# Flight control and position estimation for a tethered micro aerial robot

Ricardo Miguel Parreiral Pinheiro de Matos Martins
Instituto Superior Técnico, Lisboa

December 2019

**Abstract**

This work focuses on the study of tethered flights of a small quadcopter from a ground-base, which could be used to supply continuous power to the aerial robot and significantly extend its flight time. Some features that facilitate the interaction between the quadcopter and the ground-base are studied and implemented. The research starts by introducing a method capable of estimating the tension applied to a quadcopter by using the inertial information from the IMU sensors and the quadcopter's thrust. To filter the undesired noise presented in the IMU measurements a Kalman filter is implemented. After the filtering process, the tension estimate is used to present two alternative methods to control the UAV, based on the tension applied to it. The first method allows to change the quadcopter's position by applying a force to it. This solution is extended to implement a process of landing a UAV by pulling the tether tethered to it. Moreover, the second method allows to develop a novel position control structure, based on the tension that the tether applies to the quadcopter, and on the shape of a tether that outlines a catenary curve. At last, the position of the quadcopter is estimated taking into account the quadcopter's tension estimates, and the shape of a tether that fits into the catenary model.

**Keywords:** Tethered flight – Quadcopter modeling – Catenary curve – Kalman Filter

# 1 Introduction

## *1.1 Motivation*

Micro aerial robots are known by their versatility and ability to get information from higher altitudes, whereas ground-robots are not able to do so. However, their small size does not allow them to carry much load, which also means that their batteries must have a small size, shorting their maximum flight time. Given that ground-robots have a big payload capacity, higher computational power and long battery life time, the small quadcopter can be attached to a ground-robot, increasing the quadcopter's benefits. For simplification purposes, throughout this work, the implemented solutions consider a non-movable ground-base, but they can be transposed to a scenario with a mobile ground-base. Nevertheless, the quadcopter can also provide to ground-robots the awareness and versatility that they usually do not have. This way, the UAVs would have longer flight times and the ground-robots would have access to information that could not capture before.

## *1.2 Problem definition*

Despite the benefits mentioned above, there are several challenges that must be considered when conducting tethered flights, such as the tension that the tether applies to the UAV. In case of a non-rigid tether connecting the quadcopter to a ground-base, the shape of the tether approximately outlines a catenary curve.

The flight control of the UAV can be performed based on the tension applied to it. Thus, it is necessary to have an accurate estimate of that tension. However, the IMU measurements have a high noise level that does not allow to estimate the tension with the desired accuracy.

The position estimate can be improve by exploiting the properties of the tether and using the tension applied to the UAV. The tension on the end-points of a catenary curve is related to its shape, which means that the relation between the catenary model and the tension applied to the quadcopter can be expressed mathematically.

## *1.3 Contributions*

This paper focuses on the study of tethered flights, implementing features that can improve the interactions between a quadcopter and a ground-robot. Aiming for that, the following procedures are developed: **a computational simulation** that allows to pre-test the developed features in a safe environment, taking into account both tethered and untethered flights; two alternative ways of **controlling the position of the quadcopter based on the tension** applied to it, allowing to land the UAV by pulling the tether and removing the necessity to have an external motion system; a procedure **to estimate the position of the quadcopter**, while assuming that the tether outlines a catenary curve.

### 1.4 Literature Review

The use of the catenary to model the tether attached to a single UAV is studied in [1]. The use of multiple UAVs tethered between them is approached in [2] and [3], which can be useful in cooperative missions, such as the transport of heavy payloads. In several situations, the UAVs can be used to provide additional information to a ground-robot, such as described in [4] and [5]. A possible design for the electrical and mechanical components of the ground-base platform, which supplies the power to the UAV, is introduced in [6]. Furthermore, the resistance and weight of the cable cannot be neglected, and so, different cables have different performances. In [7] a method to make a cable selection according to the relation between "power loss in the cable and payload is discussed" in order to increase the system performance. Additionally, a method that enables to switch between a tethered powered supply and a battery supply, while the quadcopter is flying, is presented in [8].

## 2 Quadcopter Modelling

Start by defining the world frame $W$, where $\{w_1, w_2, w_3\} \in W$, and the body frame $B$, where $\{b_1, b_2, b_3\} \in B$. The relation between them is given by the position vector $\vec{Rr}$ - with its horizontal projection being defined as the radial distance vector $\vec{r}$ - and by the rotation matrix $R(\eta)$. Figure 1 illustrates both frames ($\{W\}$ and $\{B\}$) as well as the vector position $Rr$, and the Euler angles roll, pitch and yaw ($\theta$, $\phi$, and $\psi$, respectively) [2], [9] [10].

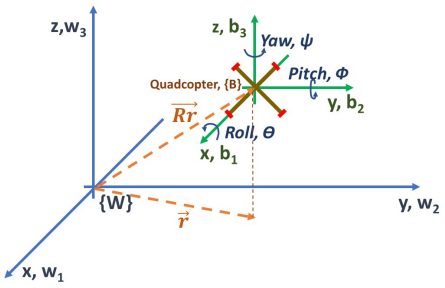



Figure 1: World frame $\{W\}$, rigid-body frame $\{B\}$, radial vector $\vec{Rr}$, and its horizontal projection $\vec{r}$.

