# Motion Control of Unmanned Aerial Vehicle

Vilhelm Dinevik and Paula Carbó

***Abstract*—The aim of this report is to investigate how to safely navigate multiple quadrotor UAVs to their respective goals in a 3D-environment filled with obstacles. This is done by first designing a mathematical model for a single UAV using kinematics and dynamics based on mechanics....The mathematical model is then linearized so it could be written as a steady state function, and the system's observability and controllability is then checked. An LQR controller is designed .... To navigate the drone to the goal we designed a potentialfield function The potentialfield function was then implemented in to a matlab simulation..The conclusions from the report is that the potentialfield method is a very fast and easy way to implement a navigation system although the drones sometimes get stuck behind obstacles.Future development could be to actually implement the simulation in to a real UAV to see how it performs.**

## I. Introduction

UNMANNED aerial vehicles, also known as UAVs, are becoming nowadays more and more popular because they are small, cheap to produce, have low operating and maintenance cost, have great maneuverability, can perform steady flight operations and are able to enter high-risk areas without having to compromise human safety. Most applications that involve UAVs have been used in open areas without any obstacles and with a human in control of the UAV. But in recent years people have come up with more modern applications of UAVs that will need UAVs to fly autonomously in densely populated areas, with a lot of other autonomous UAVs around, e.g. Amazon Prime Air delivery system, AltiGator drones services for inspection and data adquisition, or multi-UAVs used to deploy an aerial communications network. This places high demands on UAVs obstacle avoidance capabilities for both moving and static obstacles.

There are many different manufacturers and a vast amount of different UAV models, all with different motors, weights, sensors and lift-to-weight ratio. To make a standard autonomous flight applicable to all these kinds of UAVs, a simple and easy-to-implement multi-UAV mathematical model, that will still be able to avoid obstacles with as few sensors as possible, is needed.

This project aims to develop a navigation method that can make multiple UAVs fly safely to their goals in an environment filled with obstacles. To assure that the vehicles move according to what is desired, a fast and reliable controller is needed. For this purpose a mathematical description of the UAV and the sensors onboard it is needed. The mathematical model's controllabillity and observability have to be verified.

The section II where the mathematical model for the quad is developed. In section III, an LQR controlled is studied. Section IV analyses the recommended sensors onboard the UAV and how to measure the state accurately with them. In section V, the navigation method applied in our case, potential fields, is explained and described. The simulation performed is explained in section VI. Finally, in the last two sections of this paper, the results are presented, explained and discussed.

## II. Quadcopter modelling

### A. Overview

The UAV is a rigid body quadcopter, with a cross-shaped body and four electrical propellers. Front and rear rotors rotate in a clockwise direction, while right and left rotors rotate in a counterclockwise direction, this is illustrated in Fig.1. Its motion has 6 degrees of freedom but there are only 4 propellers, therefore the system is underactuated.

### B. Kinematics

In order to describe its motion, a kinematic model for the UAV was developed. Two right-hand reference frames are defined: the Earth frame and the body frame, as can be seen in Fig.1.

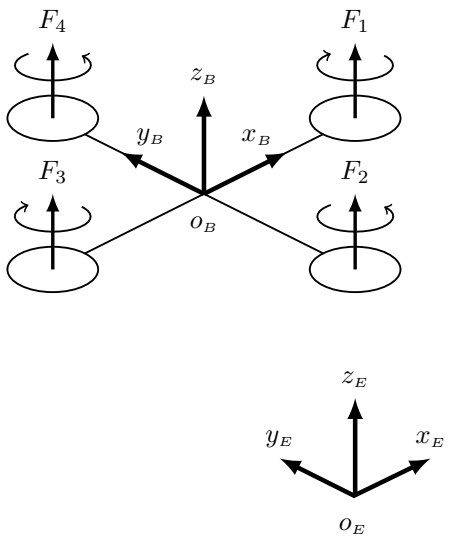


Fig. 1. Quadrotor with propellers and the two reference frames

The Earth frame is static, with the $x_E$ axis pointing towards the North, the $y_E$ axis pointing towards the West, and $z_E$ pointing upwards w.r.t. the Earth. The body frame is attached to the UAV, with the $x_B$ axis pointing towards the quadrotor's front, the $y_B$ axis pointing towards the left, and the $z_B$ axis pointing upwards. In this case, the axis origin $o_B$ coincides with the quadrotor's center of mass.

