



# OPTIMAL MULTI-SENSOR OBSTACLE DETECTION SYSTEM FOR SMALL FIXED-WING UAV

Marta Portugal and André C. Marta

IDMEC

Instituto Superior Técnico

Universidade de Lisboa

Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal

{marta.portugal, andre.marta}@tecnico.ulisboa.pt, <https://mdo.tecnico.ulisboa.pt/>

**Abstract.** *This work provides a solution for the safety enhancement of small fixed-wing UAVs regarding obstacle detection during flight. The main goal is to implement an optimal multi-sensor system configuration. To achieve it, preceding works regarding the integration of available sensors in such systems were studied. As a result, select sensors (ultrasonic sensor, laser rangefinder, LIDAR and RADAR) were modeled for collision detection and avoidance simulations using the potential fields method. An optimization study using a genetic algorithm was conducted to find the sets of sensors and respective orientation that result in the best collision avoidance performance. To do so, a set of randomly generated collision scenarios with both stationary and moving obstacles were generated. This study resulted in relatively simple detection configurations that still provided high collision avoidance success rate. The ultrasonic sensor revealed to be inappropriate given its short range, while the laser rangefinder benefited from long range but had very limited field-of-view. In contrast, both the LIDAR and the RADAR are the most promising, as they exhibit not only a significant range but also a broad field-of-view. The best multi-sensor configurations were either a front-facing LIDAR or RADAR, complimented by a pair of laser rangefinders pointing sideways at an angle of 10 or 63 degrees, respectively. Once the hardware that should integrate an optimal system was known and available, the assembly of the final system, including the sensors and a PixHawk flight controller, was designed. The appropriate software (PX4 and QGroundControl) was also built and adapted to the current work. To validate the proposed system, all sensors were first individually tested before assembling the complete system. The bench tests attested the accuracy of the sensor specifications and previous simulations. As such, ground tests using a simple rover shall follow. Once the system is validated under these conditions, flight tests may begin.*

**Keywords:** Sense and avoidance, collision avoidance, sensor fusion, optimization, laser rangefinder, LIDAR

## 1 INTRODUCTION

Unmanned Aerial Vehicles (UAVs) have received considerable attention in a myriad of operations due to their enhanced stability and endurance. Despite being initially developed for military purposes [1], there has been a notable upsurge in the civilian market for UAVs [2].

Some applications of UAVs present high collision risk. Due to their ability to work in a collaborative and cooperative manner, swarms of drones are typically used for surveillance purposes, tracking and localizing objects. One of the most significant challenges regarding the navigation of a swarm of agents is collision avoidance. Collision avoidance systems are responsible for guiding an autonomous agent to safely and reliably avoid potential collisions with other agents in the swarm as well as with other objects in the environment. The capacity to locally sense and avoid items in the environment becomes more crucial for agents to be fully autonomous and, in turn, for systems to be more robust [3]. Drones are also required to exhibit a practical resolution for a Sense and Avoid feature as part of the NextGen [4] strategy for integrating UAVs into the U.S. National Airspace System (NAS). In fact, all UAVs must deploy an automated Sense and Avoid intelligent system that provides safety levels comparable to or even superior to those of manned aircraft [5].

Fittingly, this work specifically addresses the safety enhancement of small fixed-wing UAVs (maximum take-off weight  $< 25$  kg, range  $< 10$  km, endurance  $< 2$ h and flight altitude  $< 120$  m), particularly with regard to the detection of obstacles during flight and the automatically triggered collision avoidance maneuver. It is part of a comprehensive obstacle detection and collision avoidance system, representing a two-stage "sense" and "avoid" problem, being this work more focused on the former. Preceding this work, different detection systems were simulated using laser rangefinders and RADARs in different configurations [6]. Through the Potential Fields method and resorting to an optimization algorithm, a possible configuration of the UAV detection system was reached. Subsequently, ultrasonic sensors and laser rangefinders have been employed in the hardware implementation of an effective Sense and Avoid System on a simple rover [7]. In this work, the main goal is to implement an adapted version of the forementioned systems on a small fixed-wing UAV, integrating an optimal multi-sensor configuration that can include ultrasonic sensors, laser rangefinders, RADARs and LIDARs.

## 2 SENSOR MODELLING

The next subsections describe different models of active non-cooperative sensors: the ultrasonic sensor, the laser rangefinder, the LIDAR (Light Detection and Ranging) and the RADAR (Radio Detection and Ranging), as illustrated in Fig. 1, followed by their comparative analysis in Tab. 1. Different models were developed by [8] and further adapted to the present work.

### 2.1 Ultrasonic Sensor

This sensor generates a sound, which is then reflected by the obstacle and recorded by the sensor. If the velocity of the radiated sound in the air medium is known, the distance from the point of greatest reflection to the obstacle can be calculated [13]. Using ultrasonic sensors proves to be advantageous mainly due to the ease at which this simple technology can be sized down. However, because these are proximal sensors, their signal



(a) Ultrasonic sensor. (Source: MaxBotix.)



(b) Laser rangefinder. (Source: LightWare.)



(c) LIDAR. (Source: LightWare.)



(d) RADAR. (Source: Ainstein.)

Figure 1: Active non-cooperative sensors.

Table 1: Sensor hardware specifications.

	Ultrasonic sensor MB1242 [9]	Laser rangefinder LW20/C [10]	LIDAR SF45/B [11]	RADAR US-D1 [12]
Range (m)	7	100	45	50
Horizontal FOV (°)	0	0.3	20-320	43
Resolution (cm)	1	1	1	–
Accuracy (m)	0.1	0.1	0.1	0.04
Update rate (Hz)	7	388	5000	100
Power supply voltage (V)	3-5.5	4.5-5.5	4.5-5.5	5-5.5
Power supply current (mA)	4.4	100	300	400
Outputs and interfaces	Serial and I2C	Serial and I2C	Serial and I2C, Micro USB	UART, CAN
Dimensions (mm)	22x19x15	30x20x43	51x48x44	108x79x20
Weight (g)	5.9	20	59	110

quickly attenuates and their capacity to measure distance is typically limited to less than 10 meters [14]. This type of sensor has a wide FOV that translates to a beam pattern with axial symmetry, as represented in Fig.2.

Since the ultrasonic sensor only outputs a distance, it leaves all the interior beam points located at a specific distance from the UAV as potential object positions. This results in errors that shall be avoided, as well as other issues that arise from sound reflection. The sound reflection law states that the reflected sound wave's angle with the normal of the surface is preserved. Thus, the ultrasonic sensor requires a perpendicular surface in order to detect an object, which in turn implies that the targets format is crucial to the mission's success. It is vital to recognize the final results can only be used as

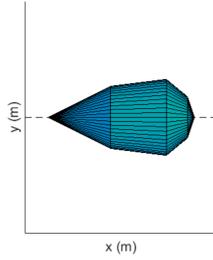


Figure 2: Ultrasonic sensor beam pattern [7].

a reference given that these simulations only use spherical-shaped targets (and different target formats could either improve or worsen the outcome). In short, the model must check for these possibilities at all times:

1. The presence of any spherical surface point within the sonar beam pattern;
2. The perpendicularity of the sound wave direction with its reflecting surface.

Verifying these conditions requires considerable computing time. Therefore, a progressively complex approach that avoids unnecessary blocks of code was implemented [7]. First, the beam pattern is reduced to a cylinder. When the center of the obstacle is found to be inside the cylinder, a more thorough analysis is performed to identify which portion of the spherical surface, if any, is in fact inside the beam pattern. The last stage addresses the perpendicularity issue. The final surface computed in the preceding phase is defined as a list of points with a sampling ratio 25 times bigger for each spherical coordinate (in relation to the first list of points).

## 2.2 Laser Rangefinder

Laser rangefinders are able to compute distances to obstacles by emitting a laser pulse and measuring the time it takes for the reflected beam to be detected, given that laser light beams move at a known speed. This principle is quite common among sensors, accounting for lightweight, low-cost technology [15]. However, it is limited by weather conditions, as laser light might scatter in the presence of clouds, fog or atmospheric attenuation.

Given that all sensors' models may be implemented at an angle  $\beta$  relative to the longitudinal axis, our model assumes the use of two symmetrical sensors at the angles  $\beta$  and  $-\beta$  whenever  $\beta \neq 0$ .