The quadcopter is modelled through the Newton-Euler equations, according to equations 1 and 2. Consider $\vec{a}$ as the acceleration of the quadcopter in the world frame $\{W\}$, $F_p$ as the force of the propellers, $m$ as the mass of the quadcopter, $\vec{g}$ as the gravity vector, $\vec{\omega}$ as the angular velocity vector of the quadcopter regarding the body frame, $I$ as the inertia matrix of the quadcopter, $\eta$ the attitude ($\theta$, $\phi$ and $\psi$), and $\tau$ as the torque made by the propellers. Therefore, the system is obtained from the Newton-Euler equation as follows:

$$m\vec{a} = R(\eta)F_p + m\vec{g} + F_{ext} \tag{1}$$

$$\dot{\vec{\omega}}I = \tau - \vec{\omega} \times (\vec{\omega}I) + \tau_{ext} \tag{2}$$

Equation 3 displays the relation between the quadcopter torques and the thrust produced by each motor, assuming that the axes of the body frame $B$ are rotated 45° degrees regarding the axes of the motor's arms.

$$\begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = \begin{bmatrix} \frac{\sqrt{2}}{2}l(-f1 - f2 + f3 + f4) \\ \frac{\sqrt{2}}{2}l(-f1 + f2 + f3 - f4) \\ C(-f1 + f2 - f3 + f4) \end{bmatrix} \tag{3}$$

The thrust of the quadcopter corresponds to the sum of the thrust produced by each motor.

$$F = f1 + f2 + f3 + f4 \tag{4}$$

Equation 5 maps the torques and total thrust of the quadcopter to the thrust of each motor.

$$\begin{bmatrix} f1 \\ f2 \\ f3 \\ f4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -\frac{\sqrt{2}}{2}l & -\frac{\sqrt{2}}{2}l & \frac{\sqrt{2}}{2}l & \frac{\sqrt{2}}{2}l \\ -\frac{\sqrt{2}}{2}l & \frac{\sqrt{2}}{2}l & \frac{\sqrt{2}}{2}l & -\frac{\sqrt{2}}{2}l \\ -C & C & -C & C \end{bmatrix}^{-1} \begin{bmatrix} F \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} \tag{5}$$

The input of the motors is a PWM signal. Therefore, equation 6 presents the map between the thrust and the input command, according to [11].

$$f = 2.130\times10^{-11}cmd^2 + 1.033\times10^{-6}cmd + 5.485\times10^{-4} \tag{6}$$

## 3 Characterization of the shape of the tether

### 3.1 Catenary Curve

The catenary model consists of a hanging cable, with no stiffness, sagging under its own weight and supported only by its ends (see figure 2).

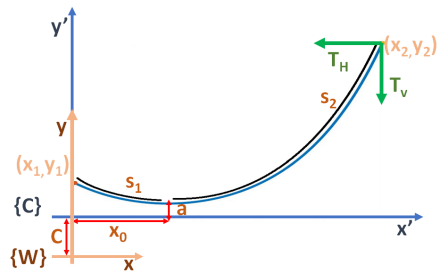



Figure 2: Catenary curve and related parameters; world $\{W\}$ and catenary $\{C\}$ frames.

The point $(x_1, y_1)$ corresponds to the origin and the point $(x_2, y_2)$ to the position of the quadcopter. The shape of the catenary can be defined according to a mathematical model, in which expression 7 presents the **equation of the catenary**.

$$y = a.cosh\left(\frac{x - x_0}{a}\right) + C \tag{7}$$

Parameter $\langle x_0 \rangle$ is the abscissa of the lowest point. Parameter $\langle a \rangle$ corresponds to the $y$ coordinate of the lowest point of the curve ($x = x_0$) regarding the catenary frame $\{C\}$, and it must always be positive.[1]. Parameter $\langle C \rangle$ is an offset between the world frame $\{W\}$ and the catenary frame $\{C\}$, which depends on the tether's parameter $\langle a \rangle$ and the $y$ coordinate of the lowest point regarding the world frame ($y_0$).

$$C = y_0 - a \tag{8}$$

Expressions 9 and 10 introduce the tether parameters $\langle s_1 \rangle$ and $\langle s_2 \rangle$, which represent the arc-length from the curve lowest point to the origin and to the UAV, respectively.

$$s_1 = a.sinh\Big(\frac{|x_1 - x_0|}{a}\Big) \tag{9}$$

$$s_2 = a.sinh\Big(\frac{|x_2 - x_0|}{a}\Big) \tag{10}$$

Equations 11 and 12 respectively present the horizontal and vertical tension on the end-points of the catenary curve. Both the horizontal and vertical tension depend on the tether's parameters - $\langle a \rangle$ or $\langle s \rangle$ - and on the weight of the tether, where $\omega$ is the tether weight per length unit [12].

$$T_H = \omega.a \tag{11}$$

$$T_V = \omega.s \tag{12}$$

The absolute value of the tension results from the euclidean norm of the horizontal and vertical tensions.

$$|T| = \sqrt{T_V^2 + T_H^2} \tag{13}$$

The quadcopter flying in a $R^3$ space allows to define the horizontal tension ($T_H$) in terms of a component along the $x$ direction and another along the $y$ direction, as shown in figure 3.

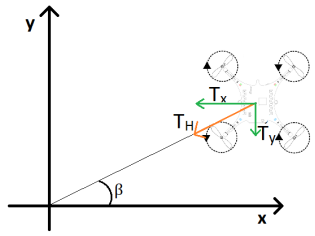



Figure 3: Horizontal tension decomposition.

The previous circumstance relates the tension along the $x$ and $y$ direction according equation 14 and 15, respectively.

$$Tx = cos(\beta).|T_H| \tag{14}$$

$$Ty = sin(\beta).|T_H| \tag{15}$$

[1] A negative value of parameter $\langle a \rangle$ would only have a physical meaning if the shape of the catenary was concave instead of convex.

The parameters of the catenary cannot be mathematically computed by knowing only the two end-points of the cable. Thus, these parameters - $\langle a \rangle$, $\langle x_0 \rangle$, $\langle C \rangle$, $\langle s_1 \rangle$, and $\langle s_2 \rangle$ - can be obtained gathering additional information in two different ways: by matching the lowest point of the catenary $\langle x_0 \rangle$ to the origin (**case 1**); or by knowing the total length of the tether $\langle s_{tot} \rangle$ (**case 2**). Furthermore, these two different approaches have *pros* and *cons*. In **case 1**, the length of the tether is not fixed but a ground-controller is necessary to match the tether lowest point to the origin; in **case 2**, the ground-controller is no longer necessary but the length of the tether is assumed to be fixed.

### *3.2 Knowing two points and matching the lowest point of the tether to the origin*

Start by considering that equations 16 and 17 describe two points, with the first point $(x_1, y_1)$ matching the origin (see equation 18).

$$y_1 = a.cosh(\frac{x_1 - x_0}{a}) + C \tag{16}$$

$$y_2 = a.cosh(\frac{x_2 - x_0}{a}) + C \tag{17}$$

$$x_0 = x_1 \tag{18}$$

Figure 4 illustrates the case.