The generalized position $\boldsymbol{\xi}$ contains the linear and angular position and is described in the Earth frame, as in (1). The linear position $\boldsymbol{x}^E$ of the UAV is the vector between the origin

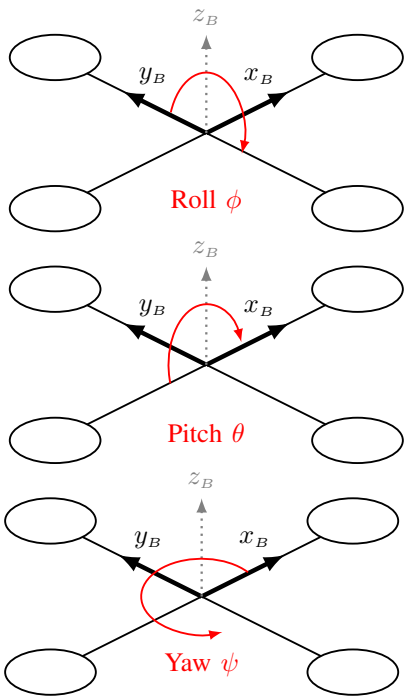


Fig. 2. Euler angles

of the Earth frame $o_E$ and the origin of the body frame $o_B$, and the Euler angles $\boldsymbol{\eta}^E$ are defined as stated in Fig.2.

$$\boldsymbol{\xi} = \begin{bmatrix} \boldsymbol{x}^E & \boldsymbol{\eta}^E \end{bmatrix}^T = \begin{bmatrix} x & y & z & \phi & \theta & \psi \end{bmatrix}^T \quad (1)$$

The generalized velocity $\boldsymbol{\nu}$ (2) contains the linear and angular velocity, and it is expressed in the body frame.

$$\boldsymbol{\nu} = \begin{bmatrix} \boldsymbol{v}^B & \boldsymbol{\omega}^B \end{bmatrix}^T = \begin{bmatrix} u & v & w & p & q & r \end{bmatrix}^T \quad (2)$$

Three rotation matrixes around each of the $x, y, z$ axes can be defined according to (3, 4, 5) respectively.

$$\boldsymbol{R}_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi) & \cos(\phi) \end{bmatrix} \quad (3)$$

$$\boldsymbol{R}_y(\theta) = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \quad (4)$$

$$\boldsymbol{R}_z(\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (5)$$

The complete rotation matrix $\boldsymbol{R}_\Theta$, that expresses the rotation from the body frame to the Earth frame, can be obtained by multiplying these three matrixes, as in (6).

$$\boldsymbol{R}_\Theta(\phi, \theta, \psi) = \boldsymbol{R}_x(\phi)\boldsymbol{R}_y(\theta)\boldsymbol{R}_z(\psi) \quad (6)$$

The transfer matrix $\boldsymbol{T}_\Theta$ that allows to change between the angular velocity in the body frame $\boldsymbol{\omega}^B$ and the Euler rates in the Earth frame $\dot{\boldsymbol{\eta}}^E$ can be determined and is as shown in (7).

$$\boldsymbol{T}_\Theta(\phi, \theta) = \begin{bmatrix} 1 & \sin(\phi)\cdot\tan(\theta) & \cos(\phi)\cdot\tan(\theta) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi)/\cos(\theta) & \cos(\phi)/\cos(\theta) \end{bmatrix} \quad (7)$$

A generalized matrix $\boldsymbol{J}_\Theta$ can be built joining the rotation and the transfer matrix (6, 7), as shown in (8).

$$\boldsymbol{J}_\Theta(\phi, \theta, \psi) = \begin{bmatrix} \boldsymbol{R}_\Theta & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{T}_\Theta \end{bmatrix} \quad (8)$$

Where the notation $\boldsymbol{0}_{3\times3}$ means a matrix filled with zeros with a $3 \times 3$ dimension.

