

Control and Estimation Methods for Unknown Load Transportation with Quadrotors

Pedro Outeiro
pedro.outeiro@tecnico.ulisboa.pt

Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal

June 2018

Abstract

This thesis presents methodologies for height control of a quadrotor that transports a piecewise constant unknown load, given the estimates on both weight and state variables, based on measurements from motion sensors installed on board. The control and estimation solutions were required to guarantee null steady state position error and provide vertical velocity estimates, given that there is no velocity sensor on board the quadrotor.

The approaches considered encompass linear and non-linear methods. They include an algebraic solution, classical control, optimal control and estimation, integrative action, multi-model methods, sliding mode control, and adaptive control techniques. All methods were tested in simulation and the three most promising approaches were selected for proposal.

The first proposed solution has as controller the sub-optimal steady state solution of a Linear Quadratic Regulator (LQR) problem, where an integrative effect is added. The in-flight estimation problem is tackled resorting to a multiple-model adaptive estimator. The second solution is a Multi-Model Adaptive Controller, using LQR with integrative action, and Kalman filter with integrative component. The last solution incorporates an adaptive controller, scheduling LQR gains and gravitational force compensation. The in-flight estimation problem is tackled resorting to a multiple-model adaptive estimator. The control systems obtained were validated resorting to an off-the-shelf commercially available quadrotor, equipped with an Inertial Measurement Unit (IMU), an ultrasound height sensor, and a barometer, among other sensors.

Keywords: Quadrotor, Height Control, Estimation, Unknown Load

1. Introduction

With the rise of self-driving cars and autonomous vehicles, one of their future uses could be to make deliveries. With high pedestrian and automotive traffic, deliveries can be delayed, but an aerial vehicle has no such inhibition. Nowadays, low altitude flight could be achieved without high traffic concerns. Additionally, the UAV of choice for these delivery systems has been multi-rotors. These vehicles have high maneuverability and have hovering capabilities, which makes them ideal for operation in environments with multiple obstacles, such as urban environments.

One problem that relates to delivery systems is the control of a quadrotor transporting a load of unknown mass. A delivery system should be capable of transporting their cargo regardless of its mass (within reason). The dynamics of the system would change based on the mass and a fixed control might not be capable of transporting loads with a mass too different from the nominal mass considered. To achieve a versatile solution, the quadrotor should

be capable of compensating, and/or estimating the unknown mass of the load, making the necessary adjustments during flight.

This is a summarized version of the thesis and only contains details on the three solutions that were selected for experimental testing.

1.1. Topic Overview

$$M\ddot{z} = f(\mathbf{u}, \dot{z}, g) \quad (1)$$

The height dynamics of a quadrotor are shown in equation 1, where z is the height of the quadrotor, M is its mass, \mathbf{u} is the thrust and g is the acceleration of gravity, assumed constant at the mission environment. Although it is a non-linear equation, control solutions with an LQR controller and a constant compensation of the gravitational component can be used. However, this is not as simple for the problem of piecewise constant unknown load mass (m) transportation. Re-writing the equation, results in

$$(M + m)\ddot{z} = f(\mathbf{u}, \dot{z}, g, m) \quad (2)$$

In this case, the solution for the control is not as immediate, the performance degrades, and the

platform stability is compromised. The gravitational effect influenced by m is, therefore, unknown. Only a lower bound for the gravity effect can be known *a priori*. Additionally, the $(M + m)\ddot{z}$ component presents an added non-linearity to the problem. Since linearization of this equation would limit a solution to only work for a specific mass and possibly a small range of masses, the use of standard linear solutions is out of question and alternative solutions should be considered.

Given the non-linearity of the dynamics, the estimation problem is harder. Additionally, the available sensors of the quadrotor do not provide a measurement of the z velocity or, if estimated based on optical flow techniques, has poor quality. To tackle the optimal control problem, the velocity is needed and, in the absence of sensory data, an estimate is required.

