

# Stabilization of a Quadcopter

Alejandro Alonso García  
Department of electronic systems  
Control and Automation  
Aalborg University  
Email: aalons16@student.aau.dk

Amalie V. Petersen  
Department of electronic systems  
Control and Automation  
Aalborg University  
Email: @student.aau.dk

Andrea Victoria Tram Løvemærke  
Department of electronic systems  
Control and Automation  
Aalborg University  
Email: @student.aau.dk

Niels Skov Vestergaard  
Department of electronic systems  
Control and Automation  
Aalborg University  
Email: nveste12@student.aau.dk

Noelia Villarmarzo Arruñada  
Department of electronic systems  
Control and Automation  
Aalborg University  
Email: nvilla16@student.aau.dk

**Abstract**—Abstract goes here.

## I. INTRODUCTION

In the last years, the interest for quadcopters has increased due to the great possibilities they offer. Among these, the most well-known ones are surveillance, inspection of big structures and search and rescue missions in difficult environments [1].

The quadcopter constitutes a control challenge due to its unstable nature and coupled behavior. The system has 6 degrees of freedom, the 3 position coordinates and the 3 orientations, and there are only four actuation variables which are the motor velocities. The dimension of the problem is explained by McKerrow in [2].

The control of a quadcopter has been addressed many times in the recent years. In Mian et al. [3] the quadcopter is controlled using a back-stepping technique and non-linear controllers. Other way of solving the issue is presented in Tayebi et al. [4] in which the quadcopter attitude is modeled using quaternions and controlled with a PD based controller. In [5], Mian and Yang model the system using its dynamic equations and use non linear controllers to achieve a steady flight while in Mokhtari et al. [6] the system is controlled by a mixture of a robust feedback linearization and a linear  $GH_\infty$ .

The approach presented here models the quadcopter by a first principles method. This approach yields a non linear model that describes the attitude and translational behavior of the quadcopter. The model is then linearized around an equilibrium point, which is chosen to be in hovering position. With the linearized equations, controllers for attitude and translational behaviors are designed. The angular controller is obtained by means of a State Space representation while the translational controller is designed using classical control techniques. In the control system, the translational constitutes an outer loop and sets the reference for the attitude controller.

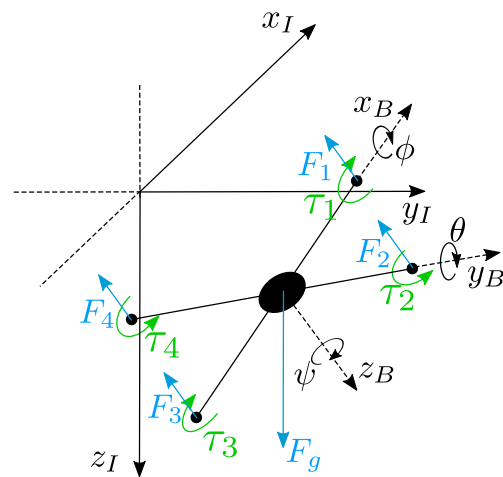
In the last part of the paper, the simulations and experimental results of the designed controllers are shown and discussed.

## II. METHOD

The methods presented in this section are constituted by the modelling of the system, the linearization of the obtained model and the design of the controllers.

### A. Model

The quadcopter system is shown in Figure 1. As it can be seen, the system is modeled by using two coordinate frames. The inertial frame is utilized to describe the translational movement while the body frame is attached to the quadcopter and used to characterize its attitude behavior. In the figure, also the positive references for rotational and translational movements are depicted, as well as the main forces and torques acting on the quadcopter.



**Figure 1:** Quadcopter diagram showing the forces and torques acting on the system and the positive references chosen for rotations and translations in both Inertial and Body coordinate frames.

The forces generated in the propeller are easily explained in the Body coordinate frame. In order to represent them in the Inertial frame a rotation matrix is used. It is built considering a 123 rotation sequence [7]

The dynamic model of the quadcopter can be explain through three sets of equations. The first describes the motor and the propeller, the second presents the attitude response of the quadcopter and the third explains how the translational variables of the system evolve with time.

