Design of a Quadcopter's Controller to track down zebu thieves

Tsimitamby Briand[1,2] , Sambatra Eric Jean Roy[1,2] , Andrianirina Charles Bernard[2,3]
[1] Higher Institute of Technology Antsiranana, Department of Industrial Engineering, Madagascar.
[2] Doctoral School in Renewable Energy and Environment, University of Antsiranana, Madagascar.
[3] Higher Institute of Science and Technology, Mahajanga, Madagascar.

Abstract:
This paper focuses on designing a controller of Quadcopter to track down the cattle rustlers called "Dahalo" in Madagascar. The purpose of this study is to assist the Malagasy police forces to detect or to locate with a better geographical precision the rustled cattle during transport. In fact, the traditional method consists of following their footpaths, which slows the search and increases the risk of losing their tracks. Hence, an analysis of "Dahalo" dynamic behavior and an estimation of their trajectories while transporting the rustled zebu were performed in previous studies. Thus, after generating the flight plan, from the location of theft to the destination place, a model of Quadcopter and a controller adapted for tracking down Dahalo have been designed and embedded in this device. Through this program and with use of its onboard camera, the drone can automatically locate and display in real time the movements of the criminals. The trajectory was embedded in digital model of the Drone. The simulations have been validated with the parameters of an Ar Drone 2.0. The results show that the designed controller follows efficiently the set flight plan.

Keywords: Drone, Quadcopter, Controller, Dahalo, Trajectory

1. Introduction

The Dahalo still remains a prime public enemy in Madagascar. These bandits who specialize in the rustling of zebus cattle cause trouble and terror, ravage the villagers and rustle zebus from a few heads to several hundred, in a just single attack. This results in extreme poverty for the local population as well as for the state [1] .To deal with this situation, studies have been conducted to assist the police to track down these criminals [2] [4] [5] . [2] has revealed the strategic routes of these rustlers, their average travel speed and their behaviors when transporting zebu. They make shortcuts on unusual paths. These strategic routes could be through water, forest or mountain. Their speed is estimated approximately at 8km/h. They move along in armed group, and carry with them" Mohara "(talismans). They set up watchtowers to monitor the roads. During the attack, the older bring the rustled zebu and the young people face the police forces and/or villagers. Then, these Dahalo proceed to a single point where the regularization of the papers of these rustled animals takes place. Until now, this place has remained impenetrable even for the police. [4] allowed us to estimate probabilistically their trajectory from the place of rustling to the laundering location. [5] provided us with the optimized flight plan by reducing the number of possible routes , evaluating the position of the cattle and Dahalo before the chase and returning to the base in case of detection, energy limit or reaching the control-distance limit point.

After obtaining the trajectory, we propose in this paper the use of a Drone in order to accelerate the tracking. Currently the research on this remote-controlled flying device is in full expansion whether for universities or industrials' purpose. Originally used for military purposes, this technology is currently used in several sectors such as rescue, inspection and detection, observation and monitoring, agriculture, home delivery, etc. This is due to its ability to take off and land vertically, its small size, its light weight , and its lower cost and its maneuverability and handling [7] [8] [9] . Moreover, the Quadcopter is useful in risky places, where it is difficult or even impossible for people to access.

The Quadcopter has six degrees of freedom, along the x, y, z axis as well as the angles of roll, pitch and yaw, controlled only by four variables: roll angle, pitch angle, yaw angle and altitude. It is thus an under-powered and dynamically unstable system [10].

In recent years, the control of this Ultra Light Plane has become a subject of interest to many scholars and a lot of literature related to it has been published [11]. Indeed, in his paper , Nizar

Hadi Abbas [8] uses the " PID tuning " to control the x, y, z positions and the angles of roll, pitch and yaw . PID tuning "is a Matlab tool that allows the adjustment of the gains of PID (Proportional Integral Derivative) controllers. Sevkuthan Kurak [7] uses LQ (Linear Quadratic control) and LQG (Linear Quadratic Gaussian control) to stabilize the Quadcopter. Koszewnik [9] , Viswanadhapalli Praveen and others [10] , [11] , [12] use PID and [1 3] uses PD (Regulator Proportional Derivative) controller . In his thesis [14], Samir Bouabdallah compares several controllers: PID, LQ controller, Backstepping, Sliding -Mode and Integral Backstepping and [15] compares PID and LQR controller (Linear Quadratic Regulator) in his paper. According to these studies, the PID seems to be the most efficient and easy to implement if the flight is quasi-stationary.

In this work, the controlled Quadcopter has a specific mission which is the track of the Dahalo. These Dahalo follow specific courses ensuring both their safety and their ability to feed the zebus [2]. The likely routes are through forests, mountains and water. The Quadcopter moves from one geographic point to another by making a straight line while tracking. The entire path can be broken down into several segments. The controller should consider this movement to stabilize the Quadcopter. Thus, this paper is divided into four sections: the first section introduces the method and the tools that have been used; then the second section deals with the design and the control; the third section presents the results of simulations and discussion; and the last section is the conclusion.

2. Methodology and Tools

Preliminary studies were carried out in [2], [4] and [5], in order to define the trajectory of the Drone. Once the trajectory is generated, four stages have been addressed:

- Dynamic and analytical Modeling of the Quadcopter: a major step to understand the mechanism of the flight. The equations were then implemented in Matlab/Simulink.
- Choice of a controller adapted to the situation. This step allows a theoretical and comparative study of existing Drone controllers.
- Implementation of the controller. This step involves implementing the controller to the digital Drone.
- Simulations, validation of controllers by experimental tests using Ar Drone 2.0, and discussion.

Figure 1 summarizes this methodology.

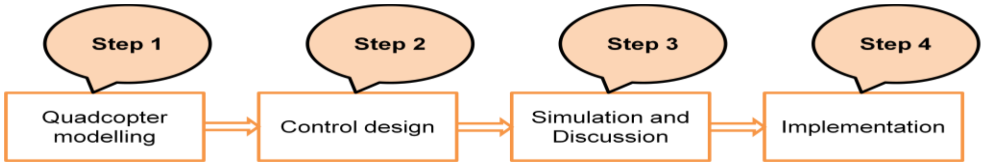


Figure 1: Design Methodology

3. Modeling and control of Quadcopter

3.1 Modeling

Before shifting to control, it is necessary to model the Quadcopter. Its movements are produced by the speed of its four propellers that are driven by four rotors. They are obtained by varying the speed of rotation of each motor. The engine inclines toward the direction of the slowest rotors. Also, it can adjust itself around three axes namely the yaw axis, the axis of the roll and the pitch axis. The yaw is the horizontal rotation movement of a mobile along a vertical axis (yaw axis). The pitch is a rotational movement along the transverse axis of a moving object (pitch axis). The roll is a rotational movement of a moving object along its longitudinal axis (roll axis). Figure 2 shows a general structure and Figure 3 shows its block diagram.