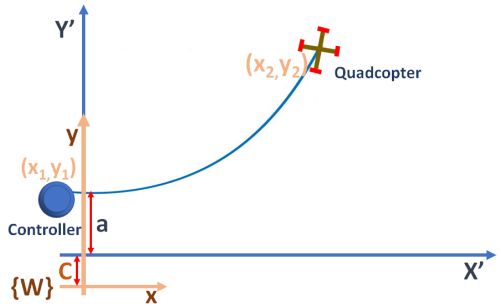



Figure 4: Quadcopter attached to the ground controller maintaining $x_0$ at the origin.

When replacing $x_0$ in equation 16 by equation 18 it follows expression 19, and subtracting equation 17 by equation 16 results the expression in equation 20, with $\Delta y = y_2 - y_1$.

$$C = y_1 - a. \tag{19}$$

$$f(a) = \Delta y - \Big(a.cosh\Big(\frac{x_2}{a}\Big) - a\Big) = 0 \tag{20}$$

Parameter $\langle a \rangle$ is computed by using the iterative method of *Newton-Raphson* in equation 20. This method does not guarantee the convergence of the parameter to be estimated. To improve the convergence of the method, it is necessary to have an accurate initial estimation of the parameter $\langle a \rangle$, which not only helps the method to converge but also reduces the necessary steps to do it, turning it faster. The expansion of the hyperbolic cosine using a Taylor series (see equation 21), and using the Cardano's formula on the

resultant cubic equation, allows to estimate an initial approximation for the parameter $\langle a \rangle$.

$$cosh(x) = \frac{e^x + e^{-x}}{2} = \sum_{n=0}^{\inf} \frac{x^{2n}}{(2n)!} = 1 + \frac{x^2}{2!} + \frac{x^4}{4!} + \frac{x^6}{6!} + ... \quad (21)$$

The Cardano's formula compute the roots of a cubic polynomial having the general expression of equation 22 with a, b and c $\in \Re$.

$$x^3 + \alpha x^2 + \beta x + \gamma = 0 \quad (22)$$

Any cubic expression having the form of expression 22 can be re-written in a reduced form, as shown in equation 23,

$$y^3 + p.y + q = 0 \quad (23)$$

where $y = x + \frac{\alpha}{3}, p = \beta - \frac{\alpha^2}{2}$ and $q = \gamma + \frac{2\alpha^3}{27} - \frac{\alpha.\beta}{3}$. After re-writing expression 22 into expression 23, the Cardano method allows to compute the roots of expression 23 (see equation 24). Notice that these roots correspond to the roots of expression 23, so it is necessary to apply the change of variable $x = y - \frac{\alpha}{3}$ in order to obtain the roots of expression 22 instead.

$$roots = \sqrt[3]{\frac{-q}{2} + \sqrt{\frac{q^2}{4} + \frac{p^3}{27}}} + \sqrt[3]{\frac{-q}{2} - \sqrt{\frac{q^2}{4} + \frac{p^3}{27}}} \quad (24)$$

Given that, equation 25 results from the fourth-order approximation of equation 20 according to the Taylor expansion in equation 21.

$$\Delta y - \frac{x_2^2}{2!a} - \frac{x_2^4}{4!a^3} = 0 \quad (25)$$

Using equation 25, it results: $\alpha = \frac{x_2^2}{2!\Delta y}, \beta = 0$ and $\gamma = \frac{x_2^4}{4!\Delta y}$. Using the Cardano's formula on equation 22, replacing $\alpha$, $\beta$ and $\gamma$ by these new values, the initial estimation for the $\langle a \rangle$ parameter is computed. Equation 19 computes parameter $\langle C \rangle$, by knowing parameter $\langle a \rangle$.

### *3.3 Knowing two points and the tether length*

Another way of solving the problem is assuming that $x_0$ is unknown, but the length of the tether is constant and known. This assumption leads to a possible and determined system of equations, similar to the one given by equations 16-18. Given that, solving the system of equations 26-30 generates the curve parameters.

$$y_1 = a.cosh(\frac{x_1 - x_0}{a}) + C \quad (26)$$

$$y_2 = a.cosh(\frac{x_2 - x_0}{a}) + C \quad (27)$$

$$s_{total} = s_2 + s_1 \quad (28)$$

$$s_1 = a.sinh\Big(\frac{|x_1 - x_0|}{a}\Big) \quad (29)$$

$$s_2 = a.sinh\Big(\frac{|x_2 - x_0|}{a}\Big) \quad (30)$$

Subtracting equation 26 from equation 27, and making use of the hyperbolic cosine properties, it follows:

$$\Delta Y = 2a.sinh(\frac{\Delta x}{a}).sinh(\frac{x_{average} - x_0}{a}), \quad (31)$$

where $\Delta x = \frac{x2 - x1}{2}$, $x_{average} = \frac{x2 + x1}{2}$ and $\Delta y = y_2 - y_1$. The expression of the length of the tether $\langle s_{total} \rangle$ is re-written by replacing equation 29 and 30 into expression 28, and using once more the hyperbolic sine properties.

$$s_{total} = 2a.sinh(\frac{\Delta x}{a}).cosh(\frac{x_{average} - x_0}{a}), \quad (32)$$

Equation 33 presents a useful relation between $\langle x_0 \rangle$ and $\langle a \rangle$, which results from the division of $\Delta Y$ by $\langle s_{total} \rangle$.

$$x_0 = x_{average} - a.tanh^{-1}(\frac{\Delta Y}{s_{total}}) \quad (33)$$

The insertion of equation 33 in equation 31 allows to obtain equation 34. Afterwards, using the Newton-Raphson method on the equation 34 produces the value of parameter $\langle a \rangle$, similar to what is done in sub-section 3.2. However when $\Delta Y = 0$ the parameter $\langle a \rangle$ is impossible to compute since equation 34 does not depend on $\langle a \rangle$ to be valid. Nevertheless, from the knowledge that $\Delta Y = 0$ it comes that $x_0$ is known and corresponds to $x_{average}$, which means that equation 32 can compute the parameter $\langle a \rangle$. An alternative approach is to force the value of $\Delta Y$ to be not null by adding a small offset.