Considering the obstacles as spheres, this can be modeled as a simple interception between a line and a spherical surface given by

$$\|\mathbf{x} - \mathbf{c}\|^2 = r^2 \quad (1a)$$

$$\mathbf{x} = \mathbf{o} + d\hat{\mathbf{u}}, \quad (1b)$$

where  $\mathbf{x}$  is a generic point on the line and/or sphere,  $\mathbf{c}$  is the centre point of the sphere,  $r$  is its radius,  $\hat{\mathbf{u}}$  is the unit vector that defines the line direction in 3D space and  $d$  is the distance from the origin of the line. Combining both equations leads to an easily solvable quadratic equation,

$$d^2(\hat{\mathbf{u}} \cdot \hat{\mathbf{u}}) + 2d[\hat{\mathbf{u}} \cdot (\mathbf{o} - \mathbf{c})] + (\mathbf{o} - \mathbf{c}) \cdot (\mathbf{o} - \mathbf{c}) - r^2 = 0, \quad (2)$$

that returns a solution if  $0 < d_{\text{sol}} < \mathbf{R}_d$ . In real conditions, the laser would not reach the furthest point, reflecting on the closest one. Therefore, if there are two solutions in this interval, only the smallest one prevails. The reflection point with the spherical surface can be easily obtained from Eq. (1b).

### 2.3 LIDAR

Light Detection and Ranging (LIDAR) emits short and precise laser light impulses with high frequency, that in turn, are reflected and received again by the sensor, measuring the time it took for them to return. Although this technology is similar to the laser rangefinder's, it is multidirectional. Thus, its execution goes beyond simply detecting an obstacle's range and 3-D point cloud can be acquired through a vast array of distance measurements.

The LIDAR model is very similar to the laser rangefinder's. As such, only the points that are closest to the sensor are detected. This implies that if an object is totally visible, it is considered that its half was detected and the remaining of the obstacle is reconstructed assuming symmetry, where the center of symmetry is the medium point of the segment connecting the first and last point of the cluster. In the present simulations, this distance corresponds to the diameter of the obstacle. This model discards obstacles that are hidden or outside FOV.

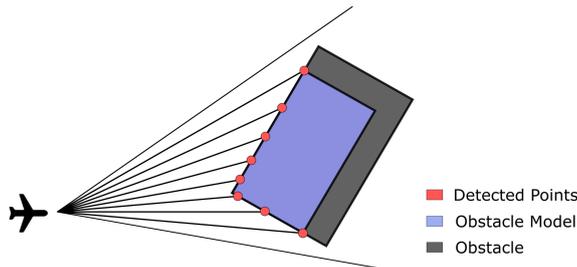


Figure 3: Obstacle reconstruction using a LIDAR [8].

A common issue lies within the higher distance between consecutive points in farther obstacles which results in smaller detected dimensions (see Fig. 3). To solve this problem, the measured diameter is passed through the time filter [16],

$$D_k = D_{k-1} + G(D_m - D_{k-1}), \quad (3)$$

where  $G(0 < G < 1)$  is the filter gain,  $D_k$  is the filtered diameter at instant  $t_k$ ,  $D_{k-1}$  is the filtered diameter at instant  $t_{k-1}$ , and  $D_m$  is the measured dimension at instant  $t_k$ . The gain must be carefully chosen because it affects how quickly the dimensions change. While a small gain (*i.e.*, slow variation) is better for noisy surroundings, it is not appropriate for objects with high relative speeds. The gain is given by

$$G = 1 - \sqrt[n]{1 - p}, \quad (4)$$

where  $p$  corresponds to a fraction that represents the desired accuracy of the dimensions and  $n$  corresponds to the number of filter cycles required to get an accuracy of  $p$ . Classic Kalman filters [17] were employed for the tracking phase, where the motion of obstacles

was assumed to be two-dimensional, linear, and constant over successive scans. This simplification, which takes into account a high scanning frequency, accurately captures the targets' state.

## 2.4 RADAR

Radio Detection and Ranging (RADAR) is one of the most popular sensing technologies. It consists of a transmitting antenna producing electromagnetic waves (in the radio or microwave spectrum) and a receiving antenna, which collects waves echoed from static or dynamic obstacles [18]. By measuring the time lapse between the transmitted and received signal, it is possible to determine the distance between the sensor and the target, since radio waves move at a known speed, in a way that can be projected mathematically. Despite being very similar to the LIDAR, RADAR technology is distinguished by the frequency of the emitted radiation.

In this case, the state estimation is more complex than the one employed in the LIDAR model, given the RADAR sensor provides the range, bearing, and elevation of the observed obstacles. These outputs are polar, while the intruder dynamics are best described in rectangular coordinates. Due to its straightforward implementation, the converted measurement Kalman filter (CMKF) was chosen in [8]. The 2-D model used in the simulations shown is represented by

$$\begin{cases} x_m^u &= \lambda_\alpha^{-1} r_m \cos(\alpha_m) \\ y_m^u &= \lambda_\alpha^{-1} r_m \sin(\alpha_m) \end{cases} \quad (5)$$

where  $(x_m^u, y_m^u)$  are the measurements converted to the Cartesian frame,  $r_m$  is the measured range,  $\alpha_m$  is the measured azimuth and  $\lambda_\alpha$  is the bias compensation factor expressed as

$$\lambda_\alpha = e^{-\sigma_\alpha^2/2}, \quad (6)$$

where  $\sigma_\alpha$  is the standard deviation of the noise in the azimuth measurements. The compensation of the bias is multiplicative due to the use of the unbiased conversion and modeling the measurement errors as Gaussian white noise. The covariance matrix used in the Kalman Filter is given by

$$\mathbf{R}_u = \begin{bmatrix} \text{var}(x_m^u | r_m, \alpha_m) & \text{cov}(x_m^u, y_m^u | r_m, \alpha_m) \\ \text{cov}(x_m^u, y_m^u | r_m, \alpha_m) & \text{var}(y_m^u | r_m, \alpha_m) \end{bmatrix}, \quad (7)$$

with the details of the computation of these variances found in [19].

## 2.5 Multi-Sensor Data Fusion

All of these sensors (and respective models) provide input that allow the avoidance system to actuate. However, if the system's architecture is composed by more than one sensor, the data provided must be merged in some way. Following best practices [20], the weighted filter method is used in the present study. The principle behind this method is simple: each sensor is given a weight that is based on how reliable it is. Reference data sensors that provide information about the UAV state must be installed. Considering that changes in the distance to obstacles correspond to changes in the UAV location, reference data sensors like IMUs and optical flow sensors are used to assess the accuracy of the main

data and aid in selecting the best sensor. In the particular case of fixed obstacles, the aforementioned variances in distance ought to match. The weights are then calculated by applying a differential norm to compare all conceivable sensor combinations of main data and reference data. In each instant, the obstacle distance measurement corresponding to the sensor with the lowest weight is chosen, and the remaining measurements are discarded on the grounds that they are corrupted. Nonetheless, the sensor readings are fused in accordance with their weights if the computed weights have a low variation.

### 3 OPTIMAL SENSING SYSTEM

An optimization study was conducted to find the types of sensors and respective orientation that result in the best collision avoidance performance. To do so, a set of randomly generated collision scenarios with both stationary and moving obstacles were generated. The sensors modelled in Sec.2 were tested for each of these scenarios, varying their orientation until optimal configurations were reached. The scenario generation algorithm and multi-sensor optimization was further developed based on [8].

#### 3.1 Scenarios Generation

In order to create scenarios that are suitable for this study, a scenario generation algorithm was created. Each scenario must specify the obstacle's initial position, velocity and radius. It also includes a pre-planned path and waypoints that the UAV must follow.

Figure 4 is based on the graphical representation of this algorithm [7], depicting the processes that lead to generating a scenario. Different bounds are defined regarding the kinematic and dimensional properties of the obstacles and the UAV itself. Various stochastic and partially stochastic processes were then extracted from these intervals, creating random values for the different variables.

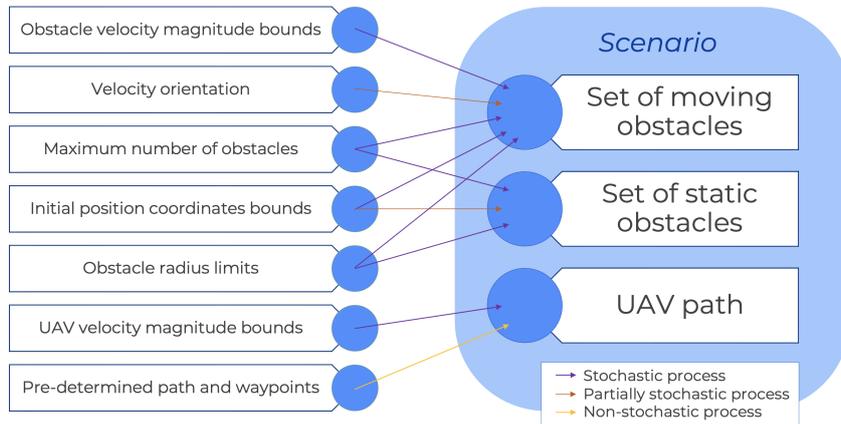


Figure 4: Scenario generation algorithm [7].