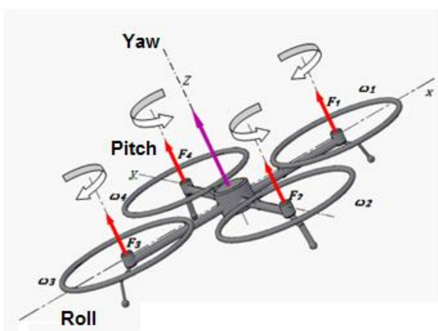


Figure 2: General structure of a Quadcopter

With:

F_i: Lift forces (i varies from 1 to 4)

ω_i: Rotational speeds of each propeller

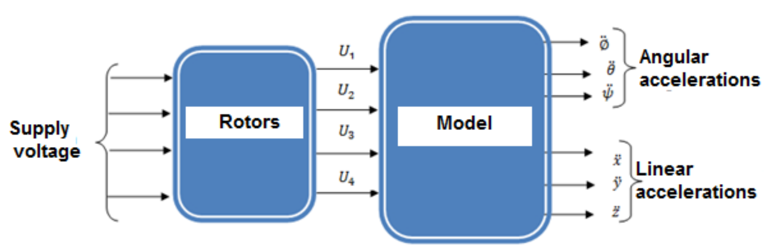


Figure 3: Dynamic Model of Quadcopter

U_1: Sum of liftings

U_2: Moment of roll

U_3: Moment of pitch

U_4: Moment of Yaw

$\ddot{x}, \ddot{y}, \ddot{z}$: A linear acceleration of the Quadcopter according to the fixed reference (linked to the ground)

x, y, z : Position of the Quadcopter according to the x, y and z axis, respectively

$\ddot{\phi}, \ddot{\theta}, \ddot{\Psi}$: Angular acceleration of the Quadcopter according to the fixed reference

ϕ: Roll angle (in radian)

θ: Pitch angle (in radian)

Ψ: Yaw angle of (in radian)

The stability of the flight is achieved by alternately reversing the rotation of the propellers of the engines produced by the Quadcopter. In other words, both rotors located on the same arm rotate in an opposite way [16], [1].

According to Bouabdallah [14] and Young-Cheol Choi [18], the dynamic model of Quadcopter is described in the following formulas:

$$\begin{cases} \ddot{x} = (sin\Psi sin\phi + cos\Psi sin\theta cos\phi)\frac{U_1}{m} \\ \ddot{y} = (-cos\Psi sin\phi + sin\Psi sin\theta cos\phi)\frac{U_1}{m} \\ \ddot{z} = cos\theta cos\phi \frac{U_1}{m} - g \end{cases} \tag{3.1}$$

$$\begin{cases} \ddot{\phi} = \dot{\theta}\dot{\Psi}\frac{(I_{yy} - I_{zz})}{I_{xx}} - \frac{J_r\ \dot{\theta}\ \Omega_r}{I_{xx}} + l\frac{U_2}{I_{xx}} \\ \ddot{\theta} = \dot{\phi}\dot{\Psi}\frac{(I_{zz} - I_{xx})}{I_{yy}} + \frac{J_r\ \dot{\phi}\ \Omega_r}{I_{yy}} + l\frac{U_3}{I_{yy}} \\ \ddot{\Psi} = \dot{\theta}\dot{\phi}\frac{(I_{xx} - I_{yy})}{I_{zz}} + \frac{U_4}{I_{zz}} \end{cases} \tag{3.2}$$

Ω_r: Rotational Speed of Quadcopter (rad/s)

J_r: Inertia Moment of rotor (kg m ^{2})

g: Acceleration of gravity (m/s ^{2})

l: Distance between the rotor and the center of mass (center of gravity of the Quadcopter) (m)

m: Mass of Quadcopter (kg)

I_{xx}, I_{yy}, I_{zz}: Moment of inertia of the Quadrotor along the axes x, y and z (kgm ^{2})

$$\begin{cases} U_1 = b({\Omega_1}^2 + {\Omega_2}^2 + {\Omega_3}^2 + {\Omega_4}^2) \\ U_2 = b(-{\Omega_2}^2 + {\Omega_4}^2) \\ U_3 = b(-{\Omega_1}^2 + {\Omega_3}^2) \\ U_4 = d({\Omega_1}^2 - {\Omega_2}^2 + {\Omega_3}^2 - {\Omega_4}^2) \\ \Omega_r = \Omega_2 + \Omega_4 - \Omega_1 - \Omega_3 \end{cases} \tag{3.3}$$

Ω_i : Speed of rotors i (i = 1 to 4)

b: Thrust Coefficient

d: Drag coefficient

3.2 Control of Quadcopter

The purpose of the control is to stabilize the Quadcopter and to position it to an accurate geographic coordinates. The Quadcopter is a nonlinear dynamic system. Its momentum equation is very complex and its stabilization is still a great challenge for all researchers in this field. It is a system whose inputs and outputs are not equal (four inputs : z ,ϕ, θ, Ψ and six outputs : x, y, z, ϕ, θ, Ψ), which makes control difficult [12].

There are several techniques used to control Quadcopters but in general, they can be classified into two categories:

- Linear Controllers: PD, PID, LQR (Linear Quadratic Regulator) , LQG, etc. Linear controllers are obtained by linearizing the nonlinear model to a simplified model. They can be adopted in the case of the flight near the hover.
- Nonlinear Controllers: adaptive control, control by sliding mode, Backstepping, nonlinear H∞, visual control, etc.

Linearizing the controllers distances the model from reality [14], for example, canceling the gyroscope effect, the inclinations of Quadcopter toward the front, the back, left and right are limited. Large angles cannot be covered. However, a nonlinear controller shall take into consideration the real cases, and equations are often complex.

The literature review about the Quadcopter testifies that the different techniques have advantages and disadvantages depending on the situation. The choice of technique depends on the objective of the Quadcopter project [15]. For example, for the case of linear controller, LQR is very efficient in terms of gain and error but the response time is very slow, whereas PID has a very fast response time, but the gains are not as robust as LQR [15].

To perform a test on a model, it is first preferable to work on a linear model. In this paper, we have been working on the PID regulator. The latter is widely used in the field of robotics thanks to the simplicity of its structure, its high performance on several processes according to several tests carried out by researchers and the opportunity to make adjustments (tuning) without the exact model of the system [1 2] .