In order to relate the derivate of the generalized position in the Earth frame with the generalized velocity on the body frame, the transfer matrix (7) can be used, and that is the final model of the quadrotor's kinematics [1], [2].

### C. Dynamics

The dynamic model for the UAV relates the acceleration of the vehicle with the forces and torques acting on the quadrotor. The Newton-Euler formulation allows to express the variables in the body frame, as in equations (9) and (10), as clearly stated by Bresciani in [2].

$$\boldsymbol{F}^B = m(\dot{\boldsymbol{v}}^B + \boldsymbol{\omega}^B \times \boldsymbol{v}^B) \quad (9)$$

$$\boldsymbol{\tau}^B = \boldsymbol{I}\ \dot{\boldsymbol{\omega}}^B + \boldsymbol{\omega}^B \times (\boldsymbol{I}\ \boldsymbol{\omega}^B) \quad (10)$$

## III. Quadcopter control

The individual UAV systems in this project could be described with a block diagram as the one pictured in figure (3) To check that our quadrotor system is controllable and

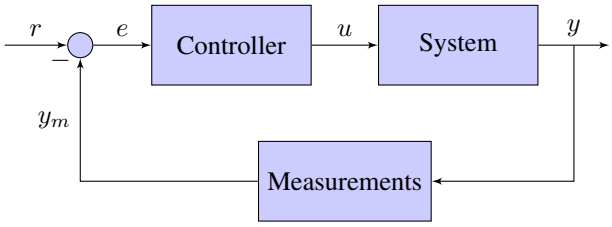


Fig. 3. block diagram for individual quadrotor

observable we linearise the mathematical model of the quadrotor according to [1] and then rewrite it in the steady state form[SOURCE] of our model seen in equation (11) and (12)

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} \quad (11)$$

$$\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} \quad (12)$$

Since all the states of the system is both observable and controlable The controller used in this project is an LQ-controller. The controller is used to check that our mathematical model of the UAV can be controlled well enough to do steady flight operations. An LQR utilizes a cost function and minimises said cost function to optimise the controller [3]. To design the LQR we use some of the constants in the steady state equation to express the Q matrix in the following way.

$$\boldsymbol{Q} = \boldsymbol{C}^T\boldsymbol{C}\alpha_1 \quad (13)$$

where $\alpha_1$ is a scaling factor for our $\boldsymbol{Q}$ matrix. the Q matrix is then implemented into the following cost function J.

$$J = \int_0^\infty (\boldsymbol{x}^T(t)\boldsymbol{Q}\boldsymbol{x}(t) + \boldsymbol{u}^2(t))dt \tag{14}$$

J is then minimised to obtain the optimal LQR with the given $\alpha_1$. The reason behind using a LQR is that it usually has small steady-state errors as stated in [4]. Since the main goal of this project is to avoid colliding any of our UAVs it would be reasonable to focus on getting the deviations from the optimal track to be as small as possible, hence minimising the error.

## IV. Measuring the UAV state and environment

The core of this project focuses on how to make aerial vehicles fly autonomously from an initial position to a goal. Therefore, apart from the main algorithm that makes this possible, it is really important that the vehicle can acquire precise information about its condition and its surroundings. Sensors do not only have to provide information about the state of the UAV as to close the loop for the controller, but also about the objects the vehicle may encounter throughput its path, as to make the navigation safe and prevent and block possible crashes.

### A. Inertial Measurement Unit

This module is in charge of measuring almost all the variables related to the movement of the vehicle. Usually inside this module a 3-axis accelerometer, a 3-axis gyroscope and a 3-axis magnetometer can be found. The most affordable ones are those that cointain simply this three sensors integrated in the same circuit board by the manufacturer. The most expensive and precise IMUs integrate specially designed sensors and sometimes include a GPS, a RS232 transceiver and a processor, that runs a real-time Kalman filter in order to provide the most accurate data directly to the CPU.