Partially stochastic processes have been used in two different cases: determining the velocity orientation of moving obstacles and setting the position of static obstacles. In the former, the goal is to ensure that deviation from the obstacles to the center of the graphical window is not predicted by initial conditions, *i.e.*, initially, the direction of the obstacle's velocity shall point to the centre of the window, rather than pointing outwards, increasing the possibility of collision. In the latter, the initial position of the static obstacle

must not be within the safety radius around the waypoint, given that the UAV must pass through it.

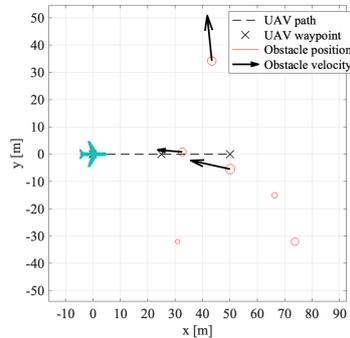


Figure 5: Randomly generated scenario.

An example of a resulting scenario is plotted in Fig. 5. This scenario generating function simply accepts the predetermined path and waypoints of the UAV as an input before combining them with a list of moving and static obstacles to produce a scenario. If the UAV does not go beyond any obstacle’s safety radius throughout the whole simulation (without any sensors), the scenario will be discarded. Until there are  $n$  scenarios with an impending collision, this process is repeated.

### 3.2 Optimization Technique and Problem Formulation

To determine the optimal sensor configuration, different sensor sets were tested. The parameters that characterize each sensor model were obtained from their technical manuals or inferred from available data summarized in Tab. 1. Since our simulations were restricted to the horizontal plane of motion, the vertical FOV is not relevant.

Forty collision-leading scenarios were randomly generated, with obstacle parameters varying according to the limits set in Tab. 2.

Table 2: Data for randomly generated imminent collision scenarios.

UAV speed	# fixed obst.	# moving obst.	obst.radius	obst.speed	obst.direction
[5, 15]m/s	{0, 1, 2}	{0, 1, 2}	[0.5, 2]m	[5, 15]m/s	[0, 90] <sup>o</sup>

In order to optimize the sensor orientation  $\beta$ , a S&A metric function  $f(\beta)$ , to be minimized, was defined as

$$f(\beta) = \sum_j \sum_i (-d_{\min}(i) + \phi_1 |\max(R_s(i) - d_{\min}(i), 0)|^2 + \phi_2 |\max(R_c(i) - d_{\min}(i), 0)|^2), \quad (8)$$

where the first term drives the evasion maneuver to maximize the minimum distance  $d_{\min}$  between the UAV and the obstacle  $i$ , the second term represents the penalty when the minimum distance violates the safety radius  $R_s$  ( $d_{\min} \leq R_s$ ), and the last term represents the penalty when the minimum distance violates the obstacle collision radius  $R_c$  ( $d_{\min} \leq R_c$ ). The metric accumulates not only for every obstacle  $i$  in each scenario but also for all scenarios  $j$ . In order to penalize collision cases more than close-calls, the weights were set to  $\phi_1 = 10$  and  $\phi_2 = 50$ .



(a) Pair of laser rangefinders.

(b) Pair of laser rangefinders and a RADAR.

Figure 6: S&amp;A metric as function of laser rangefinder orientation.

Figure 6 shows the metric defined in Eq. (8) for two particular sensor solution cases: i) using a pair of laser rangefinders with a 100m range, symmetrically pointing forward with an angle  $\beta$  with respect to the UAV longitudinal axis; and ii) adding a RADAR with a 120m range pointing in the direction of the UAV longitudinal axis.

In both cases, the metric proves to be noisy. Thus, the optimization technique selected to find the minimum of  $f(\beta)$  was the Genetic Algorithm (GA) implemented in MATLAB. This gradient-free, population-based method, deals with a set of solutions that are updated simultaneously in each iteration. In practice, compared to other minimization algorithms, this reduces the likelihood of the result being a relative minimum. This problem can be posed as

$$\begin{aligned}
 & \text{Minimize } f(\beta) \\
 & \text{w.r.t. } \beta \\
 & \text{subject to } \beta_{min} \leq \beta \leq \beta_{max}
 \end{aligned} \tag{9}$$

where  $\beta_{min}$  and  $\beta_{max}$  are the lower and upper bounds of  $\beta$ , respectively, to be defined for each particular case. Notice that  $\beta$  is a vector if multiple sensors are used.

The initial population was set to be created with a uniform distribution; the crossover function was set to create 80% of the population in each generation; because the variables are bounded, the mutation function randomly generates directions that are adaptive with respect to the last successful or unsuccessful generation, where the chosen direction and step length satisfy the set bounds. The convergence criteria were set such that the global minimum was found in a timely but accurate manner: a function convergence of  $10^{-3}$  was used with 10 stall generations, and a maximum of 50 generations prescribed. The population size was set to 30 individuals. These parameters were chosen following best practices. The simulations were run on an 1,4 GHz Intel quad-core i5 with 8 GB 2133 MHz RAM.

### 3.3 Optimal Sensing Configurations

The following subsections are dedicated to detailing the proposed sensing architectures, further explaining each solution and the respective optimal result. In the end, the performance of the different sensor sets will be summarized and compared, in order to implement the best solutions.

### 3.3.1 Two Ultrasonic Sensors

For a set of two ultrasonic sensors, the orientation of each sensor was bounded between  $0^\circ$  and  $90^\circ$  from the longitudinal axis and the range was set to 6m. To simplify the problem, the two sonars were considered to have a symmetrical orientation, resulting in just one design variable. A narrow beam pattern was adopted to reduce computational cost.

The GA minimization terminated at 20 iterations, due to average change in the fitness value less than the specified tolerance, after performing 592 function evaluations for 39 hours and 40 minutes. It reached an optimal orientation of  $36.5^\circ$  (see Fig. 7).

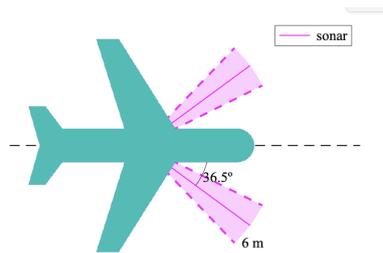


Figure 7: Optimal orientation for two ultrasonic sensors configuration.

The results, summarized in the first line of Tab. 3, are not satisfactory for either case, since the sensor's scanning pattern allows for the safety radius to be breached too many times. This was expected due to the short range of ultrasonic sensors, that makes it impossible for the UAV to detect the obstacle and replan its trajectory in a timely manner.

### 3.3.2 Two Laser Rangefinders

Analogous to the previous case, a set of two laser rangefinders with symmetrical orientation was considered, but adopting a sensing range of 100 m.

After 19 generations, the GA optimization algorithm finished, corresponding to 564 function evaluations and a computing time of approximately 6 hours. The optimal sensor orientation was  $34.4^\circ$ , which corresponds well with one of the approximate minimum shown in the preliminary study in Fig. 6. The optimal two laser rangefinder sensor configuration is illustrated Fig. 8.

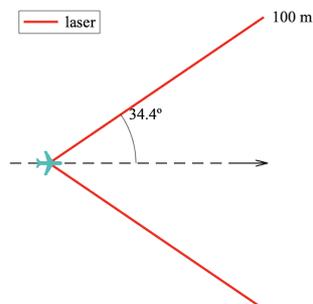


Figure 8: Optimal orientation for two laser rangefinder configuration.

The performance of this optimal configuration is summarized in the second line of Tab. 3. Although the optimal configuration only fails once in 40 scenarios, the safety radius was breached in 23 of them. This result was expected, since a UAV equipped only with two laser rangefinders is not capable of properly tracking the moving obstacles when collisions are imminent.

Compared to the previous case of ultrasonic sensors, these simulations demonstrated that laser rangefinders not only prevent more collisions but also more close calls. Overall, these sensors perform better under the given circumstances.

### 3.3.3 Two RADARs

Once again, the two RADAR sensors were considered to be symmetrical about the UAV longitudinal axis and the orientation spanned from  $0^\circ$  to  $90^\circ$ . Each RADAR had a range of 50 m, an accuracy of 0.04 m and a FOV of  $43^\circ$ .