$$\Delta Y - 2.a.sinh(\frac{\Delta x}{a}).sinh(tanh^{-1}(\frac{\Delta Y}{s_{total}})) = 0 \quad (34)$$

The expansion of the Taylor series derives the initial estimation of parameter $\langle a \rangle$.

$$sinh(x) = \frac{e^x - e^- x}{2} = \sum_{n=0}^{\inf} \frac{x^{2n+1}}{(2n+1)!} = x + \frac{x^3}{3!} + \frac{x^5}{5!} + ... \quad (35)$$

By applying this approximation to expression 34, one can re-write this last equation as a result of a $5^{th}$ order approximation for the hyperbolic sine, according to equation 36.

$$\Big(\frac{\Delta Y}{2.sinh\big(tanh^-1(\frac{\Delta Y}{s_{tot}})\big)} - \Delta x\Big)a^4 - \frac{\Delta x^3}{3!}a^2 - \frac{\Delta x^5}{5!} = 0 \quad (36)$$

The assumption that $\alpha = a^2$ reduces equation 36 to the $2^{nd}$ order. Furthermore, the quadratic formula presented in equation 37 produces the solution for this $2^{nd}$ order expression

$$x^2 = \frac{-b + / - \sqrt{b^2 - 4.a.c}}{2.a}, \quad (37)$$

where

$$a = \frac{\Delta Y}{2.sinh(tanh^{-1}(\frac{\Delta Y}{s_{total}}))} - \Delta x, b = \frac{-\Delta X^3}{3!}, c = \frac{-\Delta x}{5!}. \quad (38)$$

Reverting the variable substitution produces the desired value for $\langle a \rangle$, according to equation 36.

$$a = \sqrt{(\alpha)}, \tag{39}$$

Since equation 36 is a $4^{th}$ order equation it produces 4 roots - in the relevant domain, two of them are complex roots and two of them real roots, one positive and other negative. Only the positive real root has a physical meaning and so it is the only one to be taken into account.
The replacement of $\langle x_0 \rangle$ and $\langle a \rangle$ into equation 26 or into equation 27 allows to compute the $\langle C \rangle$ parameter.

# 4 Tension Estimation - Kalman filtering

## *4.1 General information*

The sensor readings concerning the attitude, accelerations, and thrust of the quadcopter present a high level of noise, which means that it is impossible to determine the tension of the quadcopter with the desired accuracy. To filter the undesired noise, and assuming that the noise is white Gaussian, a Kalman filter is implemented.
Equations 40 and 41 describe a linear system in which $w_k$ and $v_k$ are the process and the observation noise, respectively.

$$x_k = Ax_{k-1} + Bu_k + w_k \tag{40}$$

$$y_k = Cx_k + D + v_k \tag{41}$$

Additionally, denote Q and R as the process and observation noise covariance matrices.

$$Q = \begin{bmatrix} w_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & w_n \end{bmatrix}, R = \begin{bmatrix} v_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & v_n \end{bmatrix} \tag{42}$$

The Kalman filter presents an optimal solution for linear systems affected by white-Gaussian noise.
The Kalman filter has two main steps, which are the prediction and correction. The prediction step estimates the system state having only into account the model (*a priori* state estimate); the correction step takes the information from the state observation and re-computes the state estimate (*a posteriori estimate*). The weight between the *a priori* state estimate and the observations is managed through the Kalman gain.
**Prediction:**

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_k \tag{43}$$

$$P_k^- = AP_kA^T + Q \tag{44}$$

**Correction:**

$$K_k = P_k^- C^T (CP_k^- C^T + R)^{-1} \tag{45}$$

$$\hat{x}_k = \hat{x}_k^- + K_k(y_k - C\hat{x}_k^-) \tag{46}$$

$$P_k = (I - K_kC)P_k^- \tag{47}$$

## *4.2 Filtering process using a constant model*

The implemented solution considers a model where the tension remains the same, which can be extended to situations where the wire is not abruptly pulled and do not have considerable oscillations.

$$\hat{x}_{k+1} = \hat{x}_k \tag{48}$$

The state estimate ($\hat{x_k}$) is a $\Re^3$ vector, including the tension along $x$, $y$ and $z$ directions, and $y_k$ is the observation vector, which includes the tension measurements. The observation vector is computed through indirect measurements, using the inertial information from the quadcopter. The state and the observation vector as well as $A$, $B$, $C$ and $D$ matrices are represented in 49.

$$\begin{gathered} \hat{x}_k = \begin{bmatrix} Tx_k \\ Ty_k \\ Tz_k \end{bmatrix}, y_k = \begin{bmatrix} Tx_k^{obs} \\ Ty_k^{obs} \\ Tz_k^{obs} \end{bmatrix}, \\ A = C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, B = D = 0 \end{gathered} \tag{49}$$

The tension measurements $T^{obs}$ are obtained through equation 50, replacing its parameters by the sensor readings. The variables in equation 50 have the same notation as in section 2.

$$T^{obs} = m(\vec{a} + \vec{g}) - R(\eta)F_p + F_{ext} \tag{50}$$

## *4.3 Filtering process using a non-constant model*

In section 4.2, the Kalman filter models the tension of the tether as being constant. This section introduces a model that takes into account the first and the second derivatives, in order to predict the pulling movement that may be applied. To simplify the development of the model, without loss of generality, consider only the tension along the $x$ direction. Assume that equation 51 defines the Kalman model in discrete time.

$$\begin{gathered} Tx_{k+1} = Tx_k + a(Tx_k - Tx_{k-1}) + \\ + b(Tx_k - Tx_{k-1} - (Tx_{k-1} - Tx_{k-2})) \end{gathered} \tag{51}$$

Equation 52 represents the state space derived from the equation 51.

$$\begin{bmatrix} x_{1[k+1]} \\ x_{2[k+1]} \\ x_{3[k+1]} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ b & -(a+2b) & (1+a+b) \end{bmatrix} \begin{bmatrix} x_{1[k]} \\ x_{2[k]} \\ x_{3[k]} \end{bmatrix} \tag{52}$$