1.2. Objectives

In this work, it is intended to provide a solution for controlling the height of a quadrotor carrying a piecewise constant unknown load. Solutions for this problem can be a controller which is capable of handling the height control without knowing the mass, or devise a way of estimating the additional mass and have the controller adapt accordingly. Both of these possibilities will be explored.

Additionally, estimation techniques for filtering and fusing sensory data will be studied, as these go hand in hand with many control techniques and may also be capable of performing identification.

1.3. State of the Art

There is scarce control system design work made for unknown loads. One such example can be found in [1]. On the other hand, known load cases have been explored extensively. In [2] a Mixed Integer Quadratic Program was designed for the control of a quadrotor with a load suspended by a cable for aggressive maneuvering. In [3] the same problem is studied, but using a backstepping controller. Another approach to suspended load transportation is to use multiple UAVs, like the one shown in [4]. Here, a cooperative control scheme where the quadrotors simultaneously carry the load is proposed.

1.4. Extended Abstract Outline

This extended abstract is organized as follows: The details of the model, physical representation, and software used are discussed in section 2. The mathematical definition of the estimation methods is presented in section 3. The mathematical definition of the control methods is presented in section 4. In section 5 the simulation results for the discussed approaches are presented and analyzed. The experimental results are presented and analyzed in section 6. Finally, in section 7 some concluding remarks are

drawn and possible future work is suggested.

2. Aerial Vehicle

The quadrotor model used is the Parrot Ar.Drone 2.0. The drone is equipped with four brushless motors. These actuators are controlled through the use of PWM commands ranging from 0 to 100.

This quadrotor is equipped with an inboard IMU (Inertial Measurement Unit), and front and ground cameras, the later being an optical flow camera. The inboard IMU is equipped with an accelerometer, a gyroscope, a magnetometer, an ultrasound sensor, and a barometer. With these sensors it is capable of measuring its height, angles, acceleration, horizontal velocities and angular velocities.

The physical model of a quadrotor is described in [5]. Since the purpose of this thesis is to control the height, most work will be performed using the isolated z component of the dynamics. Small angles will be assumed, neglecting the effect of the rotation on the z dynamics. Additionally, a linear drag effect will be considered. Therefore, the simplified model with the added load mass is the one presented in (3), where M is the mass of the drone, m is the mass of the load, a is actuation factor, g is gravity, \mathbf{u} is the thrust generated by the rotors, γ is the drag coefficient, and \ddot{z} is the height acceleration.

$$(M + m)\ddot{z} = a\mathbf{u} - (M + m)g - \gamma\dot{z} \quad (3)$$

Since most of the proposed techniques rely on optimal control methods, a state-space representation is required. Taking into account that for the height dynamics the available sensors are the ultrasound (or barometer) and the accelerometer, the resulting state-space representation is as presented in (4):

$$\begin{aligned} \begin{bmatrix} \dot{z} \\ z \end{bmatrix} &= \underbrace{\begin{bmatrix} -\frac{\gamma}{M+m} & 0 \\ 1 & 0 \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} \dot{z} \\ z \end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix} \frac{a}{M+m} \\ 0 \end{bmatrix}}_{\mathbf{B}} \underbrace{\left(\mathbf{u} - \frac{(M+m)g}{a} \right)}_{\mathbf{U}} \\ \begin{bmatrix} \ddot{z} \\ \dot{z} \end{bmatrix} &= \underbrace{\begin{bmatrix} -\frac{\gamma}{M+m} & 0 \\ 0 & 1 \end{bmatrix}}_{\mathbf{C}} \begin{bmatrix} \dot{z} \\ z \end{bmatrix} + \underbrace{\begin{bmatrix} \frac{a}{M+m} \\ 0 \end{bmatrix}}_{\mathbf{D}} \mathbf{U} \end{aligned} \quad (4)$$

Finally, values for the relevant parameters are presented in Table 1 and were taken from the drone specifications and from [6].

Table 1: Drone Parameters.

Parameter	Value
a	1
M (Kg)	0.420
γ	0.1

3. Estimation

In this section, the estimation methods are presented and detailed.