After 11 generations, the optimizer finished 340 function evaluations. The optimal RADAR orientation was  $9.2^\circ$ , as illustrated in Fig.9.

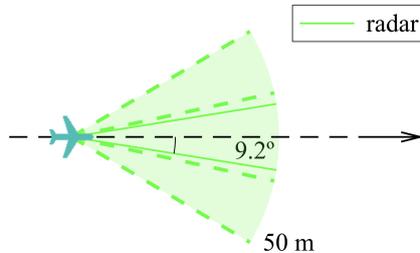


Figure 9: Optimal orientation for two RADAR configuration.

Another configuration worth studying would be a sensor orientation close to  $21.5^\circ$ , which would yield the same result as if the UAV were equipped with a single RADAR with double FOV ( $86^\circ$ ). Table 3 includes the comparison between this configuration, the optimal orientation and a single RADAR pointing forward.

Regarding actual collisions, obstacles that approach the UAV from an angle are more likely to be detected by the optimal solution rather than by the single RADAR configuration. As can be seen in Tab. 3, the number of failures increase as the orientation decreases (for this particular case), which in turn makes the success rate decrease.

By overlapping the FOV of the two sensors, the accuracy is reduced through the data fusion algorithm. Thus, in this case, having a narrower FOV ( $\beta = 9.2^\circ$ ) and in turn, the juxtaposition of both RADARs proved to be almost as effective as the double FOV configuration ( $\beta = 21.5^\circ$ ).

These simulations showed that the reduced accuracy of the RADAR proves to be impactful on the precision of obstacle tracking compared to that of the laser sensors. Despite having a broader FOV and resulting in less close calls, the RADAR solution led to just as many collisions, which means that the two laser rangefinder configuration remains as promising (same success rate). It is reasonable to say that while RADAR FOV is more crucial for detecting obstacles, the sensor's accuracy is the most significant factor for effective collision avoidance.

### 3.3.4 Two LIDARs

Each LIDAR was modelled with a range of 45 m, an accuracy of 0.1 m and a variable FOV. According to hardware specifications (see Tab.1), this FOV can range from  $20^\circ$  to  $320^\circ$ , thus, a FOV of  $180^\circ$  was chosen. This value ensures a reasonable trade-off between timely scanning frequency and a broad scope.

However, this makes optimization redundant, as illustrated in Fig. 10. This happens due to the nature of the scenario generation algorithm used: because the obstacles are spawned inside the limits of the scenario, it is worthless to track the area behind the UAV in the initial instant. Furthermore, from this instant on, if an obstacle were positioned behind the UAV, it would have already been tracked before due to the wide FOV and long range of the LIDAR. The overlapping of the FOV in the case of a two LIDAR solution does not prove to be advantageous either. Note that this is only verified for a FOV of  $180^\circ$ . If the FOV were smaller, it would be convenient to optimize the sensor orientation.

In this particular case, it is fair to state that the most beneficial solution would be to use a single LIDAR pointing forward, since it decreases hardware cost. This configuration is illustrated in Fig. 11.

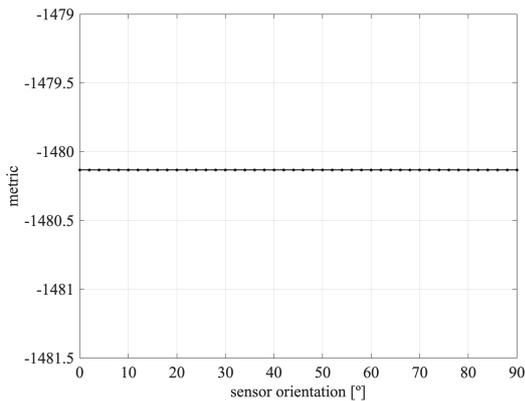


Figure 10: S&A metric as function of sensor orientation for a set of two LIDAR.

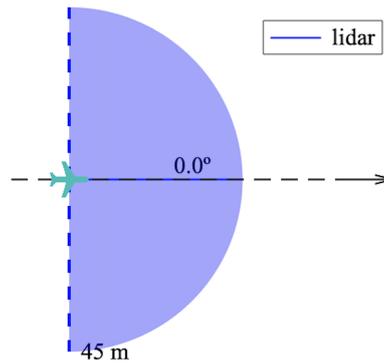


Figure 11: Single LIDAR configuration.

Table 3 includes the performance comparison for different orientations of two LIDAR. As mentioned above, the success rate is the same for both cases. Compared to the previous types of sensors studied, the LIDAR performs better overall. The wide FOV reduces the chances of close calls and eliminates the possibility of failure.

### 3.3.5 Performance Comparison of Sensor Sets

Other solutions that involved three sensors were optimized, for example, including two laser rangefinders symmetrical about the UAV longitudinal axis, whose orientations were bonded between  $0^\circ$  and  $70^\circ$ ; and one fixed RADAR pointing forward. This configuration was also replicated with two lasers and one LIDAR, two RADARs and one laser, and 2 RADARs and one LIDAR. The performance of the optimal version of these sets of sensors is summarized in Tab. 3, as well as the results from the solutions with only one type of sensor. Optimizations with different sets of sensors were performed but left out of this table in order to avoid redundancy of results.

Table 3: Comparison of the optimal performance for the different sensor sets studied.

Sensors	Metric	Failure	Close call	Success rate
2 SONARs @ 36.5 °	804.0	4/40	30/40	90.0%
2 lasers @ 34.4 °	-414.0	1/40	23/40	97.5%
2 lasers @ 63.4 ° + 1 RADAR @ 0 °	-1240.4	0/40	11/40	100.0%
2 lasers @ 10.0 ° + 1 LIDAR @ 0 °	-1606.4	0/40	8/40	100.0%
2 RADARs @ 9.2 °	-1171.0	1/40	12/40	97.5%
2 RADARs @ 21.5 °	-1141.7	1/40	12/40	97.5%
2 RADARs @ 35.3 ° + 1 laser @ 0 °	-1480.1	0/40	9/40	100.0%
2 RADARs @ 28.1 ° + 1 LIDAR @ 0 °	-1574.3	0/40	9/40	100.0%
1 LIDAR @ 0 °	-1480.1	0/40	9/40	100.0%

For the set of scenarios tested, the RADAR performed better than the laser rangefinder, which in turn performed better than the ultrasonic sensor if only one sensor type is to be used. Nonetheless, this is tightly dependent on the sensor characteristics, such as range, FOV and accuracy. Furthermore, a single LIDAR was enough to outperform all other types of sensor.

As expected, all the solutions that present a 100% success rate include either a RADAR or a LIDAR in their configuration. If the LIDAR is kept out, it is the two RADAR and one laser rangefinder solution that produced the least collisions and led to the least close calls. From these findings, and because of the 0° FOV of a laser rangefinder, it is expected that increasing even more the number of sensors would lead to even better performance, thought at a higher hardware cost.

Comparing the solutions that include a LIDAR, it is proved that it is not significantly advantageous to pair it with other types of sensors, since it already performs distinctively well on its own. Regardless, the two laser and the two RADAR solution are beneficial due to reducing the likelihood of close calls. Despite the LIDAR having a wide FOV that is not increased by either configuration, the chances of breaching the safety radius decrease because the other sensors provide additional detection capacity. *I.e.*, since the LIDAR sweeps the designated area at a certain frequency, there are time instants when a fraction of the area within the LIDAR FOV is 'unsupervised'. Therefore, it is useful to have another set of sensors that track obstacles approaching from that specific area.

To summarize, the optimized configuration had a very similar performance in four different cases (reflected in the *Metric* column), being the most promising one composed of one LIDAR pointing forward, complemented by two laser rangefinders pointing at 10° sideways. These four configurations are illustrated in Fig. 12.

## 4 HARDWARE AND SOFTWARE IMPLEMENTATION

This section provides an introduction to the flight controller and ground control station, including the steps that must be taken to build and adapt this software to the current work, and the electrical layout of the final system.

Some basic concepts are needed in order to build and fly an unmanned vehicle using PX4 [21]. PX4 is a core part of a broader drone platform that includes the QGroundControl ground station, the Pixhawk hardware, and MAVSDK for integration with companion computers, cameras and other hardware using the MAVLink protocol.