Equation 53 presents the matrices A, B, C, D, as well as the state estimate and the observation vector of the Kalman filter - $\hat{x_k}$ and $y_k$. The tension $T_x^{obs}$ is obtained through equation 50.

$$\begin{gathered} \hat{x}_k = \begin{bmatrix} Tx_{k-2} \\ Tx_{k-1} \\ Tx_k \end{bmatrix}, y_k = Tx_k^{obs}, \\ A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ b & -(a+2b) & 1+a+b \end{bmatrix}, C^T = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, B = D = 0 \end{gathered} \tag{53}$$

## 4.4 Simulation results - Tension estimation

### 4.4.1 Kalman filter using a constant model

To simulate the high level of noise that the real sensors may have, the simulated reading of the accelerations, attitude, and thrust pass through an Additive White Gaussian Noise channel (AWGN), in Matlab. Figure 5 presents the tension estimates, before and after the Kalman filtering, taking as input a step signal of magnitude $-0.02N$ at time $t = 140s$, concerning the $x$ direction.

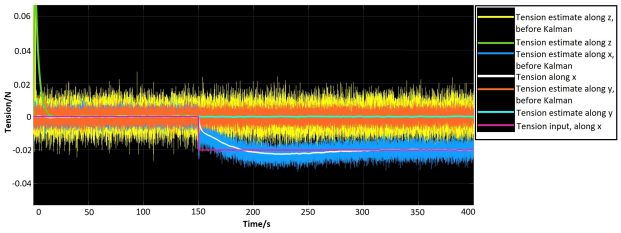



Figure 5: Noisy measurements from the sensors, the tension's true value, and the estimated tension.

On the one hand, the Kalman filter has a significant impact on reducing the tension's noise. On the other hand, and since a constant model is used, it introduces a delay when abrupt changes are applied.

### 4.4.2 Comparison between the constant and non-constant model in Kalman filter

The performance of these two models is tested in Matlab for the case where the applied tension is a step signal, but without considering the drag effect.

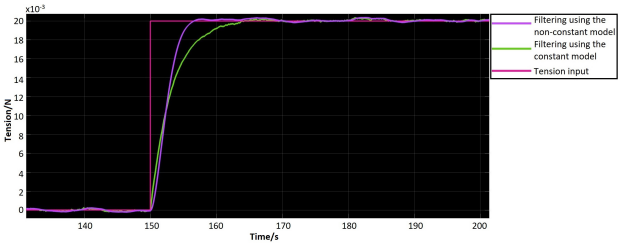



Figure 6: Real tension (pink), tension estimated using the constant model (green) and tension estimated using the derivative model (purple).

As it can be observed, the new model presents slight improvements since it can converge faster to the real value and it also presents less noise for the steady values, as illustrated in figure 7.

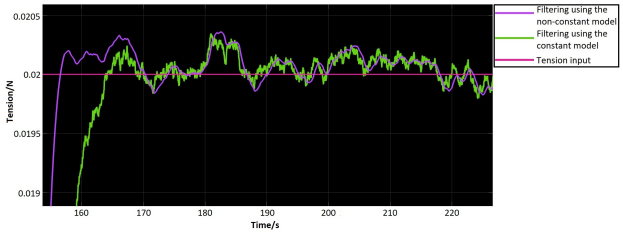



Figure 7: Zoom of figure 6 for steady values.

These results may look contradictory since the initial model is ideally "perfect" for modeling constant values. However, this new model allows to increase the confidence in the model (by decreasing the process noise variance), making it less susceptible to the observation noise without losing the ability of following fast changes.

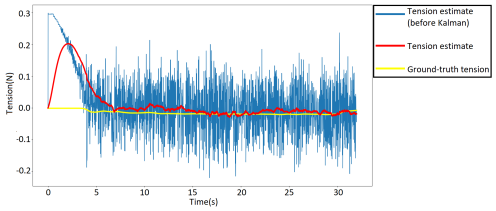
## 4.5 Experimental results

### 4.5.1 Estimation of the vertical tension

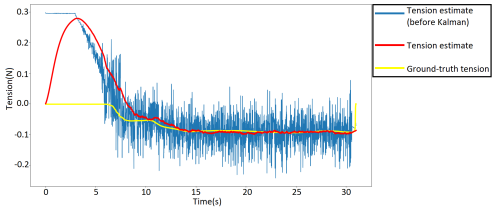
The validation of the vertical tension estimate is done by performing a vertical tethered takeoff, which represents the simplest experiment to compute its ground-truth value $Tv_{gt}$. This last one is calculated through the height of the quadcopter $z$, and the weight per length unit of the the tether $\omega$.

$$Tv_{gt} = \omega.z \tag{54}$$

Figure 8 illustrates the estimation of the vertical tension computed in two different scenarios: for a vertical takeoff of height **0,3m** and **1,3m**. As it can be observed, the estimated tension converge to the ground-truth value, and the implementation of a Kalman filter is essential to suppress the sensor's noise.




**(a)** Hovering at a height of 0,3m.




**(b)** Hovering at a height of 1,3m.

Figure 8: Ground-truth values and estimates of the vertical tension applied to the UAV.

The tension estimation procedure does not take into account the force that the ground exerts on the quadcopter. Thus, in the initial instants, the vertical ten-

sion estimate does not correspond to its ground-truth values.

#### 4.5.2 *Estimation of the horizontal tension*

To validate the horizontal tension estimate, its ground-truth value is computed using a small mass (coin) attached to a wire. Figure 9 illustrates the scheme of the test-bench used to compute the ground-truth values for the horizontal tension.

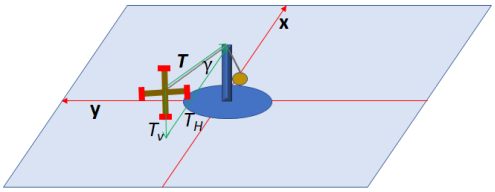


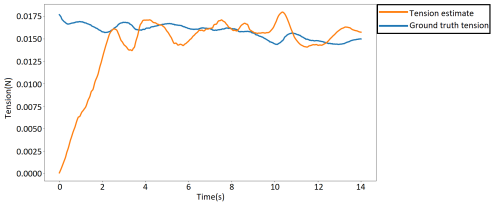

Figure 9: Test-bench for computing the horizontal tension.