3.1. Multiple-Model Adaptive Estimation

The Multiple-Model Adaptive Estimation (MMAE) algorithm [7] is a combined state-estimation and system identification method. Its uses include pro-

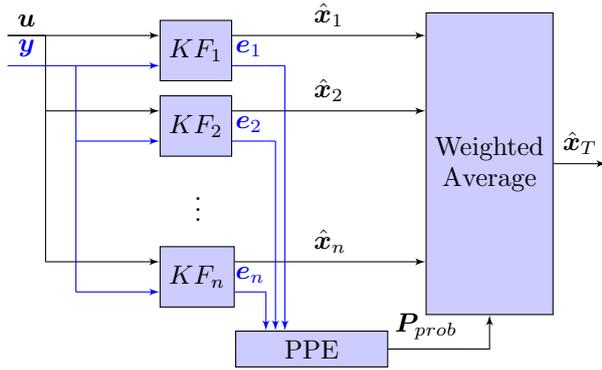


Figure 1: MMAE Structure.

viding a solution for parametric uncertainties and for non-linear state-estimation (using different linearizations). As the name implies, it relies on multiple models for the same system, but assuming different values of the parameter (or linearization points). For each of these underlying models a Kalman filter is designed, providing accurate optimal estimates for its assumed model. The merging and processing of the information provided by the bank of filters is computed resorting to the Bayesian Posterior Probability Estimator (PPE) that selects the most accurate filter.

The PPE assesses the accuracy through the residues of the known sensory data, by assigning a probability to each filter. The initial value of these probabilities are known as the *a priori* probabilities and are commonly initialized equal for the n filters ($1/n$), unless there is *a priori* knowledge to support giving a higher or lower probability at start. Following values are called the posterior probabilities $\mathbf{P}_{prob_k}(t+1)$, calculated iteratively using the past probabilities $\mathbf{P}_{prob_k}(t)$ and residues of the filters e_i according to (5-7) [8] (h represents the number of sensors used). The residual covariance matrix of each filter (\mathbf{S}_i) is also used as a weighting parameter in the calculations. $\beta_i(t)$ is a weighting parameter based on the residual covariance and number of sensors, and $w_i(t)$ is a quadratic weighting parameter for the residue which also uses the residual covariance.

$$\mathbf{P}_{prob_k}(t+1) = \frac{\beta_k(t+1)e^{-\frac{1}{2}w_k(t+1)}}{\sum_{j=1}^n \beta_j(t+1)e^{-\frac{1}{2}w_j(t+1)}} \mathbf{P}_{prob_k}(t) \quad (5)$$

$$\beta_i(t+1) = \frac{1}{(2\pi)^{\frac{h}{2}} \sqrt{\det \mathbf{S}_i(t+1)}} \quad (6)$$

$$w_i(t+1) = \mathbf{r}'(t+1) \mathbf{S}_i^{-1}(t+1) \mathbf{r}(t+1) \quad (7)$$

The state estimation of the MMAE algorithm is obtained with a weighted average (see [7]) using the posterior probabilities as a weighting factor, as shown in (8), providing low pass filtered state estimates.

$$\hat{\mathbf{x}}_T = \sum_{j=1}^n \mathbf{P}_{prob_j}(t) \hat{\mathbf{x}}_j \quad (8)$$

The resulting structure is shown in Fig. 1.

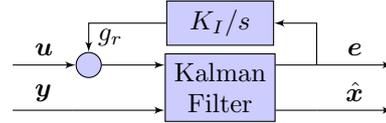


Figure 2: Kalman Filter with Integrative Component.

3.2. Kalman Filter with Integrative Component

Since the gravitational force will only match the real value when there is a corresponding model, an additional mechanism for handling the unknown gravitational force is necessary to account for the error of the assumed gravitational force in the filters.

Since the residue (e) is a must for the MMAE, it is possible to use it to adjust the gravitational force (g_r). By creating a feedback loop to the actuation input (u) with an integrator, it allows for the height estimate to follow the height measurement closely and provide a more accurate estimate of the velocity. For tuning purposes a gain can be given to the integration, allowing to adjust the overshoot and speed of the estimate. The resulting structure resembles Fig. 2.