The Quadcopter is supposed to fly almost flat, that is to say the roll and pitch angles are very small and the gyroscopic effects are negligible. According to the small angle theorem, we have:

ϕ and θ are small; $\dot{\phi} = \dot{\theta} = 0$

$$\cos\phi = \cos\theta = 1$$

$$\sin\phi = \phi \; ; \sin\theta = \theta$$

Equations (3.1) and (3.2) become:

$$\begin{cases} \ddot{x} = (\phi \, sin\Psi + \theta \, cos\Psi)\frac{U_1}{m} \\ \ddot{y} = (-\phi \, cos\Psi + \theta \, sin\Psi)\frac{U_1}{m} \\ \ddot{z} = \frac{U_1}{m} - g \end{cases} \qquad (3.4)$$

$$\begin{cases} \ddot{\phi} = l\frac{U_2}{I_{xx}} \\ \ddot{\theta} = l\frac{U_3}{I_{yy}} \\ \ddot{\Psi} = \frac{U_4}{I_{zz}} \end{cases} \qquad (3.5)$$

As mentioned earlier, the Quadcopter is controlled by four variables: angles of roll, pitch, yaw and z altitude. Its momentum equation is defined by:

$$\begin{cases} \ddot{z} = \frac{U_1}{m} \\ \ddot{\phi} = l\frac{U_2}{I_{xx}} \\ \ddot{\theta} = l\frac{U_3}{I_{yy}} \\ \ddot{\Psi} = \frac{U_4}{I_{zz}} \end{cases} \quad (3.6)$$

Assuming that in ascending flight $\frac{U_1}{m} \gg g$ and $\ddot{z} = \frac{U_1}{m}$

This equation can be written as follows:

$$\begin{cases} \ddot{z} = b_1 U_1 \\ \ddot{\varphi} = b_2 U_2 \\ \ddot{\theta} = b_3 U_3 \\ \ddot{\varphi} = b_4 U_4 \end{cases} \quad (3.7)$$

The following figure shows the internal loop block diagram:

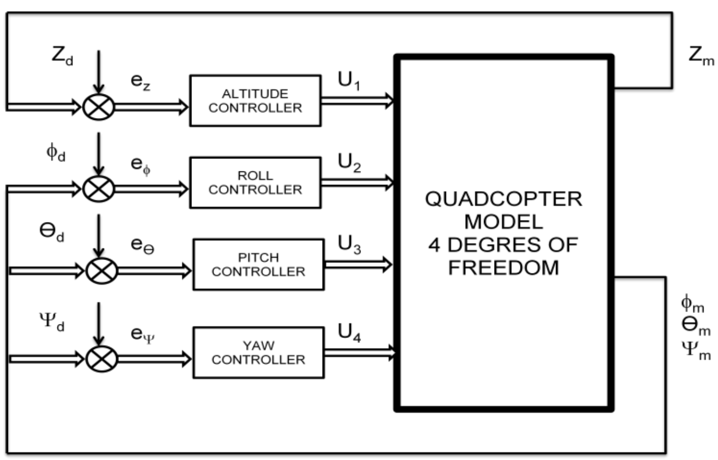


Figure 4: Quadcopter internal loop block diagram (4 degrees of freedom)

To control 6 degrees of freedom, according to equation (3.4), the angles of roll and of pitch are deducted from the x and y positions such as:

$$\begin{cases} \phi = (\sin\Psi\, \ddot{x} - \cos\Psi\, \ddot{y})\frac{m}{U_1} \\ \theta = (\cos\Psi\, \ddot{x} + \sin\Psi\, \ddot{y})\frac{m}{U_1} \end{cases} \quad (3.8)$$

According to [10], the Quadcopter makes a quasi stationary flight with respect to the z axis during its displacement. The lift force must cancel the gravitational force, which gives us: $\frac{U_1}{m} = g$

Equation (3. 8) becomes:

$$\begin{cases} \phi = (\, sin\Psi \, \ddot{x} - \, cos\Psi \, \ddot{y}) \frac{1}{g} \\ \theta = (cos\Psi \, \ddot{x} + \, sin\Psi \, \ddot{y}) \frac{1}{g} \end{cases} \quad (3.9)$$

Equation (3.9) allows us to determine the angles of roll and of yaw from the x and y positions. To control the entire system, position controllers must be implemented. The following figure shows the block diagram of the complete system.

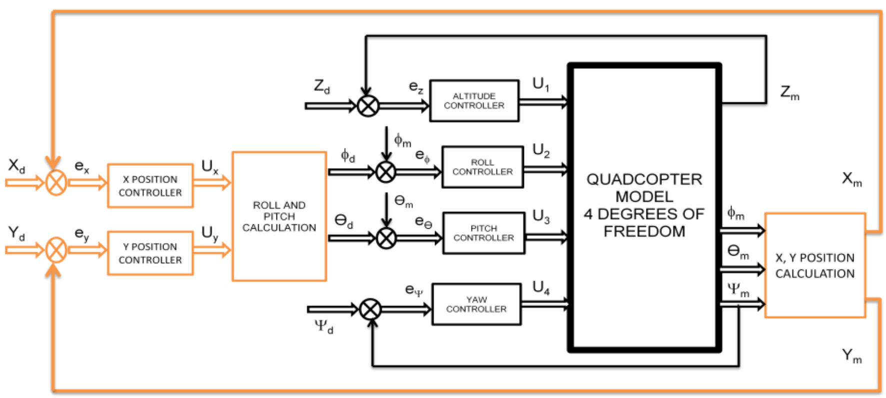


Figure 5: Quadcopter block diagram (six degrees of freedom)

As previously mentioned, we have focused on the PID controller as our development tool, which includes a graphical interface tool for adjusting the gains of the PID controller. The latter makes it possible to control a physical system in real time via a control loop. Its implementation requires the Laplace transform. Thus the' equation (3.7) becomes:

$$\begin{cases} \frac{Z(s)}{U_1\,(s)} = \frac{b1}{s^2} \\ \frac{\varphi(s)}{U_2\,(s)} = \frac{b2}{s^2} \\ \frac{\theta(s)}{U_3\,(s)} = \frac{b3}{s^2} \\ \frac{\Psi(s)}{U_4\,(s)} = \frac{b4}{s^2} \end{cases} \quad (3.10)$$

With:

s : Laplace Variable

U_i: Control variables (i varies from 1 to 4)

Table 1 shows the control equations

Table 1: Elevation, pitch, roll and yaw control equations

Equations	**Order**	**Controlled parameter**
$U_1 = K_{p1}(z_d - z) + K_{d1}(\dot{z}_d - \dot{z}) + K_{i1}\int(z_d - z)$	(3. 11)	Altitude
$U_2 = K_{p2}(\varphi_d - \varphi) + K_{d2}(\dot{\varphi}_d - \dot{\varphi}) + K_{i2}\int(\varphi_d - \varphi)$	(3. 12)	Roll
$U_3 = K_{p3}(\theta_d - \theta) + K_{d3}(\dot{\theta}_d - \dot{\theta}) + K_{i3}\int(\theta_d - \theta)$	(3. 13)	Pitch
$U_4 = K_{p4}(\Psi_d - \Psi) + K_{d4}(\dot{\Psi}_d - \dot{\Psi}) + K_{i4}\int(\Psi_d - \Psi$	(3. 14)	Yaw
$U_x = K_{px}(x_d - x) + K_{dx}(\dot{x}_d - \dot{x}) + K_{ix}\int(x_d - x)$	(3. 15)	Position x
$U_y = K_{py}(y_d - y) + K_{dy}(\dot{y}_d - \dot{y}) + K_{iy}\int(y_d - y)$	(3. 16)	Position y