Equation 55 computes the ground-truth value of the horizontal tension, in which T corresponds to the weight of the mass and $\gamma$ is computed according to equation 56 - $z_q$ and $z_a$ are the height of the quadcopter and the height of the vertical arm, respectively, and $r_q$ is the quadcopter's radial coordinate, assuming the inertial frame presented in figure 9.

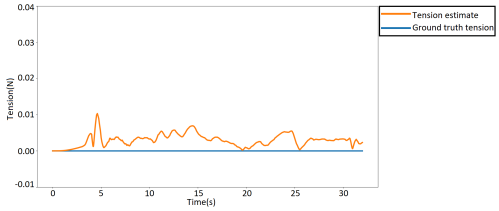
$$T_H = cos(\gamma).T \tag{55}$$

$$\gamma = tan^{-1}(\frac{z_q - z_a}{r_q}) \tag{56}$$

Figure 10(a) displays the tension estimate and the ground-truth value for an attached mass of 2,3g, and figure 10(b) shows the result for a tethered vertical takeoff, where the tether direction is mainly vertical.




**(a)** Hovering flight at position $x = -0.3$, $y = z = 0$.




**(b)** Hovering flight at $x = y = 0$ and $z = 0.4$.

Figure 10: Ground-truth values and estimates of the horizontal tension applied to the UAV.

As figure 10(b) displays, the horizontal tension estimate is nearly null - less than 0.01N.

# 5 Tension-based flight control

## 5.1 *Tension following*

The tension applied to the quadcopter is estimated in real-time, according to section 4. The goal position of the quadcopter changes if the estimated tension is greater than a pre-defined threshold. Thus, when this occurs, the goal position of the UAV is successively updated to its current position, making the quadcopter to follow the pull's direction. When the tension applied to the quadcopter is no longer greater than the pre-defined threshold, the goal position stops being updated and the quadcopter remains hovering at its last position.

To observe the behaviour of the *tension following* feature a few videos were taken [2], where figures 11 and 12 correspond to screenshots of those videos.

Figure 11: Screenshots of the quadcopter following the tension's direction.

Moreover, the implementation of the *tension following* feature can also be used for the landing process. Given that, after the first tug, a flag is activated indicating that the *tension following* feature is on. Thus, if the quadcopter flies under a certain height, the motors are turned off. Without using an external motion system, the same principle can be applied using a sensor distance, which deactivates the motors if the distance to the landing platform is smaller than a threshold. Figure 12 illustrates the mentioned landing process.

Figure 12: Screenshots of the landing of the quadcopter using the *tension following* feature.

## 5.2 *Tension-based position control*

In the previous section, the quadcopter's position is updated if the estimated tension is greater than a pre-defined threshold. This means that it is necessary to know the quadcopter's current position, which implies the use of an external motion system. An alternative way is to assume that the tether outlines a catenary curve and the outer loop controller no longer uses the goal and current position but, instead, uses a goal tension and a current tension, as shown in figure 13.

[2] The full videos are presented on Youtube here.

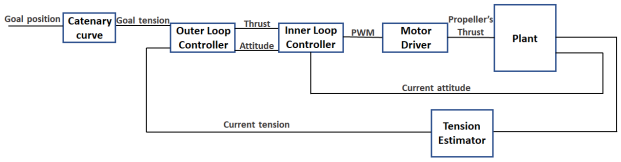
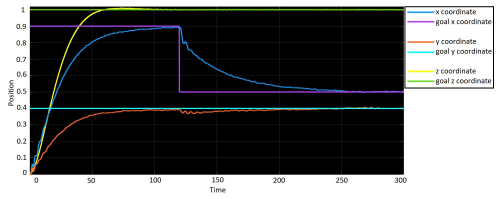
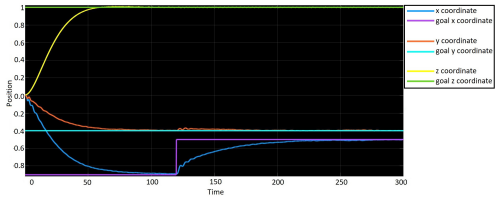



Figure 13: Tension-based Control Structure.

On the one hand, the quadcopter's current tension is estimated during flight as shown before, using the attitude, thrust, and acceleration of the quadcopter (see section 4). On the other hand, the goal tension is computed by providing a goal position, regarding the tether's frame, and using the catenary's properties to compute the goal tension assuming to a perfect catenary (see section 3).
Figure 14 presents the simulation results from Matlab, obtained using the control structure from figure 13. However, due to the inaccurate estimation of the vertical tension in the initial instants, the altitude's feedback loop of the quadcopter uses the altitude values - the goal and the current altitude - instead of using the values of the vertical tension. In practice, the altitude can be computed using a distance sensor or, for cases of higher altitudes, by using an altimeter sensor, without needing the use of an external motion system.




**(a)** Current and goal position along $x$, $y$ and $z$, flight test one.




**(b)** Current and goal position along $x$, $y$ and $z$, flight test two.

Figure 14: Simulation results of the UAV's flight using a tension-based position control.

# 6 Position Estimation

## 6.1 *Matching the lowest point of the tether to the origin*

Using the equations previously defined in section 3, and additionally assuming that $r_0 = 0$[3], it is developed a procedure to recover the position of the quadcopter. The tether's parameter $\langle a \rangle$ and the tether's length $\langle s \rangle$ are computed based on the horizontal tension $H$ and the vertical tension $T_v$, according to equations 57 and 58, respectively.

$$a = \frac{T_H}{\omega} \tag{57}$$

$$s_2 = \frac{T_v}{\omega} \tag{58}$$

Remember that the length variable $\langle s_2 \rangle$ is the arc-length distance from $\langle r_0 \rangle$ (radial distance of the tether's lowest point) to the quadcopter.
Using the relation between the length of the tether with the radial distance $r$ - distance along the plane of the catenary curve - and the tether parameter $\langle a \rangle$, one can obtain the value of $r$.