1) *Motor and Propeller*: The four motors in the quadcopter generate the rotation required so the propeller creates the force that lifts the quadcopter. This force is called thrust force and can be modeled as proportional to the square of the motor rotational velocity. MAYBE SOURCE. The coefficient for this equation is called thrust coefficient and has been found by experiments. The rotation of the propellers also generates a torque on each motor due to drag between air and propeller. Drag torque is compensated in the quadcopter by having two of the motors turning in one direction and the two others in the opposite. It can also be described as proportional to the square of the velocity by terms of a drag coefficient that has also been obtained through tests. Equation 1 and 2 show the expression for the thrust force and drag torque caused by the rotation of the propeller.

$$F = k_{th}\omega^2 \quad (1)$$

$$\tau = k_d\omega^2 \quad (2)$$

#### INCLUDE WHERE

This equations are included in the attitude and translational models derived below.

2) *Attitude Model*: The attitude model equations are based on Newton Second Law for rotational movement and are represented in Equation 3, 4 and 5.

$$J_x\ddot{\phi} = k_{th}(\omega_4^2 - \omega_2^2)L \quad (3)$$

$$J_y\ddot{\theta} = k_{th}(\omega_1^2 - \omega_3^2)L \quad (4)$$

$$J_z\ddot{\psi} = k_d(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \quad (5)$$

#### INCLUDE WHERE

The expressions above state how the thrust force difference between motors 1 and 3 affects the roll angular acceleration, how that between motors 4 and 2 affects the pitch angle and how the yaw acceleration depends on the four motors by means of the drag torque generated on the propeller.

3) *Translational Model*: The equations describing the response of the system along the x, y and z axes is derived from Newton's Second Law. The forces that act on the system are those from the propellers and the gravitational. These expressions are shown in Equation 6, 7 and 8. FIX EQUATIONS

$$m\ddot{x}_I = -k_{th}(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2)\sin(\theta) \quad (6)$$

$$m\ddot{y}_I = -k_{th}(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2)(-\sin(\phi)) \cdot \cos(\theta) \quad (7)$$

$$m\ddot{z}_I = F_g - k_{th}(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2)\cos(\phi) \cdot \cos(\theta) \quad (8)$$

It is worth mentioning that, as the thrust forces always point in the negative z direction in the body coordinate frame, the accelerations in x and y directions in the inertial frame are zero as long as pitch and roll angles are 0.

#### B. Linearization

The linearization of the model equations has been developed following the first order Taylor approximation around an equilibrium point of the system. The chosen point is the hovering position and that implies that all variables have a value of zero, that is, the translational and attitude accelerations, velocities and positions. Choosing a zero acceleration equilibrium point along the Inertial z axis yields a equilibrium rotational speeds so that the necessary thrust is generated to compensate for the gravitational force.

The resulting equations for the attitude model after the linearization are shown in Equation 9, 10 and 11.

$$J_x\Delta\ddot{\phi} = 2k_{th}L(\bar{\omega}_4\Delta\omega_2 - \bar{\omega}_2\Delta\omega_4) \quad (9)$$

$$J_y\Delta\ddot{\theta} = 2k_{th}L(\bar{\omega}_1\Delta\omega_1 - \bar{\omega}_3\Delta\omega_3) \quad (10)$$

$$J_z\Delta\ddot{\psi} = 2k_d(\bar{\omega}_1\Delta\omega_1 - \bar{\omega}_2\Delta\omega_2 + \bar{\omega}_3\Delta\omega_3 - \bar{\omega}_4\Delta\omega_4) \quad (11)$$

INCLUDEWHERE Similarly, the equations of the translational model are linearized. The result is shown in 12, 13 and 14. FIX EQUATIONS

$$m \cdot \Delta\ddot{x}_I = -k_{th}(\bar{\omega}_1^2 + \bar{\omega}_2^2 + \bar{\omega}_3^2 + \bar{\omega}_4^2)\cos(\bar{\theta})\Delta\theta \quad (12)$$

$$m \cdot \Delta\ddot{y}_I = k_{th}(\bar{\omega}_1^2 + \bar{\omega}_2^2 + \bar{\omega}_3^2 + \bar{\omega}_4^2)\cos(\bar{\phi})\cos(\bar{\theta})\Delta\phi \quad (13)$$

$$m\Delta\ddot{z}_I = -2k_{th}(\bar{\omega}_1\Delta\omega_1 + \bar{\omega}_2\Delta\omega_2 + \bar{\omega}_3\Delta\omega_3 + \bar{\omega}_4\Delta\omega_4)\cos(\bar{\phi})\cos(\bar{\theta}) \quad (14)$$

#### INCLUDEWHERE

#### C. Control

The control of the system is divided into two control systems. One handles the attitude equations and the other controls the translational variables.