With:

K_{pi} : Gain of P regulators

K_{di}: Gains of D regulators

K_{ii} : Gains of I regulators

K_{px}: Gains of P regulator (position along x)

K_{dx}: Gains of D regulator (position along x)

K_{ix}: I regulator (position according to x)

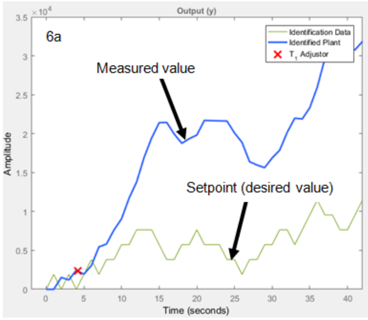
K_{py}: Gains of P regulator (position along o y)

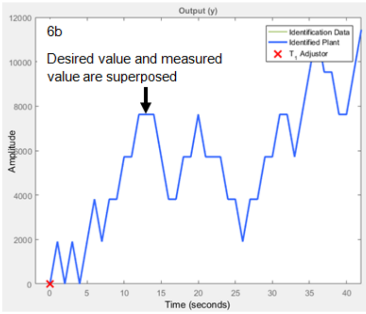
K_{dy}: Gain of D Regulator (position along y)

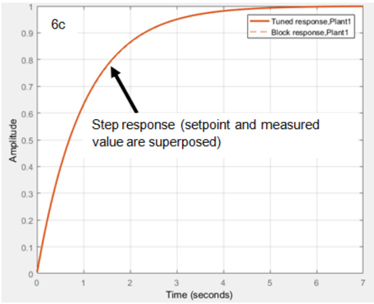
K_{iy}: Gain of I regulator (position along y)

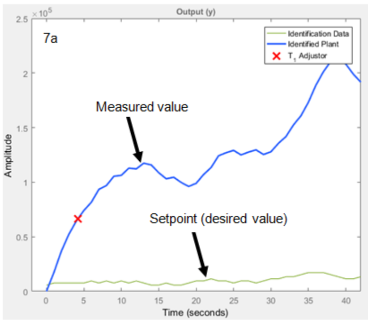
i varies from 1 to 4

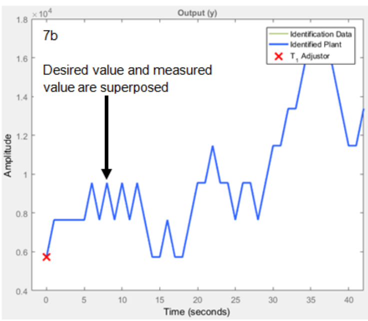
To illustrate these models, some simulation results are presented in Figures 6 and 7. These positions respectively x and y before and after the use of PID Tuner:


Output (y)
6a
Measured value
Setpoint (desired value)
Identification Data
Identified Plant
Amplitude
Time (seconds)


Output (y)
6b
Desired value and measured
value are superposed
Identification Data
Identified Plant
Amplitude
Time (seconds)


6c
Tuned response,Plant1
Block response,Plant1
Step response (setpoint and measured
value are superposed)
Amplitude
Time (seconds)


Output (y)
7a
Measured value
Setpoint (desired value)
Identification Data
Identified Plant
Amplitude
Time (seconds)


Output (y)
7b
Desired value and measured
value are superposed
Identification Data
Identified Plant
Amplitude
Time (seconds)

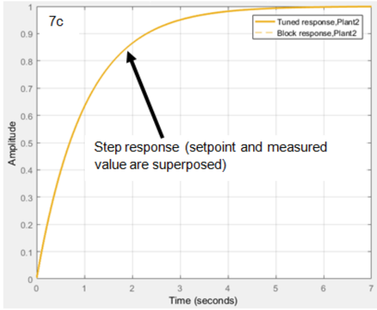

7c
Tuned response,Plant2
Block response,Plant2
Step response (setpoint and measured
value are superposed)
Amplitude
Time (seconds)

Figure 6a: x position before and after Tuning (unregulated system)

Figure 6b: x position before and after Tuning (regulated system)

Figure 6c: step response of the setpoint and measured value (x position)

Figure 7a: Position y before and after Tuning (unregulated system)

Figure 7b: Position y before and after Tuning (regulated system)

Figure 7c: step response of the setpoint and measured value (position y)

The Figures 6a and 7a show the identified signals before Tuning. The blue color indicates the instructions and in green the measured values. After the utilization of a PID tuner, on the Figure 6b, 7b, the data to the input controller (setpoints) and to the output of the system (measured values) are superimposed; similarly for the step responses of Figure 6c and 7c . This shows that the system is well regulated.

4. **Trajectory Generation**

The trajectory of the Drone is defined by the Dahalo's dynamic behavior model. It is a characteristic of displacement (movement) of Dahalo while transporting the rustled Zebu. According to this model, during the flight, the Dahalo always follow the strategic routes (forests, mountains or watering points).The geographical coordinates of all these different strategic routes constitute their trajectory.

The Table 2 shows some examples of strategic routes, places in which the Dahalo travel systematically when they transport the rustled zebu.

Table 2: List of strategic routes [2]

Name of the strategic routes	**Justification of choice**
Forest	The presence of other people is rare
Mountain	Makes search and attack difficult
Paths	Rustled zebus need to be fed during their travels Facilitates flight in case of retaliation
Watering point	Zebus need to drink while moving along the routes

Figure 8 shows an example of the dynamic behavior model of Dahalo when transporting rustled zebus to the destination [6].

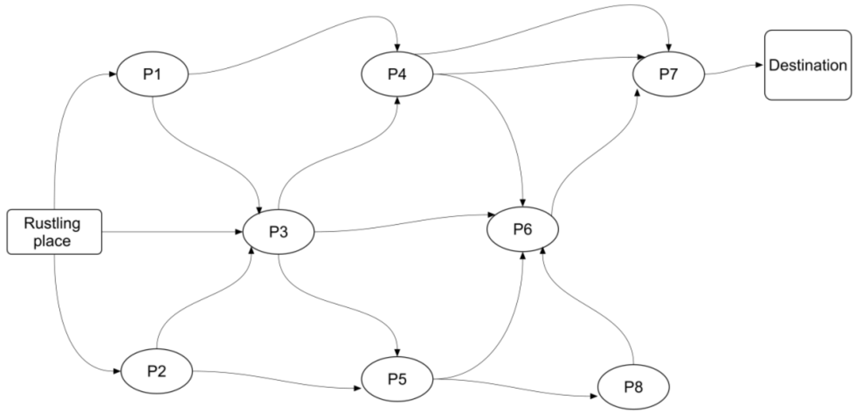


Figure 8: Dahalo Dynamic Behavior Model

P j (j = 1 to 8) are strategic routes.