*1) Triple axis accelerometer:* This sensor measures proper acceleration along the three axes on the body frame if the accelerometer's axes match these. It can measure dynamic acceleration as a result of the motion of the drone. As shown in (15), the rotation matrix is used to change from acceleration provided by the IMU to the acceleration in the Earth frame [5].

$$\boldsymbol{a}_{\mathrm{IMU}} = \boldsymbol{R}_\Theta^T\ (\ddot{\boldsymbol{x}}^E - g\vec{\boldsymbol{z}}) \tag{15}$$

*2) Triple axis gyroscope:* This device can measure angular rates in its three axes. Therefore, it gives the angular velocity of the body frame relative to the Earth frame, expressed in the body frame (16).

$$\boldsymbol{\omega}_{\mathrm{IMU}} = \boldsymbol{\omega}^B \tag{16}$$

*3) Triple axis magnetometer:* This kind of sensors are able to measure the ambient magnetic field. Ideally this corresponds to the Earth's magnetic field, therefore the orientation of the vehicle can be measured (17).

$$\boldsymbol{m}_{\mathrm{IMU}} = \boldsymbol{R}_\Theta^T\ \boldsymbol{m}_{\mathrm{Earth}} \tag{17}$$

Where $\boldsymbol{m}_{\mathrm{Earth}}$ corresponds to the Earth's magnetic field expressed in the Earth frame. This measure can be accurate if the bias caused by the local magnetic disturbance $\boldsymbol{b}_{\mathrm{m}}$ is taken into account (18) and the sensor is places as far as possible from the elements that may cause this disturbance onboard the UAV, such as the wires that power the rotors [5].

$$\boldsymbol{m}_{\mathrm{IMU}} = \boldsymbol{R}_\Theta^T\ \boldsymbol{m}_{\mathrm{Earth}} + \boldsymbol{b}_{\mathrm{m}} \tag{18}$$

In all the specified sensors, bias and noise are also present. Gyroscopes are usually robust against this noise. But one placed in an UAV, accelerometers are affected by the vibration, and need filtering for its masurements to be considered reliable.

### B. GPS receiver

This device is basically a receiver that makes use of the satellite-based Global Positioning System to calculate the vehicle's geographical position (longitude and latitude) thanks to a 24 satellite constellation around Earth and with trilateration. Casual and inexpensive GPS devices have some meters of accuracy, therefore either better GPS devices or suplementary information from other sensors are needed in order to estimate the position of the vehicle as accurately as possible. For example, motion tracking via smart cameras together with Simultaneous Localization and Mapping solver techniques [5]. GPS may also not function indoors, so its usefulness is limited.

### C. Infrared sensors

In order to sense the UAV's immediate surroundings, a device that is able to know if there is any obstacle around and its relative position to the drone is needed. An array of active infrared sensors correctly placed on the quadrotor is a good solution for this application. An IR sensor consists basically in a LED acting as a emitter and a photodetector acting as a receiver. Both need to have a peak in the same wavelength for optimal power radiation in the emitter and sensitivity in the receiver. The LED emits a light beam in the infrared range ($700\,\mathrm{nm}$ to $1\,\mathrm{mm}$ wavelength), and when the beam finds an obstacle, it is reflected. The receptor is a Position-Sensitive Device that is able to detect the angle of the received beam, and thefefore the device is able to detect the distance to the obstacle thanks to triangulation, as can be seen in Fig.4. Since the light beam needs to be reflected by

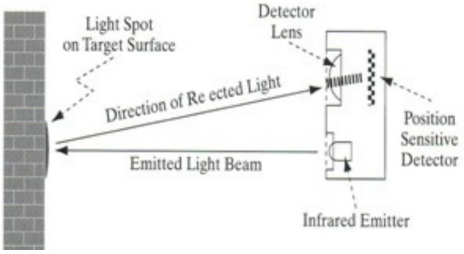


Fig. 4. Infrared obstacle detection diagram

an object, its reflectance is an important factor to take into account, since poor reflective objects could not be detected on

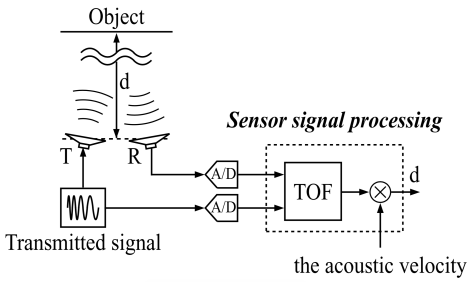


Fig. 5. Ultrasonic obstacle detection diagram, obtained from [7]

time. Also, some other natural or artificial sources of radiation such as the Sun may cause interferences. To improve the circuit's response to interferences the signal must be properly conditioned and modulated [2], [6].