$$r = a.sinh^{-1}(\frac{s_2}{a}) \tag{59}$$

The altitude can be directly computed according to equation 60, also mentioned in section 2.

$$z = a.cosh(\frac{r}{a}) - a \tag{60}$$

The radial distance is re-written in terms of $x$ and $y$ distances (see equations 61 and 62 ,respectively), where the angle between them corresponds to $\beta = tan^{-1}\left(\frac{T_y}{T_x}\right)$ (see figure 3). The tensions $T_x$ and $T_y$ are the tension's estimates along the $x$ and $y$ direction, respectively, regarding the world frame.

$$x = r.cos(\beta) \tag{61}$$

$$y = r.sin(\beta) \tag{62}$$

The horizontal tension $H$ of a vertical takeoff is nearly null, which implies that the tether parameter $\langle a \rangle$ is also going to be approximately zero (equation 57). This raises an indetermination of type $0 \times \infty$ in equations 59 and 60. The limits of those equations when $a \to 0$ are presented in equation 64 and 63.

$$\lim_{a \to 0} z = \lim_{a \to 0} a.cosh(\frac{r}{a}) - a = |s_2| \tag{63}$$

$$\lim_{a \to 0} r = \lim_{a \to 0} a.sinh^{-1}(\frac{s}{a}) = 0 \tag{64}$$

## 6.2 *Knowing the total length of the tether*

In a situation where $r_0 \neq 0$, another way of recovering the position of the quadcopter is by knowing the tether's full length. Using, once more, the equations defined in section 3, it is possible to compute the tether parameters $\langle a \rangle$ and $\langle s_2 \rangle$ (see equations 57 and 58). Notice, however, that the length $\langle s_2 \rangle$ no longer matches the total length since $\langle r_0 \rangle$ cannot be considered null (equation 65).

$$s_{tot} = s_2 + s_1 \tag{65}$$

This way, replacing $x_0$ by $r_0$ in expression $s_1$ (see equation 29), and using the relation presented in equation 65, one can derive equation 66.

$$r_0 = r_i + a.sinh^{-1}(\frac{s_{tot} - s_2}{a}) \tag{66}$$

[3]The $r$ variable is the horizontal axis $x$ of the catenary in a $\Re^3$ space (section 3).

Furthermore, by replacing $x_0$ by $r_0$ in expression $s_2$ (equation 30), the radial distance $r$ comes as:

$$r = r_0 + a.sinh^{-1}(\frac{s_2}{a}) \tag{67}$$

At last, equation 68 uses the catenary's expression and computes the quadcopter's altitude.

$$z = a.cosh(\frac{r - r_0}{a}) + C \tag{68}$$

Equation 69 computes parameter $\langle C \rangle$.

$$C = y_i - a.cosh(\frac{x_i - x_0}{a}) \tag{69}$$

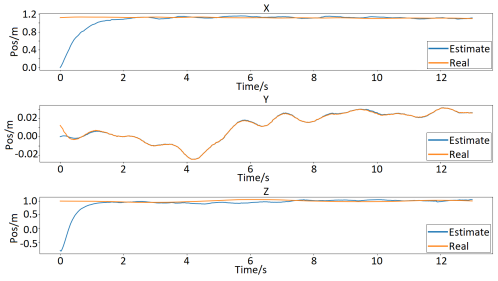
The vertical takeoff implies that equations 67 and 68 have a mathematical indetermination of type $0 \times \infty$. The limits of those equations are presented in equations 70 and 71.

$$\lim_{a \to 0} z = \lim_{a \to 0} a.cosh(\frac{r}{a}) + C = z_i + |s_2| - |s_1| \tag{70}$$

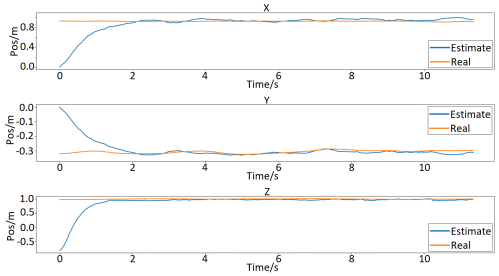
$$\lim_{a \to 0} r = \lim_{a \to 0} a.sinh^{-1}(\frac{s_1}{a}) + a.sinh^{-1}(\frac{s_2}{a}) = 0 \tag{71}$$

### *6.3 Experimental Results - knowing the tether full length*

To avoid the modeling of the tether's oscillations, the results presented throughout this section concern hovering flights. Additionally, the angle $\beta$, which relates the radial distance with the $x$ and $y$ coordinates, is also a source of inaccuracy. It is initially assumed that the $\beta$ angle is known and is computed using the Mocap system. In practice, the angle $\beta$ could be computed using a visual or mechanical system on the ground controller that could indicate the direction of the tether. Figure 15 presents a pair of experiments in which the height corresponds to 1m and the radial distances are similar between them.



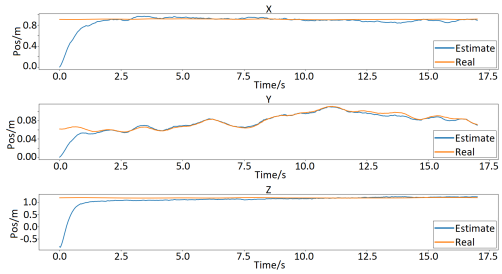

**(a)** Hovering at $x \simeq 1,2m$, $y \simeq 0m$ and $z \simeq 1m$.



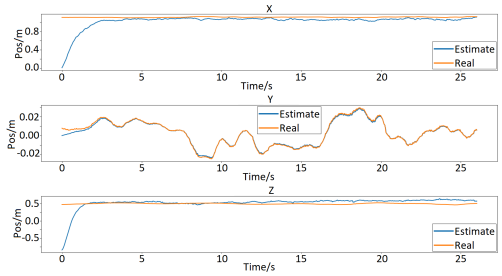

**(b)** Hovering at $x \simeq 1m$, $y \simeq -0.3m$ and $z \simeq 1m$.

Figure 15: Hovering at same height $z$ and similar radial distance $(x, y)$ for small angle $\beta$ $(< 20^o)$.