1) *Attitude Controller*: The attitude controller for the quadcopter has been designed using a state space representation of the system. This helps handling the coupled angular response of the quadcopter. The chosen states for the system are the three angular positions and the three angular velocities. The input of the attitude system are the four motor rotational speeds and the output are the three angles roll, pitch and yaw. Below, the state vector, the input and the output are represented.

$$\mathbf{x}(t) = [\phi \quad \theta \quad \psi \quad \dot{\phi} \quad \dot{\theta} \quad \dot{\psi}]^T$$

$$\mathbf{y}(t) = [\phi \quad \theta \quad \psi]^T$$

$$\mathbf{u}(t) = [\omega_1 \quad \omega_2 \quad \omega_3 \quad \omega_4]^T$$

The state space matrix representation is displayed in Equation 15 and 16.

$$\dot{\mathbf{x}}(t) = \mathbf{A} \cdot \mathbf{x}(t) + \mathbf{B} \cdot \mathbf{u}(t) \quad (15)$$

$$\mathbf{y}(t) = \mathbf{C} \cdot \mathbf{x}(t) + \mathbf{D} \cdot \mathbf{u}(t) \quad (16)$$

INCLUDEWHERE The values for the A, B, C and D matrices are obtained from the linearized attitude equations,

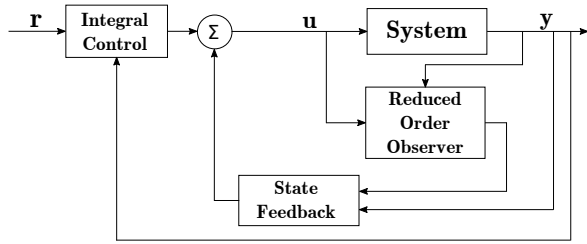
yielding the matrices shown below. As D is a zero matrix, only A, B and C are shown.

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & -\frac{2 \cdot k_{th} \cdot L \cdot \bar{\omega}_2}{J_x} & 0 & \frac{2 \cdot k_{th} \cdot L \cdot \bar{\omega}_4}{J_x} \\ \frac{2 \cdot k_{th} \cdot L \cdot \bar{\omega}_1}{J_y} & 0 & -\frac{2 \cdot k_{th} \cdot L \cdot \bar{\omega}_3}{J_y} & 0 \\ \frac{2 \cdot k_d \cdot \bar{\omega}_1}{J_z} & -\frac{2 \cdot k_d \cdot \bar{\omega}_2}{J_z} & \frac{2 \cdot k_d \cdot \bar{\omega}_3}{J_z} & -\frac{2 \cdot k_d \cdot \bar{\omega}_4}{J_z} \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

The designed attitude controller is depicted in the block diagram shown in Figure 2.



**Figure 2:** Control structure for regulating the attitude response of the quadcopter. It includes the state feedback, the reduced order observer and the integral term.

## 2) Translational Controller:

### III. RESULTS

Simulation vs. reality.

Comment on the results and how that correlates with reality, without discussing possible issues or improvements.

### IV. DISCUSSION

Discussing possible issues or improvements of the above results.

### V. CONCLUSION

Summary - what we want the reader to remember.

### VI. ACKNOWLEDGMENTS

Henrik Schiøler, associated professor at Aalborg University  
Christoffer Sloth, associated professor at Aalborg University

### REFERENCES

- [1] *10 incredibly interesting uses for drones*, Web Page, 2014. [Online]. Available: <http://dronebuff.com/uses-for-drones/>.
- [2] P. McKerrow, "Modelling the draganflyer four-rotor helicopter", *Proceedings of the 2004 IEEE International Conference on Robotics and Automation*, 2004.
- [3] D. W. Ashfaq Ahmad Mian Mian Ilyas Ahmad, "Back-stepping based nonlinear flight control strategy for 6 dof aerial robot", *International Conference on Smart Manufacturing Application*, 2008.
- [4] S. M. A. Tayebi, "Attitude stabilization of a four-rotor aerial robot", *43rd IEEE Conference on Decision and Control*, 2004.
- [5] D.-b. W. Ashfaq Ahmad Mian, "Dynamic modeling and nonlinear control strategy for an underactuated quad rotor rotorcraft", *Journal of Zhejiang University SCIENCE A*, 2008.
- [6] B. D. A. Mokhtari A. Benallegue, "Robust feedback linearization and gain controller for a quadrotor unmanned aerial vehicle", *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2005.
- [7] E. W. Weisstein, "Rotation matrix", *MathWorld—A Wolfram Web Resource*, 2016.