The choice of the routes from the rustling places to another location is determined by several parameters. For example, "according to the police’s testimony and villagers," the forest is given priority over the others, but it also depends on the traveled distance, the remaining distance and flight information with respect to their itinerary. The watering points will be crossed only after having traveled some distance to quench the thirst of the zebus. This supply is necessary to give strength to the zebus as they move at an average speed of 8 to 10 km / h [2].

The exact route of Dahalo in relation to strategic routes is theoretically unpredictable. The fugitives define their trajectory according to the present context, but the pursuers (law enforcement and or villagers) do research on each neighboring point and thus continue gradually until the rustled zebus are located or beyond the coverage area. Once zebus are rustled, a coverage area will be defined by the police. They will only stop their search when they will exceed the limit area they have defined. But as for the Drone, it runs probabilistic paths determined in advance in [4], in this area of coverage. Table 3 shows an extract of the geographical coordinates of the strategic routes of the Kirindy Mitea National Park, Morondava , Toliary, Madagascar (geographical coordinates : -20.739726, 44.172005).

Table 3: Data Extracted from calculations and simulation of National Park Kirindy Mitea [3]

id	Longitude	Latitude	Elevation	HUE	TYPE	Group	Prob to move to G	Prob to move to B
0	44.099807739257997	-20.551151842359999	16	102.54545454545455	G	0	0.69999999999999996	0.29999999999999999
1	44.099807739257997	-20.567224667215999	16	102.54545454545455	G	0	0.53199999999999992	0.46799999999999997
2	44.099807739257997	-20.583295800443	10	102.54545454545455	G	0	0.51279999999999992	0.48719999999999991
3	44.099807739257997	-20.599365240956001	7	102.74615451464325	G	0	0.5051199999999999	0.49487999999999988
4	44.099807739257997	-20.615432987666999	8	103.24189526184539	G	0	0.50204799999999983	0.49795199999999984
5	44.099807739257997	-20.631499039493001	15	103.72471702930414	G	0	0.5008191999999998	0.49918079999999981
6	44.099807739257997	-20.647563395350002	13	104.14617006324666	G	0	0.50032767999999983	0.49967231999999978
7	44.099807739257997	-20.663626054152999	13	104.21052631578947	G	0	0.50013107199999984	0.49986892799999977
8	44.099807739257997	-20.679687014818999	13	104.21052631578947	G	0	0.50005242879999978	0.49994757119999977
9	44.099807739257997	-20.695746276266998	14	104.21052631578947	G	0	0.5000209715199998	0.49997902847999975
10	44.116973876952997	-20.486843648398001	1	126.80975838611307	B	1	0.50000838860799979	0.49999161139199977
11	44.116973876952997	-20.502923228893	8	159.0812799643216	B	1	0.50000335544319974	0.49999664455679976
12	44.116973876952997	-20.519001122111	6	131.72325581395356	B	1	0.50000134217727976	0.49999865782271974
13	44.116973876952997	-20.535077326962998	10	102.54545454545455	G	2	0.50000053687091173	0.49999946312908772
14	44.116973876952997	-20.567224667215999	9	103.38086936640853	G	2	0.50000021474836454	0.49999978525163491
15	44.116973876952997	-20.583295800443	13	103.92599915266203	G	2	0.50000008589934564	0.49999991410065375
16	44.116973876952997	-20.599365240956001	16	104.20795553434579	G	2	0.50000003435973805	0.49999996564026128
17	44.116973876952997	-20.615432987666999	14	104.21052631578947	G	2	0.50000001374389502	0.49999998625610431
18	44.116973876952997	-20.631499039493001	18	104.21052631578947	G	2	0.50000000549755774	0.49999999450244148
19	44.116973876952997	-20.647563395350002	19	104.21052631578947	G	2	0.50000000219902285	0.49999999780097631
20	44.116973876952997	-20.663626054152999	20	104.21052631578947	G	2	0.50000000087960883	0.49999999912039023
21	44.116973876952997	-20.679687014818999	20	104.21052631578947	G	2	0.50000000035184322	0.49999999964815578
22	44.116973876952997	-20.695746276266998	18	104.21052631578947	G	2	0.50000000014073698	0.49999999985926197
23	44.134140014647997	-20.486843648398001	2	129.90851403433646	B	3	0.50000000005629441	0.49999999994370442
24	44.134140014647997	-20.502923228893	6	102.67016622922137	G	4	0.50000000002251743	0.4999999999774814

Table 3 was obtained by modeling the studied geographical environment and the development of algorithms for identification and localization of strategic routes conducted in [3]. This program extracts the geographical coordinates of the studied area (longitude, latitude and altitude) and then classifies the environment of the strategic routes (Type and Group) depending on the hue value. Finally, it calculates the probability of getting into the cell of the same environments or of a different one. The first column of Table 3 shows the geographic cells of each element. The G Type, which means Green, indicates the presence of vegetation cover and B indicates the presence of water. They constitute the strategic routes. Figure 9 shows strategic routes in the Kirindy Mitea National Park, Morondava imported into Google maps (GPS coordinates: -20.739726, 44.172005)

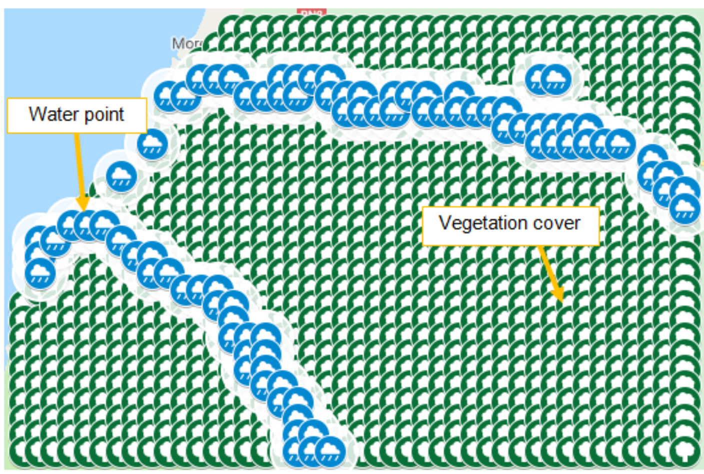


Figure 9: Geolocation of the vegetation cover and water points of Kirindy Mitea National Park, Morondava for a cell size of 1911m x 1911m

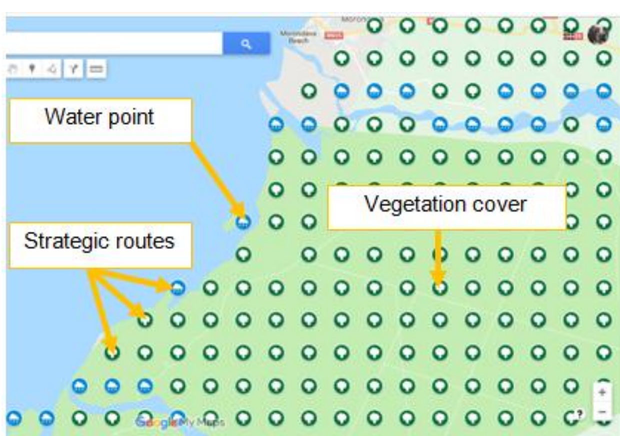


Figure 10: Zoom of the zone in Figure 9

After identifying the probable routes of Dahalo, an algorithm of trajectory estimation was applied to the data in Table 3 to minimize the search cost (time, energy, etc.) [4] [5] . These works allowed us to have a limited list of strategic routes to travel in order to find the rustlers of zebus during transport. Tables 4 and 5 show the examples of a limited routes list after the application of this algorithm. These different geographical points are then projected in the Mercator plan and after that implemented in the Quadcopter.