Since this method solves the gravitational force issue, it could be considered sufficient to use a single model approach. However, even if disregarding the gravitational force, the mass still has weight on the dynamics of the quadrotor and the larger the difference in the assumed mass of the filter, the higher the error of the velocity estimate. Therefore, the use of the MMAE algorithm is still beneficial.

3.3. Filter Design

Using the proposed state-space representation, the Kalman gains L are obtained and are combined with the integrative gain for the residue of the height to provide the filter presented in (9) and (10).

$$\begin{bmatrix} \ddot{z} \\ \dot{z} \\ \dot{g}_r \end{bmatrix} = \begin{bmatrix} \mathbf{A} - \mathbf{L}\mathbf{C} & \mathbf{B} - \mathbf{L}\mathbf{D} \\ [0 & -\mathbf{K}_I] & 0 \end{bmatrix} \begin{bmatrix} \dot{z} \\ z \\ g_r \end{bmatrix} + \begin{bmatrix} \mathbf{B} - \mathbf{L}\mathbf{D} & \mathbf{L} \\ 0 & [\mathbf{K}_I] \end{bmatrix} \begin{bmatrix} U \\ \mathbf{s} \end{bmatrix} \quad (9)$$

$$\begin{bmatrix} e_z \\ e_z \end{bmatrix} = \begin{bmatrix} [1 & 0](-\mathbf{A} + \mathbf{L}\mathbf{C}) & [1 & 0](-\mathbf{B} + \mathbf{L}\mathbf{D}) \\ [0 & -1] & 0 \end{bmatrix} \begin{bmatrix} \dot{z} \\ z \\ g_r \end{bmatrix} + \begin{bmatrix} [1 & 0](-\mathbf{B} + \mathbf{L}\mathbf{D}) & [1 & 0](-\mathbf{L} + \mathbf{I}) \\ 0 & [0 & 1] \end{bmatrix} \begin{bmatrix} U \\ \mathbf{s} \end{bmatrix} \quad (10)$$

4. Control

In this section, the control methods are presented and detailed.

4.1. LQR with Integrative Action

To account for the unknown gravitational force, methods using integration were considered. The LQR controller with integrative action is a variation consisting of a cascading controller with an inner feedback of all the state-variables and an outer layer that integrates the difference between reference and current value of the control variable, which is equivalent to the structure shown in Fig. 3.

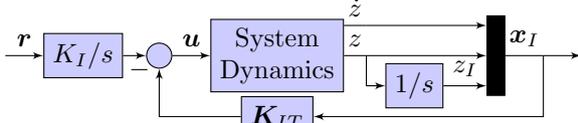


Figure 3: LQR with Integrative Component.

To obtain a controller like this using LQR control it is only necessary to modify the model of the dynamics when calculating the LQR gains. By using the modified version of the model presented in (11), there will be a state-variable associated with the integration that will be used for defining the integrative control gain.

$$\mathbf{A}_I = \begin{bmatrix} \mathbf{A} & 0 \\ -\mathbf{C} & 0 \end{bmatrix} \quad \mathbf{B}_I = \begin{bmatrix} \mathbf{B} \\ 0 \end{bmatrix} \quad (11)$$

Having a modified model, the next step is very straightforward, just calculate the LQR gains for the new model. Finally, from the resulting gains we obtain two different sets. \mathbf{K} is the vector of gains for the state variables and K_I is the gain for the integrative component, selected according to \mathbf{K}_{calc} :

$$\mathbf{K}_{calc} = [\mathbf{K} \mid -K_I] \quad \mathbf{K}_{IT} = [\mathbf{K} \quad K_I] \\ \mathbf{K} = [K_v \quad K_h]$$

Since this is a methodology for linear systems, a restructuring of the dynamics is required. Therefore, the restructuring used in Section 3.2 will also be used on the design of the controller. As mentioned in section 1.1, feedback linearization is a possible solution for the known load case, but it can still be used with an unknown load. Using a feedback linearization of the mass of the quadrotor lessens the work required by the LQR controller with integrative action and results in a faster control. The controller structure used is, therefore, presented in Fig. 4.