### D. Ultrasonic sensors

This devices, together with the IR sensors, allow measuring the distance from the vehicle to an obstacle. An ultrasonic sensor consists of a high-frequency sound emitter and a receiver. Both are electrical signals – sound wave transducers, and their operation is similar to the IR sensors: the emitted wave is reflected by the obstacle, and when received, the distance to the obstacle can be calculated based on the time of flight (TOF) of the signal in the air, as can be seen in Fig.5. Since the velocity of the sound in the air is known, just by knowing the time that passed between emission and reception, the distance to the obstacle can be known, according to (19).

$$d_{\text{obstacle}} = \text{v}_{\text{sound,air}} \ \frac{TOF}{2} \tag{19}$$

This sensors may also be used to measure the height of the UAV, that can be combined with a barometer to know both the relative and absolute altitude. According to Adarsh in [8], both IR and ultrasonic sensors have usually high correlation between the measured values, except for some specific materials. Both types have proven to be accurate when performing further processing techniques of the acquired data.

## V. Quadcopter navigation

There are several methods to find the optimal path to follow from an initial position to a goal. In the case that is being studied in this paper, there are many UAVs flying towards their goals, in an environment filled with obstacles. Some proposed methods to solve this problem are Rapidly-exploring Random Tree (RRT) algorithms or A* search algorithms and other Dijkstra extensions. Nonetheless, these techniques do not perform very well when the test environment is constantly changing. These methods excel at finding the appropriate path to follow in a labyrinth-like environment. However, if the environment is different in every iteration, these algorithms are inefficient since the UAV may end up following an inefficient path [9], [10].

We choose to use the potential fields method because it is a simple, movement-efficient algorithm that can navigate the UAV to its goal using a short path, while being computationally fast. With this technique a high computational speed and an optimised path can be achieved, with simple and elegant calculations[11]. The main objective is to test how a lot of UAVs would behave in an open, filled with obstacles environment and if the potential field method can be considered good enough.

Potential fields is the sum of an attractive and a repulsive potential. In this case the goal generates the attractive potential and the obstacles generate repulsive potentials. This will determine the movement of the UAV by following the negative gradient of the potential field. The repulsive and the attractive potentials are calculated according to [12], as can be seen in equations (20) and (21) respectively.

$$U_{\text{att}}(\boldsymbol{q}) = \begin{cases} \frac{1}{2}\xi\rho^2_{\text{goal}}(\boldsymbol{q}) & \text{if } \rho_{\text{goal}}(\boldsymbol{q}) \leq d \\ d\xi\rho_{\text{goal}}(\boldsymbol{q}) & \text{if } \rho_{\text{goal}}(\boldsymbol{q}) > d \end{cases} \tag{20}$$

$$U_{\text{rep}}(\boldsymbol{q}) = \begin{cases} \frac{1}{2}\eta(\frac{1}{\rho_{\text{obst}}(\boldsymbol{q})} - \frac{1}{\rho_0}) & \text{if } \rho_{\text{obst}}(\boldsymbol{q}) \leq \rho_0 \\ 0 & \text{if } \rho_{\text{obst}}(\boldsymbol{q}) > \rho_0 \end{cases} \tag{21}$$

Where $\boldsymbol{q} = \begin{bmatrix} x & y & z \end{bmatrix}^T$ is the UAV's linear position in the space for the 3D case. The variables $\rho_{\text{goal}}(\boldsymbol{q})$ and $\rho_{\text{obst}}(\boldsymbol{q})$ are the distances between the vehicle and the goal or the nearest obstacle, calculated with the Euclidean norm $\rho_{\text{x}}(\boldsymbol{q}) = \| \boldsymbol{q} - \boldsymbol{q_{\text{x}}} \|$. The $d$ parameter is the distance from the goal where the attractive function changes from a conic to a parabolic well, because the combination of both configurations resolves the problems each one has separately. The constant $\rho_0$ is the sensing radius of the UAV. Finally, $\xi$ and $\eta$ are positive scaling factors.