Table 4: GPS coordinates of the strategic routes after application of the trajectory's estimating algorithm (Simulation 1) [5]

Longitude	**Latitude**		**Longitude**	**Latitude**
44.1513062	-20.615433		44.2199707	-20.5511518
44.1684723	-20.615433		44.2199707	-20.5350773
44.1856384	-20.615433		44.2371368	-20.5350773
44.1856384	-20.5993652		44.254303	-20.5190011
44.1856384	-20.5832958		44.2714691	-20.5190011
44.2028046	-20.5832958		44.2886353	-20.5350773
44.2199707	-20.5672247		44.3058014	-20.5350773

Table 5: GPS coordinates of the strategic routes after application of the trajectory's estimating algorithm (Simulation 2) [5]

Longitude	**Latitude**		**Longitude**	**Latitude**
44.1513062	-20.6475634		44.3401337	-20.5029232
44.1684723	-20.631499		44.3401337	-20.4868436
44.1856384	-20.615433		44.3572998	-20.5029232
44.2028046	-20.5993652		44.3744659	-20.4868436
44.2199707	-20.5832958		44.3916321	-20.4707624
44.2199707	-20.5993652		44.3916321	-20.4868436
44.2371368	-20.5993652		44.4087982	-20.4707624
44.254303 0	-20.5832958		44.4259644	-20.4707624

44.2714691	-20.5672247	44.4431305	-20.4707624
44.2886353	-20.5511518	44.4431305	-20.4868436
44.3058014	-20.5511518	44.4602966	-20.5029232
44.2886353	-20.5350773	44.4774628	-20.5029232
44.3058014	-20.5190011	44.4946289	-20.5029232
44.2886353	-20.5029232	44.511795 0	-20.5029232
44.2886353	-20.4868436	44.5289612	-20.4868436
44.3058014	-20.5029232	44.5461273	-20.4868436
44.3229675	-20.5029232		

For the simulation, the geographic coordinates have been converted to Cartesian coordinates using the Mercator projection.

The trajectory will be also presented as horizontal, vertical or oblique segments.

5 Simulation, Results and Discussion

This section is intended to present the simulation results. Figure 11 shows the block diagram of the complete closed loop system.

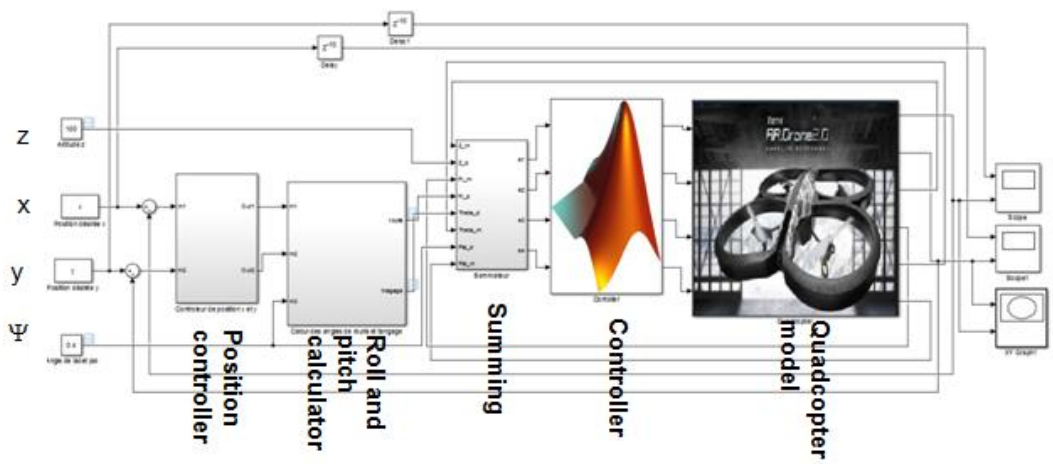


Figure 11: Simulink model of the complete system

The simulink model contains the input variables (x, y, z and Ψ□), thecontroller x and y position, the roll and pitch calculator block, the roll, pitch, yaw and altitude controller, and the dynamic model of the Quadcopter.

a. Simulation parameters: value of Quadcopter parameters,

Table 6 shows the parameters of Quadcopter used during simulation.

Table 6: Ar Drone 2.0 parameter

Parameter	Value	Unit
m	0, 429	kg
l	0, 1785	m
g	9, 8	m/s ^{2}
Ixx	2, 2383	mg.m ^{2}
Iyy	2, 9858	mg.m ^{2}
Izz	4, 8334	mg.m ^{2}
Jr	2 2 , 0321	µgm ^{2}

Tables 7 and 8 show the controllers' parameters.

Table 7: PID gain after using PID tuning tools of altitude and attitude

Controller	Kp	Ki	Kd
Altitude	0.55003715290698	0.0119958491316202	1.41377438928193
Roll	0.329765317102763	0.00192487025899562	0.125451519016567
Pitch	0.04393435971894	0.002564488683233	0.167138018400025
Yaw	0.416729543739224	0.150561679263313	0.265861154712101

Table 8: PID gain after the use of the tool " PID tuning " of command x and y position

Controller	Kp	Ki
Position x	0.0049779639839531	0.99559279679062
Position y	0.00498817220033574	0.997634440067148

b. Simulation

The purpose of these simulations is to check if the Quadcopter can follow the given trajectories. Figures 12, 13, 15 and 16 are obtained from the estimating algorithm of the likely Dahalo's routes when they transport the rustled zebus [3] [4]. The blue lines represent the route to the location where the search should be started. The green lines indicate the trajectory to the original location and the red lines are the likely routes after application of these two algorithms [3] [4]. The points colored red represents the located area where the rustling takes place, and green colored points indicate the laundering area of the rustled zebu.

To Figure 12, the position before the tracking is estimated at coordinates (5733 m, 9555 m) . For Figure 15, the position before tracking is estimated at the point of coordinates (5733 m, 5733 m). In Figure 12, because of the energy limit, the Drone returns to the base station.