In a second set of experiments (figure 16) the radial distances are also similar between them but the estimation of the quadcopter's height is evaluated for two different heights - 0,5m and 1,2m.



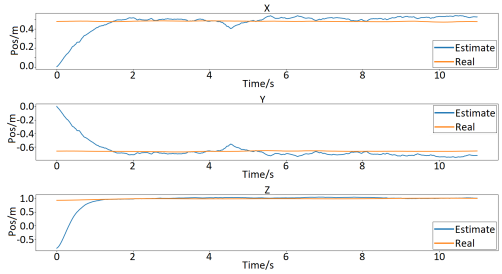

**(a)** Hovering at $x \simeq 1m$, $y \simeq 0m$ and $z \simeq 1,2m$.



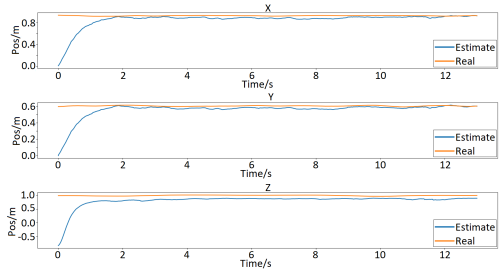

**(b)** Hovering at $x \simeq 1m$, $y \simeq 0m$ and $z \simeq 0,5m$.

Figure 16: Hovering at different heights $z$ and similar radial distance $(x, y)$ for small angle $\beta$ $(< 10^o)$.

A third set of experiment (figure 17) is performed with a bigger range for the value of the $y$ coordinate.




**(a)** Hovering at $x \simeq 0,5m$, $y \simeq -0.6m$ and $z \simeq 1m$.




**(b)** Hovering at $x \simeq 0,9m$, $y \simeq 0.6m$ and $z \simeq 1m$.

Figure 17: Hovering at same height $z$ and different radial distance $(x, y)$ for non-small angle $\beta$ $(> 30^o)$.

In the initial instants the Kalman filter assumes that the tensions $T_x$ and $T_y$ are null, which means that the

length $\langle s_2 \rangle$ and the tether parameter $\langle a \rangle$ also start to be zero (see equations 72 and 73).

$$s_2 = \frac{T_v}{\omega} \tag{72}$$

$$a = \frac{T_H}{\omega} \tag{73}$$

According to the expression deduced in equation 71, the initial estimated position of the radial coordinate is null, implying that $x$ and $y$ coordinates are also null. On the one hand, the equation 70 allows to infer that the initial estimate regarding the altitude corresponds to $z_i - |s_1|$, since $s_2$ is zero. On the other hand, the tether's total length is given by equation 65, which means that $s_1 = s_{tot}$ for a null length $s_2$. Equation 74 presents the initial estimate of the altitude over this set of experiments.

$$z = z_i - |s_1| = 0,754 - 1,6 = -0,846 \tag{74}$$

# 7 Conclusions

This work presented a method to estimate the tension applied to a quadrotor by using measurements from the IMU sensors. Due to the high level of noise, two Kalman based filtering processes were introduced. Furthermore, the tension estimate was used to present alternative ways of controlling the quadcopter, and to improve the position estimate of a tethered quadcopter.

The first flight control strategy used the tension estimate to update the quadcopter's position. Nevertheless, the position of the quadcopter must be known through an external motion system. Aiming to develop a control strategy that does not need to know the position of the UAV, a novel methodology that uses the tether's shape and the tension estimate was introduced.

Moreover, this study presented a method to estimate the position of the quadcopter based on the tension that the tether applies to the UAV and its shape.

## *7.1 Future Work*

The interaction between a ground-robot and a UAV allows almost unlimited feasible features. Using the work presented throughout this thesis, the following future implementations are suggested:

- To implement the tension-based position control (see section 5.2) into a real-life scenario.
- To develop the physical ground-controller to maintain the tether lowest point at the origin.
- To include in the ground-controller the mechanism to compute the angle $\beta$ that split the radial distance in the $x$ and $y$ components.

# References


[1] Brendan Galea and Paul G Kry. Tethered flight control of a small quadrotor robot for stippling. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1713–1718. IEEE, 2017.

[2] Marco Tognon. Attitude and tension control of a tethered formation of aerial vehicles. Master's thesis, Università degli Studi di Padova, 2014.

[3] Lorenzo Fagiano. Systems of tethered multicopters: modeling and control design. In *IFAC-PapersOnLine*, volume 50 of *1*, pages 4610–4615. International Federation of Automatic Control, Elsevier, 2017.

[4] Seiga Kiribayashi, Kaede Yakushigawa, and Keiji Nagatani. Design and development of tether-powered multirotor micro unmanned aerial vehicle system for remote-controlled construction machine. In *Field and Service Robotics*, pages 637–648. Springer, 2018.

[5] Robert Severinghaus and John Kaniarz. Quadrotor for increased situational awareness for ground vehicles. Technical report, Army tank automotive research development and engineering center warren MI, 2015.

[6] Peter Savnik. Tether control for unmanned aerial vehicle - creating a platform for tether control of uav. Bachelor thesis, Technical University of Denmark, 01 2015.

[7] Seiga Kiribayashi, Jun Ashizawa, and Keiji Nagatani. Modeling and design of tether powered multicopter. In *2015 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pages 1–7. IEEE, 2015.

[8] Wasantha Samarathunga, Guangwei Wang, and Shiqin Wang. Auxiliary power unit evaluation for tethered uav. *International Journal of New Technology and Research*, 2(7), July 2016.

[9] Abid Sulficar, Harikrishnan Suresh, Aravind Varma, and Arjun Radhakrishnan. *Modeling, simulation and complete control of a quadcopter.* Bachelor thesis, National Institute of Technology Karnataka Surathkal, 2017.

[10] Francesco Sabatino. Quadrotor control: modeling, nonlinearcontrol design, and simulation. Master's thesis, KTH Royal Institute of Technology, 2015.

[11] Julian Förster. System identification of the crazyflie 2.0 nano quadrocopter. Bachelor thesis, ETH Zurich, 2015.

[12] E.Russel Johnston Ferdinand P.Beer. *Mecânica Vetorial para Engenheiros, Estática, Volume I.* McGraw-Hill, 1980.