4.2. Multi-Model Adaptive Control

The Multiple-Model Adaptive Control (MMAC) algorithm [7] is a combined state-estimation, control and system identification method. It uses include providing a solution for parametric uncertainties and for non-linear control (using different linearizations). It performs the necessary estimation and control for different models of the system in study. It uses Multiple models and the PPE like the MMAE algorithm, but it replaces the bank of Kalman filters with a bank of combined Kalman filters and LQR controllers.

The total actuation of the MMAC can be ob-

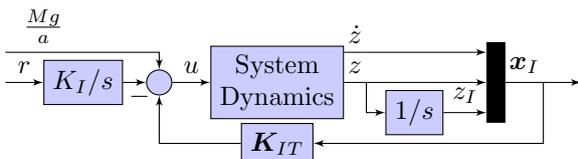


Figure 4: LQR with Integrative Action and Quadrotor Mass compensation.

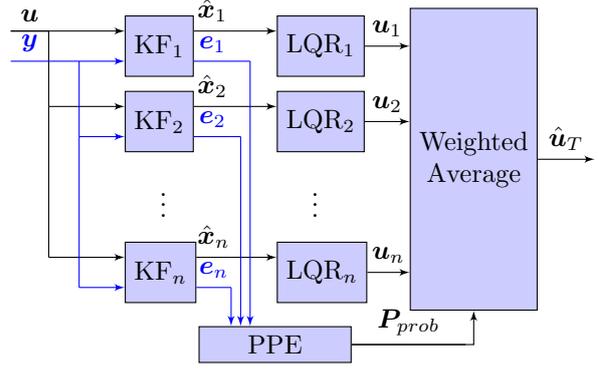


Figure 5: MMAC Structure.

tained in a similar way to the MMAE algorithm overall estimation. In this work, it was adopted the use of the average weight, as it provides low pass filtered actuation, i.e.

$$\hat{\mathbf{u}}_T = \sum_{j=1}^n P_{prob_j}(t) \mathbf{u}_j \quad (12)$$

The resulting structure is depicted in Fig. 5.

4.3. Adaptive Control

The adaptive mechanism selected is an MIT rule reference model identifier of the mass. The reference model is a second-order transfer function based on the desired height and the mechanism [9] is described by (13). Where z_p is the height measurement, z_m is the reference model height and k is an adjustment parameter.

$$\dot{\hat{m}} = k(z_m - z_p)z_m \quad (13)$$

Having an estimate of the mass that does not depend on the values of the assumed models of the MMAE, it is possible to solve the problem of the unknown gravitational force, which allows the use of a feedback linearization to remove this non-linearity from the system. In doing so, the remainder of the control will not require to account for this force, removing the need for solutions like adding integrative action.

Use of the feedback linearization solves the hover

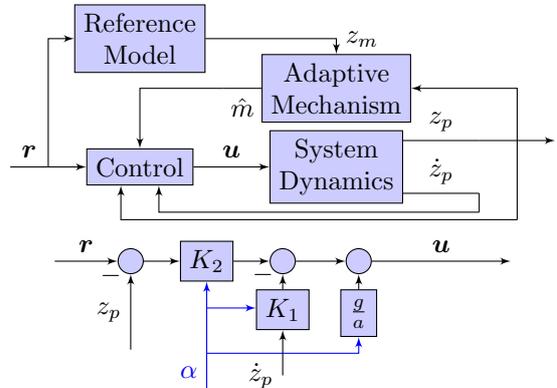


Figure 6: Top (a) - Control Structure; Bottom (b) - Controller.