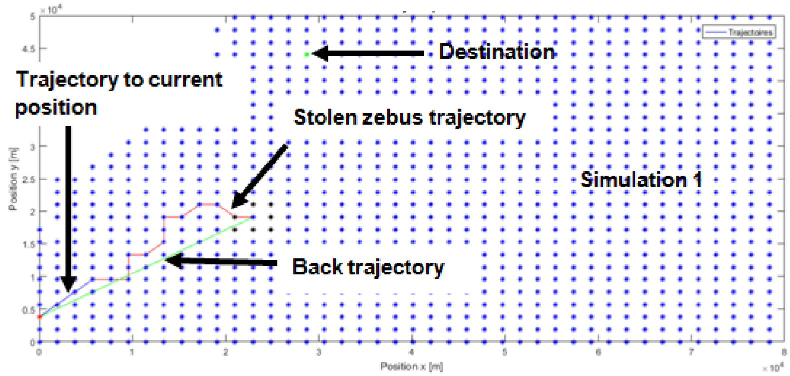


Figure 12: Dahalo tracking trajectory (Data from Table 4)

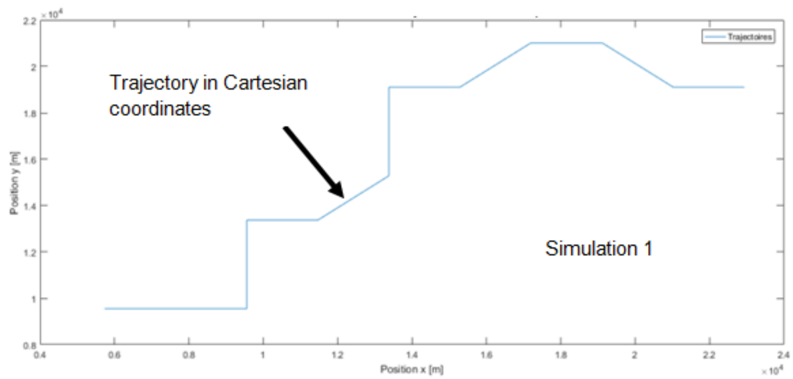


Figure 13a: Dahalo tracking trajectory in artesian coordinates (Data from Table 4)

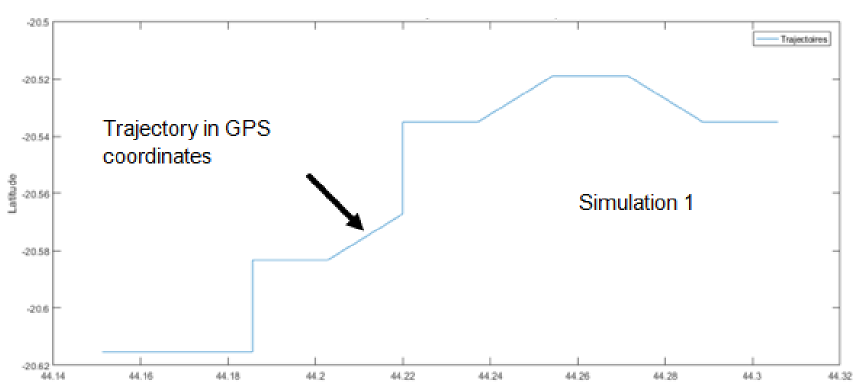


Figure 13b: Dahalo tracking Trajectory in GPS coordinates (Data Table 4)

After the application of the algorithms in [4] [5], the likely routes are saved in a file " .xls " in GPS coordinates and Cartesians, and then imported in the development tool (Matlab) as setpoints of the Quadcopter .

The figure 14 shows the result of the first simulation. The blue line presents the targeted trajectory (setpoint) and a red line is for the measured trajectory.

Figures 17 and 18 show the results of the second and third simulations. Similarly , the blue lines show the targeted trajectories (instructions) and the measured trajectories in red lines.

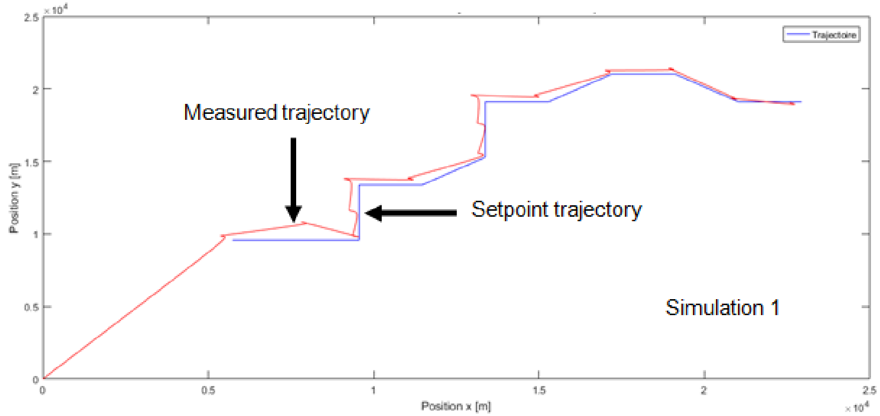


Figure 14: Trajectory tracking of Dahalo after boarding the GPS coordinates in the Drone's digital model (Table Data 4)

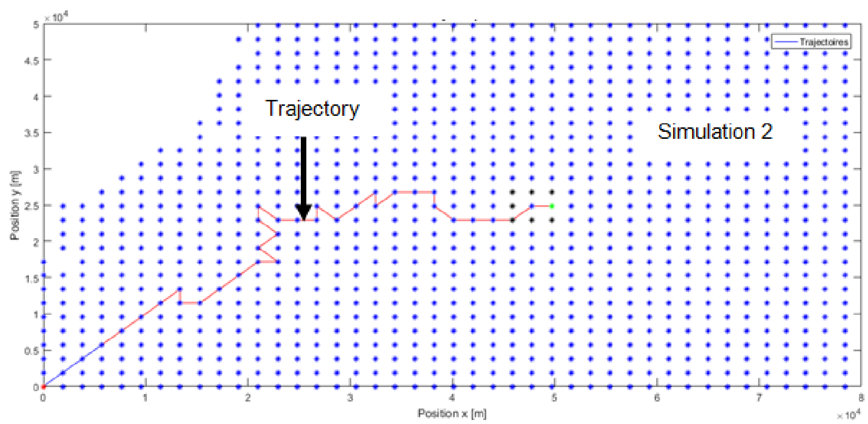


Figure 15: Dahalo tracking trajectory (Data from Table 5) (2nd simulation)

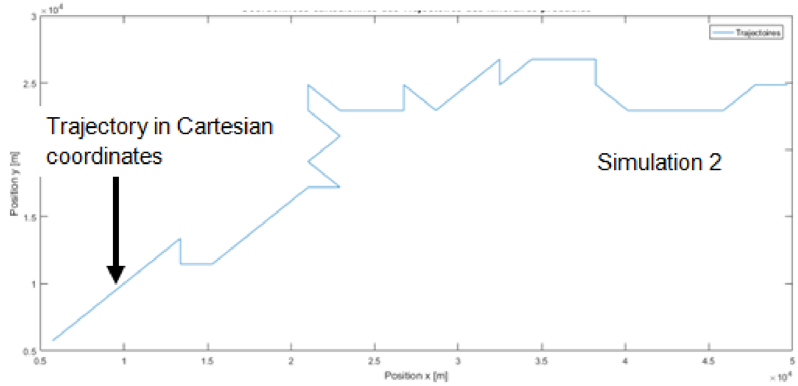


Figure 16a: Dahalo tracking trajectory in Cartesian coordinates (Data from Table 5)