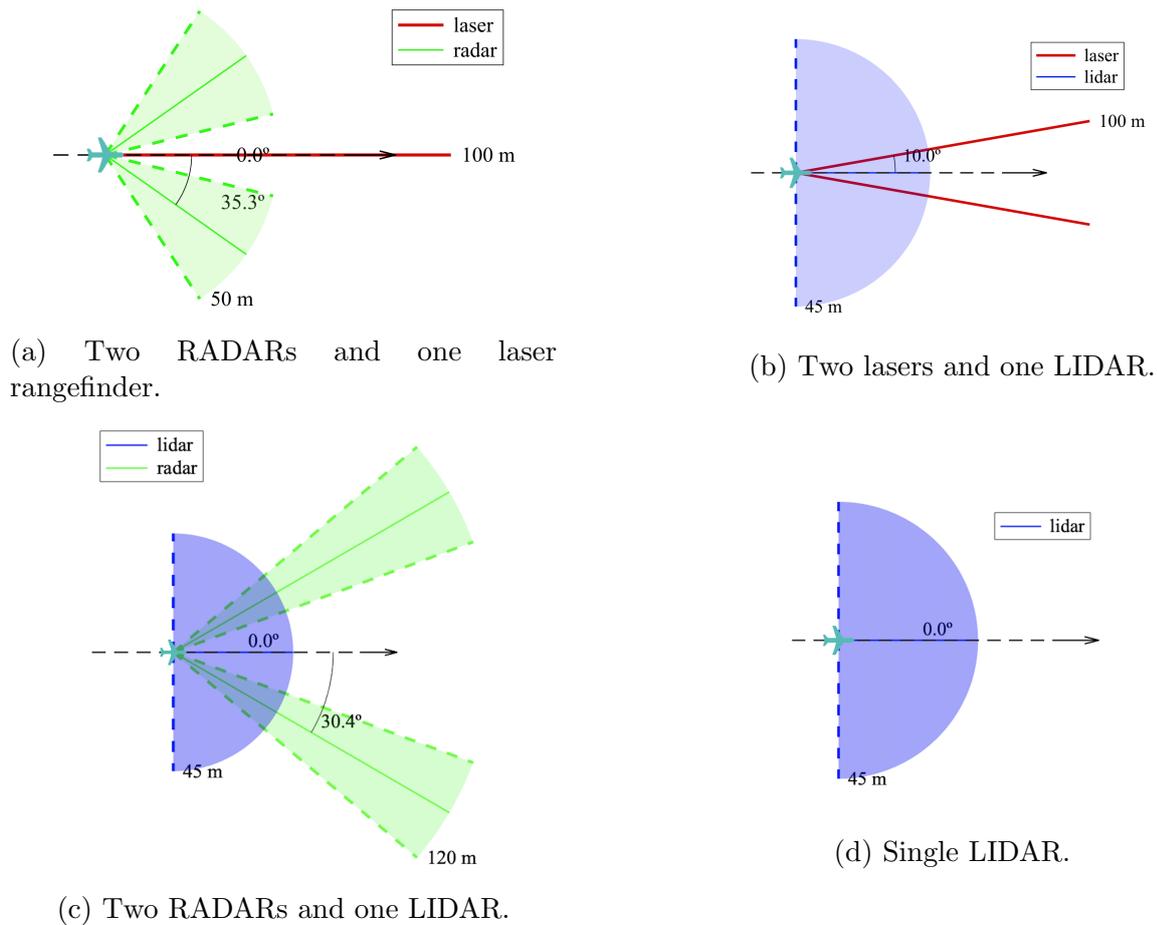


Figure 12: Optimal sensor configurations to be tested.

#### 4.1 Flight Controller

PX4 is a powerful open source autopilot flight stack that can be built on a console or in an IDE, for both simulated and hardware targets. PX4 can be run on various hardware platforms, including Pixhawk. The Pixhawk 2.1, or Hex Cube Black, was the chosen controller in [7], so it will also be used in the current work. Generally, the most recent stable released version of PX4 ought to be used, to benefit from bug fixes and get updated features. As such, this is the version that is installed by default and integrated into QGroundControl. However, the current stable release (v1.13.3) does not include the driver for the LIDAR used in this work. Consequently, it was necessary to switch to a more recent beta release (v1.14) that includes it. The PX4 source code is stored on a Github repository called PX4/PX4-Autopilot. To get release 1.14 and enable the necessary drivers, the PX4 Developer Guide includes tutorials on Building PX4 Software [22] and PX4 Board Configuration [23].

Once these steps are completed, the PX4 Autopilot firmware will be compiled, generating an executable file that can be uploaded onto the flight controller.

#### 4.2 Ground Control

A ground control station works as a virtual cockpit, serving as an interface between a flight controller and a human operator. Typically, a software running on a computer

is connected to the flight controller through wireless telemetry. This enables the human operator to communicate with the aircraft during flight, allowing the acquisition of relevant data such as position, velocity, acceleration, or any other sensor data. It can be installed simply by running the executable file available in the QGroundControl online user manual [24].

The PX4 firmware can be installed onto the flight controller by following the steps in the PX4 User Guide [25]. The user is then prompted by QGroundControl to calibrate the vehicle, including the configuration of the controller's built-in sensors, radio receiver, flying modes, power, and motors.

### 4.3 Electrical Wiring Layout

The electrical layout can be designed once the hardware is chosen and calibrated.

It is possible to connect all the components as shown in Fig. 13 using the connections and supplementary devices (GPS and power module) included within the Cube Black package. It is necessary to employ a power module to provide the flight controller a regulated power source and power the electronic speed controller (ESC) at the same time. The ESC also draws power from a battery and operates the motor using a PWM signal from one of the PWM I/O entries. A PPM Sum Receiver is also present, and it needs to be connected to an RX IN input. This component converts the PWM signals from the radio receiver into a single PPM signal that the flight controller can process. The telemetry module communicates through radio waves with a second telemetry module that is linked to a ground station. This allows real-time data to be exchanged and orders to be sent to the vehicle. Lastly, the I2C ports can be used to connect sensors, namely the ultrasonic sensor and the laser rangefinder.

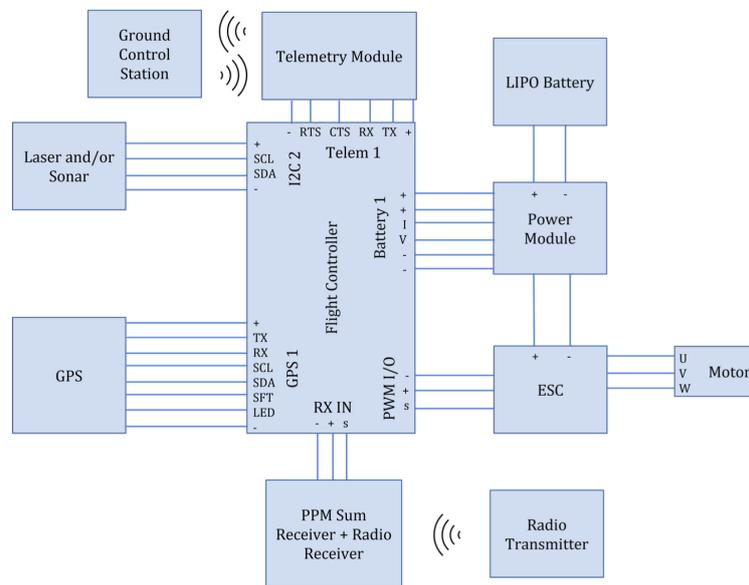


Figure 13: Electrical wiring diagram [7].

## 5 SENSOR BENCH TESTS

To validate the capabilities of a S&A system, several experiments were performed. More specifically, the sensors that integrate the optimized configurations found in Sec. 3 were individually tested before the complete system. Due to the risk associated with flight testing, experiments were based on ground tests using a simple rover. Once the system is validated under these conditions, flight tests using a small fixed-wing UAV may begin.

### 5.1 Ultrasonic Sensor

The MB1242 ultrasonic sensor must be connected to the Pixhawk 2.1 and activated in QGroundControl in order to perform the bench tests [7]. These tests included variations in material of the detected obstacle and angles. Figure 14a) demonstrates an experiment where the object to detect is in front of the sensor. In Fig. 14b), the idea is to determine the sonar capability of detecting an object which has an angular deflection ( $\theta$ ) in relation to the sensor.



Figure 14: Ultrasonic sensor bench tests.

For these first experiments, the target object is a rectangular wooden board with size 30x25cm as seen in Fig. 15. The frontal test was also repeated with a rectangular XPS board (125.5x60cm) as the target object.



Figure 15: Experimental setup.

During these experiments, each board was positioned at different distances relative to the sonar (20cm to 760cm) and the sensor data was recorded for 30 seconds for each position. This method was aimed at determining, for each position, the fraction of time

where the sensor was actually detecting its target and how much these measurements were deviated from the correct distance.