The potential fields' desired force on the UAV is calculated with (22), since the negative gradient of a potential field $-\boldsymbol{\nabla}U$ is a vector that points in the direction of steepest descent.

$$\vec{F}(\boldsymbol{q}) = -\boldsymbol{\nabla}U(\boldsymbol{q}) \tag{22}$$

This leads to (23) and (24), that show how these repulsive and attractive forces are calculated.

$$\vec{F}_{\text{att}}(\boldsymbol{q}) = \begin{cases} -\xi(\boldsymbol{q} - \boldsymbol{q_{\text{goal}}}) & \text{if } \rho_{\text{goal}}(\boldsymbol{q}) \leq d \\ -d\xi\frac{(\boldsymbol{q} - \boldsymbol{q_{\text{goal}}})}{\|\boldsymbol{q} - \boldsymbol{q_{\text{goal}}}\|} & \text{if } \rho_{\text{goal}}(\boldsymbol{q}) > d \end{cases} \tag{23}$$

$$\vec{F}_{\text{rep}}(\boldsymbol{q}) = \begin{cases} \eta(\frac{1}{\rho_{\text{obst}}(\boldsymbol{q})} - \frac{1}{\rho_0})(\frac{1}{\rho^2_{\text{obst}}(\boldsymbol{q})})\frac{(\boldsymbol{q} - \boldsymbol{q_{\text{obst}}})}{\|\boldsymbol{q} - \boldsymbol{q_{\text{obst}}}\|} & \text{if } \rho_{\text{obst}}(\boldsymbol{q}) \leq \rho_0 \\ 0 & \text{if } \rho_{\text{obst}}(\boldsymbol{q}) > \rho_0 \end{cases} \tag{24}$$

As mentioned before, there are two reasons for splitting the attractive potential and force functions into two parts. In the case of a parabolic configuration for all the space, there is a linear dependence of the force with the distance between UAV and goal, hence the force grows indefinitely

when far from the goal. At the same time, in the case of a conic configuration for all the space, there is a singular point in the goal that could make the UAV oscilate around it. The solution is to combine both configurations to solve this problems, by making the force constant when far from the goal with a conic configuration, and then using a parabolic function when reaching the goal to avoid the singularity.

Finally, the repulsive and attractive potentials and forces are added, according to (25) and (26), in order to obtain the total value.

$$U_{\text{total}}(\boldsymbol{q}) = U_{\text{att}}(\boldsymbol{q}) + U_{\text{rep}}(\boldsymbol{q}) \tag{25}$$

$$\vec{F}_{\text{total}}(\boldsymbol{q}) = \vec{F}_{\text{att}}(\boldsymbol{q}) + \vec{F}_{\text{rep}}(\boldsymbol{q}) \tag{26}$$

To simplify the simulations, first order dynamics are assumed for the quadcopter model, resulting in the control signal $\boldsymbol{u}(t)$ for the quad being directly the velocity obtained, that would equal the negative gradient of the potential, as can be seen in (27).

$$\boldsymbol{u}(t) = \dot{\boldsymbol{p}}(t) = -\boldsymbol{\nabla} U(\boldsymbol{p}(t)) \tag{27}$$

This method, nevertheless, has some clear limitations. The most important problem to solve is the local minima situation, where a vehicle can get stuck and therefore never arriving to its goal. But there are some other problems, like the difficulty of passing between closely-spaced obstacles or inherent oscilations in the trajectory when near obstacles, as stated by Koren and Borenstein in [13].

## VI. Simulation

All the necessary formulas described in this report have been implemented in MatLab to test their validity in the context. The parts we have focused more on have been the navigation with potential fields and the control.

For the navigation we developed a script to test the guidance of our potential field function by simulating drones and obstacles. All the objects are modelled as spheres, so they have a position, that corresponds to the center of the sphere, and a radius. Drones have a fixed radius, and the script generates them in a random initial position. Each of them has also a specific goal. In the same space, obstacles are generated so that they do not overlap with initial positions or goals. Once the environment is set, the script calculates for each time step the potential function of each UAV,

## VII. Results

It's bloody impossible to do this.