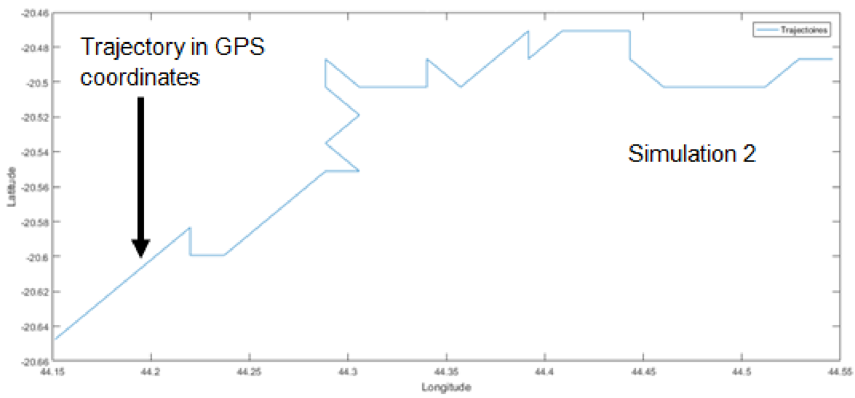


Figure 16b: Dahalo tracking trajectory in GPS coordinates (Data in Table 5)

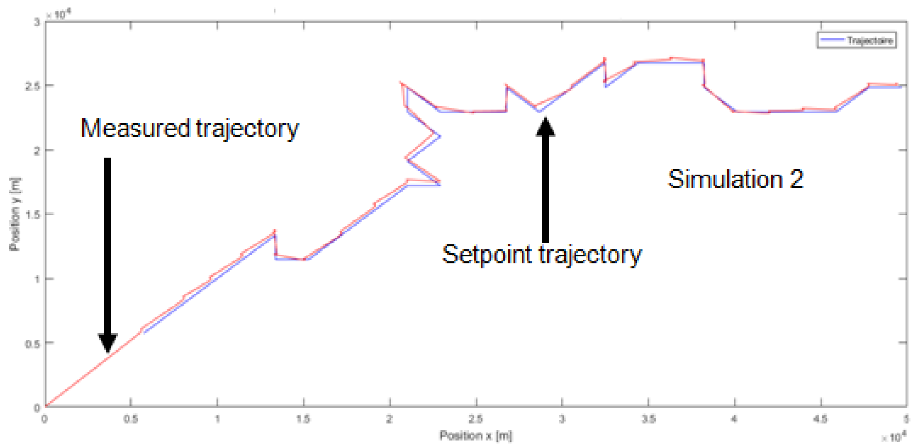


Figure 17: Dahalo tracking trajectory after boarding the GPS coordinates in the Drone's digital model (Table Data 5)

In figures 14 and 17, the contours before each change of direction are due to the fact that the Drone cannot make a sudden change in direction. Several simulations relating to a large number of possible cases have been conducted in order to determine the robustness of our approach. Thus, through the two examples taken as illustration here, we have noticed that the provided trajectories and the trajectories of the Quadcopter coincide with a negligible error.

6. Conclusion

This work designs a Quadcopter's controller to track down Dahalo or cattle rustlers in complex environments. The trajectories of the Dahalo are of oblique, horizontal or vertical lines. These are stated and drawn from previous studies. These probable routes are then used for the Quadcopter flight plan to track down the criminals.

In this approach, we used the PID controller to control the Quadcopter, thanks to its performance , ease of implementation and its presence in our development tool. The tool we have used is the PID tuner in Matlab . This tool adjusts the values of the controller parameters using its graphical interface. The results show that the tuner PID can be relied on to regulate the PID parameters because the offsets between measurements and setpoints are relatively small. Also, despite the complexity of the models while taking into account the dynamic behavior of the Dahalo, the response time of the device is quite competitive.

The model of the Quadcopter used is linear. This could cause some shifts in use of the real system such as the impacts on the accuracy of geographic points to visit. The next step in this study is to test this controller with a nonlinear model.

References

B. Samir (2007). Design and control of quadrotors with application to autonomous flying. [14]

B. Tsimitamby, E. J. (2016). Rapport intermédiaire d'évaluation de thèse: Comportement dynamique des voleurs de zébus dans la partie sud de Madagascar. Toliara. [2]

B. Tsimitamby, E. J. (2018). Conférence des 3 IST: Localisation automatique d'un groupe d'individus aux comportements dynamiques aléatoires. Antananarivo. [6]

B. Tsimitamby, E. J. (2018). Identification de points stratégiques de parcours des Dahalo par une analyse cellulaire. [3]

B. Tsimitamby, E. J. (2019). Estimation des trajectoires des Dahalo par une analyse cellulaire. Conférence des 3 ISts 2019. [4]

B. Tsimitamby, E. J. (2019). Optimisation de trajectoire de traque de Dahalo, Université d'été Majunga 2019. [5]

C. Young-Cheol, A. H.-S. (2014). Nonlinear Control of Quadrotor for Point Tracking: Actual Implementation and Experimental Tests. IEEE/ASME Transactions on mechatronics. [18]

Charly. (2017, aôut 14). Le Phénomène Dahalo. Consulté le 27avril, 2019, sur Journal d'évasion: http://www.journaldevasion.com/les-zebus-23-le-phenomene-dahalo/ [1]

D. Kotarski, Z. B. (2016). Control design for unmanned aerial vehicles with four rotors. [10]

G. Jithu, P. R. (2016). Quadrotor modelling and control. International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT). [13]

Koszewnik, A. (2014). The Parrot UAV controlled by PID controllers. acta mechanica et automatica . [9]

L. M. Argentim, W. C. (2012). PID, LQR and LQR-PID on a Quadcopter Platform. [15]

N. Guenard, T. H. (s.d.). Modélisation et élaboration de commande de stabilisation de vitesse et de correction d’assiette pour un drone de type X4-Flyer. [17]

N. H. Abbas, A. R. (2017). Tuning of PID Controllers for Quadcopter System using Hybrid Memory based Gravitational Search Algorithm – Particle Swarm Optimization. International Journal of Computer Applications. [8]

N. I. R. Pacheco, D. d. (2015). Stability Control of an Autonomous Quadcopter through PID Control Law. N. I. R. Pacheco et al. Int. Journal of Engineering Research and Application. [12]

S. Kurak, M. H. (2018). Control and Estimation of a Quadcopter Dynamical model. Periodicals and Engineering and Natural Sciences. [7]

T. Hamel, P. S. (s.d.). Modélisation, estimation et contrôle des drones à voilures tournantes : Un aperçu des projets de recherche français. [16]

V. Praveen, A. S. (2016). Modeling and simulation of Quadcopter using PID controller. International Science Press. [11]