It is important to consider that the target material could affect the performance of this sensor. The MB1242 data sheet [9] mentions that this sensor's ideal surface to detect is hard, smooth and non-porous. Although wood is not a perfect example of an ideal surface, its properties are not far from that category. To determine whether these conclusions are applicable to other materials. A target made of XPS was also tested, which is not as hard as wood, yet is more porous. Figure 16 represents the detection rate for both materials. When testing with a XPS board, the maximum range decreased and the sensor performed worse overall. This decrease in performance was foreseeable since XPS's properties do not match those of an ideal material (hard, smooth and non-porous).

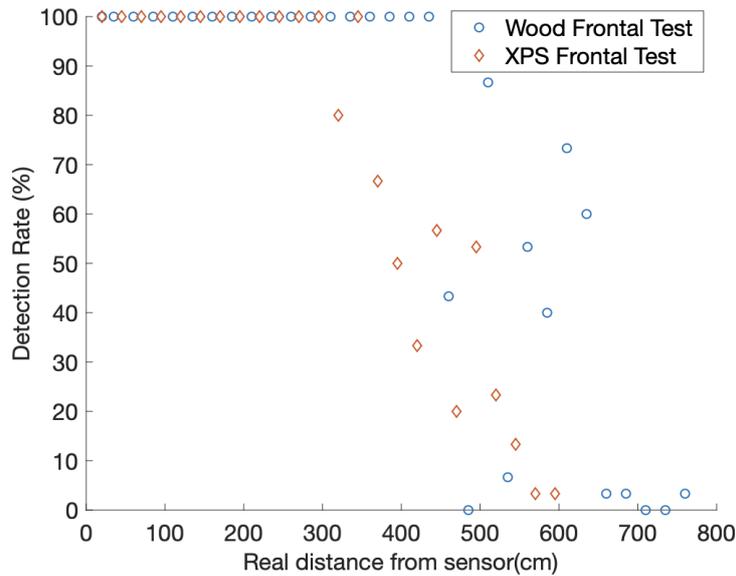


Figure 16: MB1242 detection rate for different materials.

The last significant point is that the performance of this sensor is impacted by the target's rotation within its inertial referential. Only when the sound is reflected back from the target can the sonar identify it. According to the principles of sound reflection, this is only conceivable if the normal vector of the surface in question is parallel to the trajectory of the sound being emitted until it reaches the desired target.

Empirically, this translates to the results that follow. Figure 17 shows the sensor's detection rate for various distances and orientations. As expected, the sensor performed better when the obstacle was completely in front of it, achieving a maximum range of 435 cm with perfect a detection rate, although the datasheet states 640 cm. Additionally, the maximum range decreased when augmenting  $\theta$ , which was also an expected behaviour. Moreover, this sensor proved to be very directional as it stopped detecting any targets for  $\theta \geq 40^\circ$ .

Figure 18 shows the average absolute error at each distance from the sensor. No relation between the real distance from the sensor and its error was detected, given that all points seem to be almost randomly dispersed from 300 cm onwards. Moreover, the average absolute error never surpassed 30 cm. This error occurs in the wood frontal test, which leads to the conclusion that this board might not match the ideal surface for this type

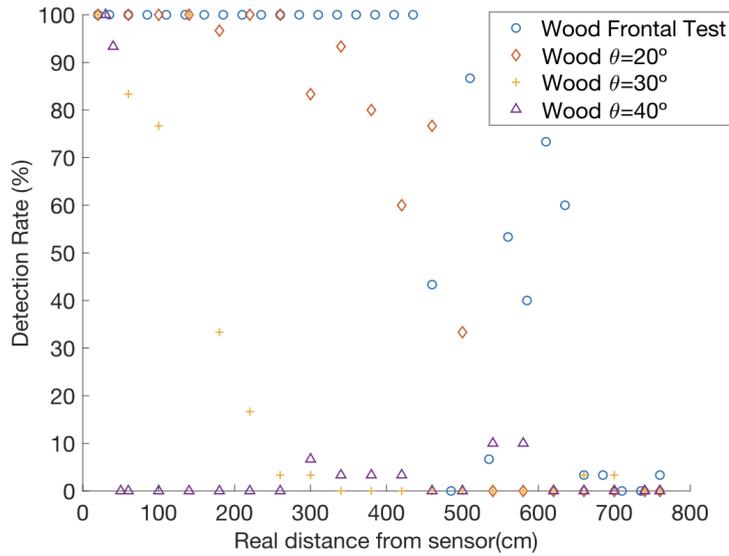


Figure 17: MB1242 detection rate for different angles of incidence.

of sensor to detect. However, the collision avoidance system needs to be robust enough for most materials, *i.e.*, if the available sensor fails this bench test, it is not sufficiently accurate for the purpose of this work.

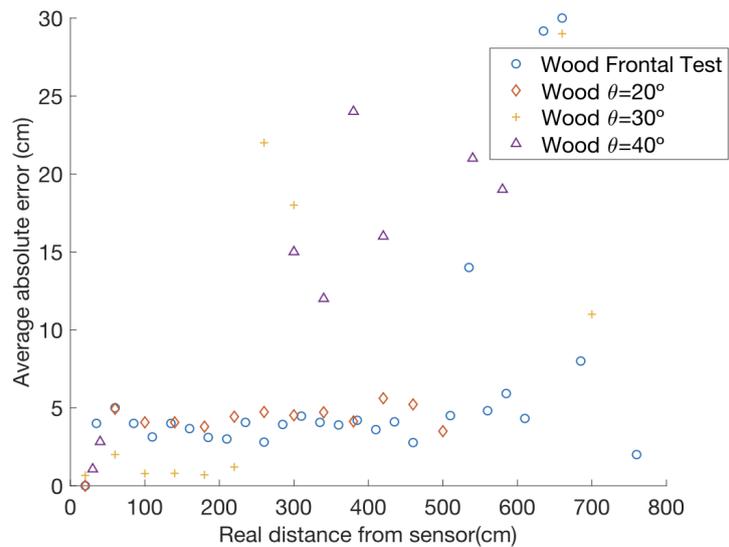


Figure 18: MB1242 average absolute error for different angles of incidence.

According to the datasheet [9], the sensor is calibrated and tested to provide stable range readings to large targets even in electrically and acoustically noisy environments. It also states that the sonar should ideally be used indoors. Nonetheless, it is important to note that these tests were done outdoors, due to the nature of this project. Since this sensor is intended to integrate a collision avoidance system for small UAVs, its applications require that it performs well outdoors. However, this also means that the results shown can and have been affected by exterior noise.

Additionally, an experimental beam pattern was generated using the maximum range recorded for each orientation, shown in Fig. 19. Such beam patterns tend to be particu-

larly advantageous for S&A systems since they restrict their detecting volume, ultimately allowing the controlling device to pinpoint the target's location with a high degree of accuracy.

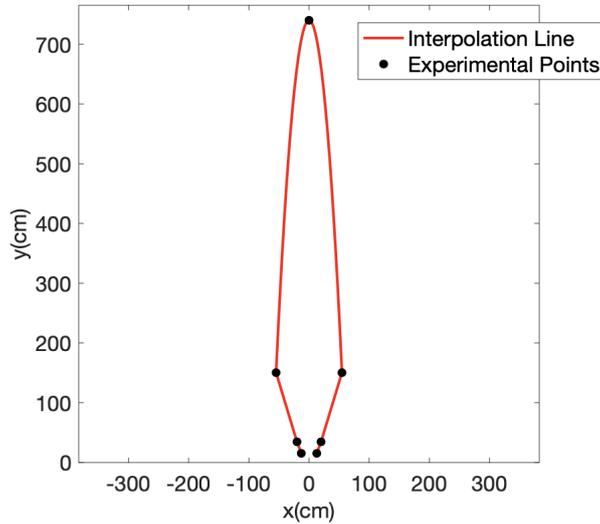


Figure 19: MB1242 empirical beam pattern.

## 5.2 Laser Rangefinder

Prior to testing, it was necessary to configure the LW20/C laser rangefinder within the flight controller's environment [7]. Once the configuration was done, the performance of the sensor could be assessed through an identical experience to that of the sonar. However, since the laser rangefinder is completely directional, it is not necessary to experiment with off-set obstacles.

In frontal tests, the laser maintained a perfect detection rate before reaching 85 m, as seen in Fig. 20. From this distance onward, the detection rate decreased non-linearly until it reached 100 m (marked as a dashed red line in Fig. 20). Ultimately, the complete range promised in the datasheet was not attained with a perfect detection rate.