### A. Control

## VIII. Discussion, conclusion and Future development

future projects could try to actually control the simulated drones with the controller we designed to see if it will actually provide a safe control of the drones. Another future project could be to focus on the difference between different navigation algorithms, i.e. Djikstra compared to A*, potential fields and so fort. another interesting application is to actually try to implement the navigation system into a real quadrocopter and see how well it functions.

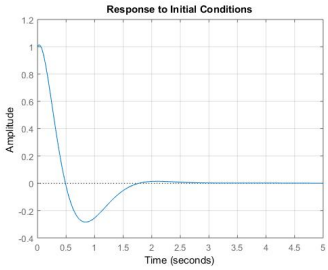


Fig. 6. Statefunction for input 1

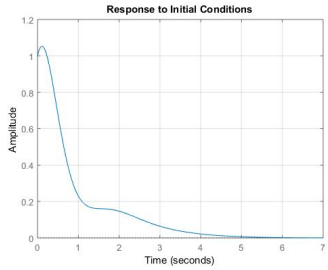


Fig. 7. Statefunction for input 12

## Appendix A
## Dynamics or something that's heavy on the maths goes here

Appendix one text goes here.

$$\dot{\boldsymbol{\xi}} = \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \\ \dot{p} \\ \dot{q} \\ \dot{r} \\ \dot{u} \\ \dot{v} \\ \dot{w} \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \tag{28}$$

## Appendix B
## Something else but similar here, matlab code?

Appendix two text goes here.

## Acknowledgment

Praising of Christos goes here! The authors would like to thank... TEST

## REFERENCES


[1] F. Sabatino and K. H. Johansson, "Quadrotor control: modeling, nonlinearcontrol design, and simulation," KTH, Skolan fr elektro- och systemteknik (EES), Reglerteknik, 2015.

[2] T. Bresciani, "Modelling, identification and control of a quadrotor helicopter," 2008, student Paper.

[3] T. Glad, *Reglerteknik : grundlggande teori*, 4th ed. Lund: Studentlitteratur, 2006, pp. 187–188.

[4] L. M. Argentim, W. C. Rezende, P. E. Santos, and R. A. Aguiar, "Pid, lqr and lqr-pid on a quadcopter platform," in *2013 International Conference on Informatics, Electronics and Vision (ICIEV)*, May 2013, pp. 1–6.

[5] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE Robotics Automation Magazine*, vol. 19, no. 3, pp. 20–32, Sept 2012.

[6] J. A. Chavez and S. Silvestre, "Infrared remote control systems," University lecture, UPC, Electronic Engineering Department, 2017.

[7] S. Hirata, M. K. Kurosawa, and T. Katagiri, "Cross-correlation by single-bit signal processing for ultrasonic distance measurement," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 91, no. 4, pp. 1031–1037, 2008.

[8] S. Adarsh, S. M. Kaleemuddin, D. Bose, and K. I. Ramachandran, "Performance comparison of infrared and ultrasonic sensors for obstacles of different materials in vehicle/ robot navigation applications," *IOP Conference Series: Materials Science and Engineering*, vol. 149, no. 1, September 2016.

[9] S. M. Lavalle, "Rapidly-exploring random trees: A new tool for path planning," 05 1999.

[10] D. S. Yershov and S. M. LaValle, "Simplicial dijkstra and a* algorithms for optimal feedback planning," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sept 2011, pp. 3862–3867.

[11] S. Ge and Y. Cui, "Dynamic motion planning for mobile robots using potential field method," *Autonomous Robots*, vol. 13, no. 3, pp. 207–222, Nov 2002. [Online]. Available: https://doi.org/10.1023/A:1020564024509

[12] N. Amato, "Potential field methods," University lecture, Università degli Studi di Padova, 2004.

[13] Y. Koren and J. Borenstein, "Potential field methods and their inherent limitations for mobile robot navigation," in *Proceedings. 1991 IEEE International Conference on Robotics and Automation*, Apr 1991, pp. 1398–1404 vol.2.