecatrónica. Bucaramanga: UNAB.

[3] D. Rivera Pinzón, «Desarrollo de un algoritmo de evasión de obstáculos para Quadrotors en ambientes dinámicos utilizando una cámara de profundidad,» Universidad Nacional de Colombia, Bogotá, 2012.

[3] F. Restrepo y P. Rizo, «Planeación de trayectorias para un robot aéreo AR. Drone 2.0 usando GPS.,» Pontificia universidad Javeriana. Bogotá, Colombia, 2014.

[4] P. Newman, S. Clark y H. Durrant-Whyte, «A solution to the simultaneous localization and map building (slam) problem,» Robotics and Automation, IEEE Transactions on, 2001.

[5] D. H. Gómez, «Desarrollo de una técnica SLAM para ambientes dinámicos tridimensionales,» Universidad Nacional de Colombia, Bogotá, 2015.