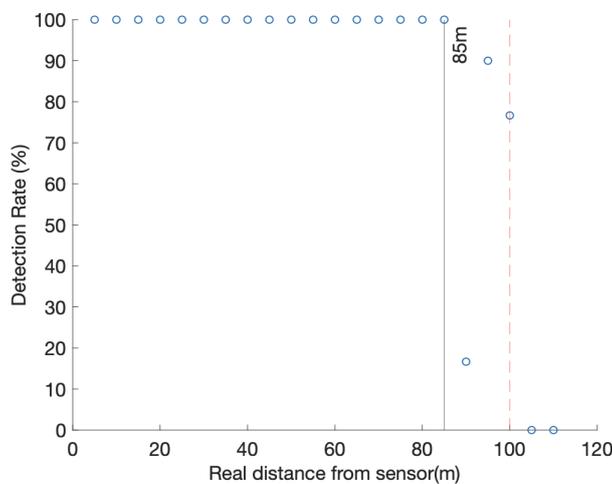


Figure 20: LW20/C detection rate.

These tests also served to prove how much of an impact directionality has on this type of sensor. The wooden board had to be perfectly aligned with the laser rangefinder in order for it to detect it correctly. When translating this to the optimal sensing system designed in Sec. 3, it means the sensor has to be flawlessly aligned with the UAV's longitudinal axis.

Lastly, the average absolute error, plotted in Fig. 21 was mostly between 0 and 50 cm, but increased with the distance from the sensor. The results were not as satisfactory for distances greater than 50m.

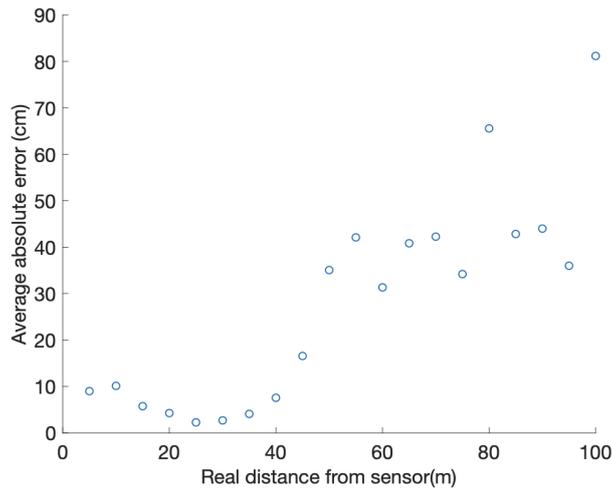


Figure 21: LW20/C average absolute error.

### 5.3 LIDAR

The chosen LIDAR was the SF45/B model by LightWare [11]. It can be connected to the flight controller's TELEM2 port (TX and RX pins) using a DF13 header. The red and black wires (VCC and GND) were connected with an external power supply and the remaining three wires (blue, white and green) were left unconnected. Figure 22 illustrates the electrical wiring diagram for this configuration.

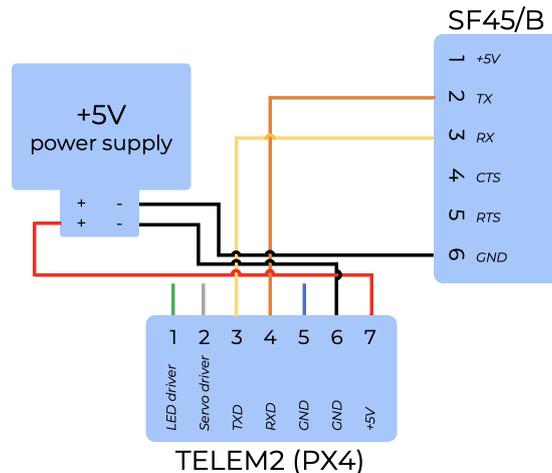


Figure 22: Eletrical wiring diagram for SF45/B.

The LIDAR also comes with a micro USB port that connects to any PC running the LightWare Studio application for visualisation of results, making configuration changes and upgrading the firmware. The sensor should be configured by setting the scanning angle limits to  $-90^\circ$  and  $90^\circ$ , the baud rate to 921600 and the update rate to 50 Hz. It is also important to make sure 'scan upon startup' is enabled.

To activate this sensor within the QGroundControl environment, it is necessary to follow the instructions in Sec. 4. After that is assured, the user can access the vehicle setup section and, within the parameters tab, set SENS\_EN\_SF45\_CFG to the desired serial port (TELEM2).

In the bench tests, the angles of the detected obstacle and scanning speed of the sonar were varied. Figure 23 shows the resulting scans from two different experiments: a) the object to detect is in front of the sensor; and b) the object to detect is at  $90^\circ$  in relation to the sensor. In these tests, the forementioned rectangular XPS board (125.5x60cm) was used as target.

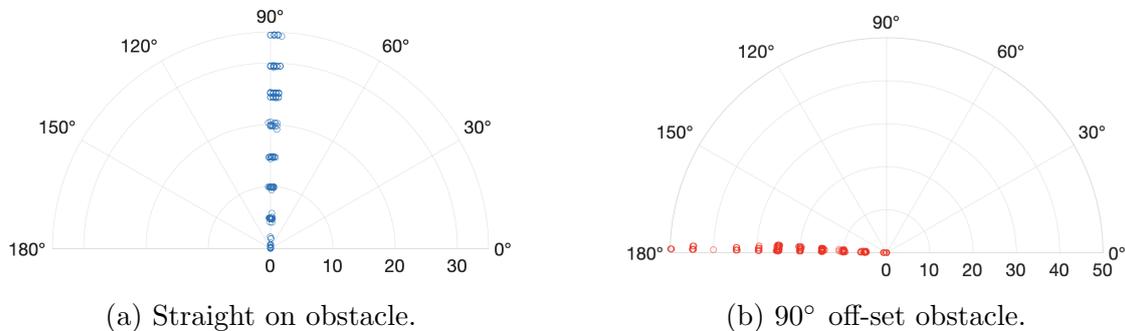


Figure 23: LIDAR bench tests.

On an unobstructed rugby field, the XPS board was positioned at different distances relative to the LIDAR (0 to 50m) and the sensor data was recorded for 30 seconds for each position. As expected, this sensor performed better than the others, maintaining a perfect detection rate through all its range in both experiments, as seen in Fig. 24. However, Fig. 25 shows that the average absolute error was overall lower when the obstacle was aligned with the sensor. This is likely because the LIDAR scans back and forth from  $-90$  to  $90^\circ$ , meaning that for each sweep, it passes twice through  $\theta = 0^\circ$  and only once through each limit.

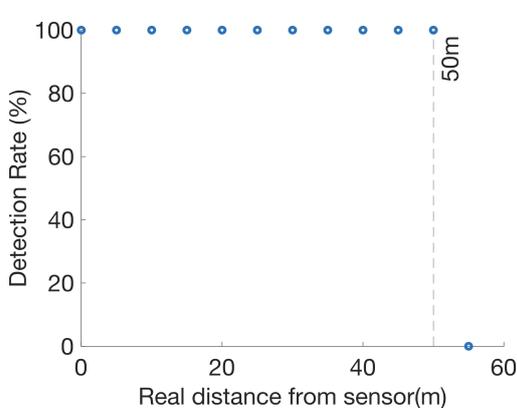


Figure 24: SF45/B detection rate.

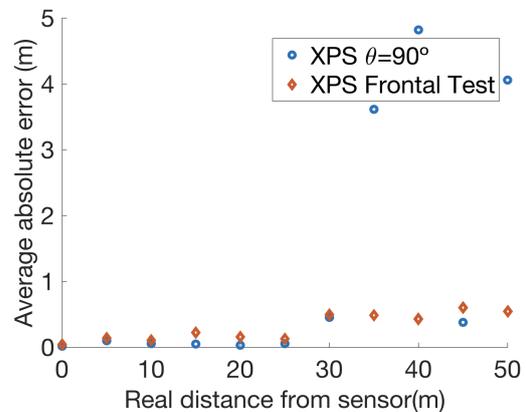


Figure 25: SF45/B average absolute error.

Although the SF45's update rate was set to 50Hz, empirically, it is, on average, 37.2Hz. In LightwareStudio, it is also possible to calibrate the sensor's cycle delay, which is inversely proportional to its scanning speed. The minimum cycle delay (5) corresponds to the maximum scanning speed (6.3 rad/s) and vice-versa. This implies that, by choosing a higher sweep speed and maintaining the angle limits, the arc of circle that is not being detected between each measurement increases. Figure 26 illustrates how the length of the arc traversed varies analytically with the distance to the sensor and the angular velocity. This graphic shows that, although the LIDAR has a 50m range, at the maximum scanning speed, it might not be possible to detect an obstacle less than 8m wide at this distance. When the scanning speed is reduced, the sensor is likely to detect a target of at least 2.2 m at maximum distance. At minimum speed, this stops being relevant within the 50 m range. However, if covering a larger area quickly is more important, sacrificing some visibility at the maximum range might be acceptable. Ultimately, the compromise should be based on the specific needs and constraints of the system.

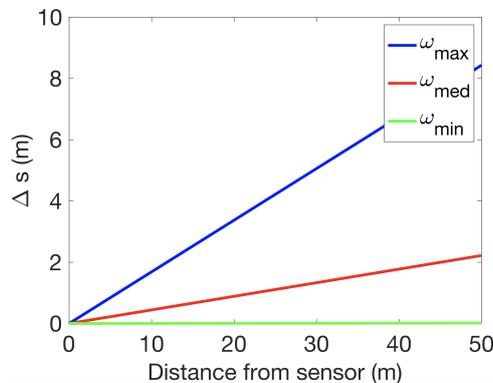


Figure 26: SF45/B undetectable arcs for different scanning speeds.

## 6 CONCLUSIONS

This work presents a comprehensive solution for enhancing the safety of small fixed-wing UAVs by addressing the critical issue of obstacle detection during flight. A set of select sensors, namely the ultrasonic sensor, laser rangefinder, LIDAR, and RADAR, were identified and further employed in modeling collision detection and avoidance simulations using the potential fields method.

To determine the best combination of sensors and their orientations, these simulations were used in an optimization study. The study revealed that relatively simple detection configurations can yield a high success rate in collision avoidance. While the ultrasonic sensor is found to be inadequate due to its limited range, the laser rangefinder benefits from a long range, but has a restricted field-of-view. On the other hand, both the LIDAR and RADAR prove to be the most promising options, offering not only a substantial range but also a wide field-of-view. Based on the optimization study, the recommended multi-sensor configurations consist of a front-facing LIDAR or RADAR, accompanied by a pair of laser rangefinders pointing sideways at either a 10 or 63 °angle.

To validate the proposed system, the necessary hardware and software were successfully implemented, which allowed for the individual testing of each sensor. The bench tests confirmed the accuracy of the sensors specifications and previous simulations. In the case

of the ultrasonic sensor, the importance of the material and the angular deflection of the obstacle to be detected was highlighted. As for the laser rangefinder, the key factor proved to be directionality. The LIDAR presented less shortcomings, as expected. However, the sensor's parameters (update rate, angular velocity and scan angle limits) directly affected its performance. More specifically, it is necessary to reach a compromise between the LIDAR scan speed and the effective range of visibility.

In the future, these sensors will be integrated into a multi-sensor configuration to be tested on a rover. Subsequently, the system's performance will be evaluated under realistic conditions through flight testing. Overall, this work provides a comprehensive methodology for testing and validation of an optimized multi-sensor system configuration and the proposed system holds great potential for enhancing the safety of small fixed-wing UAVs during flight.

## ACKNOWLEDGEMENTS

This work was supported by FCT, through IDMEC, under LAETA, project UIDB/50022/2020.

## REFERENCES

- [1] S. A. H. Mohsan, M. A. Khan, F. Noor, I. Ullah, and M. H. Alsharif. Towards the Unmanned Aerial Vehicles (UAVs): A Comprehensive Review. *Drones*, 6(6), 2022. doi:10.3390/drones6060147.
- [2] Commercial UAV Market Share, Size, Trends & Industry Analysis Report By Type; By End-Use; By Region; Segment Forecast, 2021 - 2028. *Polaris Market Research*, 2021. Accessed: 05/10/2022.
- [3] J. N. Yasin, M.-H. Haghbayan, M. M. Yasin, and J. Plosila. Swarm formation morphing for congestion-aware collision avoidance. *Heliyon*, 7(8), August 2021. doi:10.1016/j.heliyon.2021.e07840.
- [4] Next Generation Air Transportation System (NextGen). <https://www.faa.gov/nextgen>. Accessed: 10/10/2022.
- [5] Title 14 Code of Federal Regulations (CFR) Part 91.113 and RTCA. <https://www.govinfo.gov/content/pkg/CFR-2007-title14-vol1/html/CFR-2007-title14-vol1.htm>, 2007. Accessed: 05/10/2022.
- [6] N. Alturas. Modeling and Optimization of an Obstacle Detection System for Small UAV's. Master's thesis, Instituto Superior Técnico, Lisboa, Portugal, January 2021.
- [7] P. Serrano. Optimization of Obstacle Detection for Small UAVs. Master's thesis, Instituto Superior Técnico, Lisboa, Portugal, June 2022.
- [8] N. Alturas and A. Marta. Modeling and Optimization of an Obstacle Detection System for Small Fixed-wing UAV. In *Aerobest 2021 - ECCOMAS Thematic Conference on Multidisciplinary Design Optimization of Aerospace Systems*, Lisboa, Portugal, July 2021. ISBN:978-989-99424-8-6.

- [9] MaxBotix I2CXL-MaxSonar-EZ Datasheet. <https://maxbotix.com/pages/i2cxl-maxsonar-ez-datasheet>. Accessed: 13/04/2023.
- [10] LightWare LW20 LiDAR sensor Datasheet. <https://www.documents.lightware.co.za/LW20%20-%20LiDAR%20Manual%20-%20Rev%2012.pdf>. Accessed: 13/04/2023.
- [11] LightWare SF45/B product guide. <https://support.lightware.co.za/sf45b/#/specs>. Accessed: 13/04/2023.
- [12] Ainstein US-D1 Data Sheet. <https://ainstein.ai/wp-content/uploads/US-D1-Data-Sheet.pdf>. Accessed: 13/04/2023.
- [13] L. Yang, X. Feng, J. Zhang, and X. Shu. Multi-ray modeling of ultrasonic sensors and application for micro-UAV localization in indoor environments. *Sensors*, 19(8): 1770, 2019. doi:10.3390/s19081770.
- [14] M. Schirrmann, A. Hamdorf, A. Giebel, F. Gleiniger, M. Pflanz, and K.-H. Dammer. Regression kriging for improving crop height models fusing ultra-sonic sensing with UAV imagery. *Remote Sensing*, 9(7):665, 2017. doi:10.3390/rs9070665.
- [15] J. Saunders, B. Call, A. Curtis, R. Beard, and T. McLain. Static and dynamic obstacle avoidance in miniature air vehicles. In *Infotech@ Aerospace*, page 6950. Provo, UT, USA, September 2005.
- [16] F. Fayad and V. Cherfaoui. Tracking objects using a laser scanner in driving situation based on modeling target shape. *2007 IEEE Intelligent Vehicles Symposium*, pages 44–49, 2007. doi:10.1109/IVS.2007.4290089.
- [17] M. S. Grewal, A. P. Andrews, and C. G. Bartone. *Kalman filtering*. Wiley Telecom, second edition, 2020. ISBN:0-471-26638-8.
- [18] M. Skowron, W. Chmielowiec, K. Glowacka, M. Krupa, and A. Srebro. Sense and avoid for small unmanned aircraft systems: Research on methods and best practices. *Journal of Aerospace Engineering*, 233(16), June 2019. doi:10.1177/0954410019867802.
- [19] M. Longbin, S. Xiaoquan, Z. Yiyu, S. Z. Kang, and Y. Bar-Shalom. Unbiased converted measurements for tracking. *IEEE Transactions on Aerospace and Electronic Systems*, 34(3):1023–1027, 1998. doi:10.1109/7.705921.
- [20] N. Gageik, P. Benz, and S. Montenegro. Obstacle Detection and Collision Avoidance for a UAV With Complementary Low-Cost Sensors. *IEEE Access*, 3:599–609, 2015. doi:10.1109/ACCESS.2015.2432455.
- [21] PX4 Autopilot User Guide. <https://docs.px4.io/main/en/>, . Accessed: 27/10/2022.
- [22] PX4 Building Software. [https://docs.px4.io/main/en/dev\\_setup/building\\_px4.html](https://docs.px4.io/main/en/dev_setup/building_px4.html), . Accessed: 20/05/2023.

- [23] PX4 Board Configuration (Kconfig). [https://docs.px4.io/main/en/hardware/porting\\_guide\\_config.html](https://docs.px4.io/main/en/hardware/porting_guide_config.html), . Accessed: 20/05/2023.
- [24] QGroundControl User Guide. <https://docs.qgroundcontrol.com/master/en/index.html>. Accessed: 25/01/2023.
- [25] PX4 Loading Firmware. <https://docs.px4.io/main/en/config/firmware.html>, . Accessed: 20/05/2023.