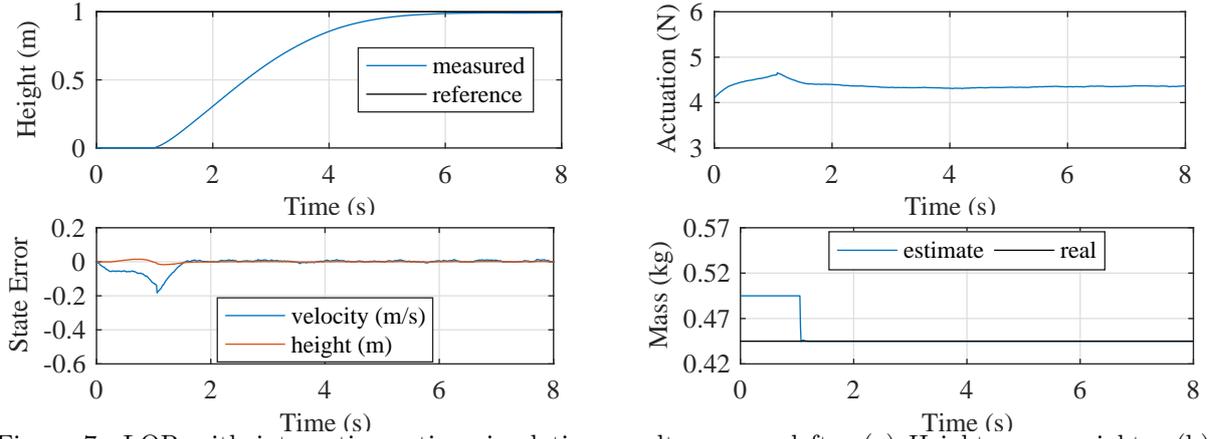


Figure 7: LQR with integrative action simulation results: upper left - (a) Height, upper right - (b) Actuation, bottom left - (c) State error, bottom right - (d) Mass estimate.

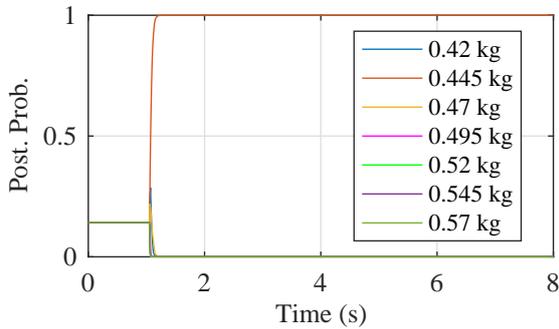


Figure 8: LQR with integrative action simulation results: Posterior probabilities.

force problem, but does not fully linearize the system. The mass of the quadrotor is still present in the remainder of the dynamics. Therefore, a static gain controller does not provide an optimal solution. Given that the adaptive mechanism provides an estimate for the mass of the quadrotor, this can be used to adjust an LQR controller according to the estimate and ensure the optimality of the controller. For this purpose, an explicit calculation of the LQR gains is necessary, that departs from the optimal steady state H2 control problem solution. The obtained steady state gains equations are presented in (14-17), where K_2 is constant.

$$\alpha = M + m \quad (14)$$

$$p_{11} = \frac{\sqrt{\alpha (r (a^2 q_1 + \gamma^2 r + 2a\alpha\sqrt{r}q_2))} - \alpha\gamma r}{a^2} \quad (15)$$

$$p_{12} = \frac{\alpha}{a} \sqrt{r} q_2 \quad (16)$$

$$\mathbf{K} = [K_1 \quad K_2] = \mathbf{R}^{-1} \mathbf{B}^T \mathbf{P} = \frac{a}{r\alpha} [p_{11} \quad p_{12}] \quad (17)$$

The structure of the full solution (a) and the controller (b) are presented in Fig. 6.

5. Simulation Results

In this section, the preparation and results of the simulation are presented. Limitations related to the

available thrust of the motor limited the range of load masses to a maximum of 0.15 kg and seven masses 0.025 kg apart were used for the multi-model algorithms.

5.1. LQR with Integrative Action with Integrative MMAE Estimation

The simulated load of this simulation was given a mass of 0.025 kg. For the purpose of the Kalman gain calculations, the covariance of the sensor noise was defined as 1 and 0.001 for the acceleration and height respectively, while the process noise was given a covariance of 0.5. The integrative gain of the filters was set as 4. The PPE was set to hold until a height of 0.01 m had been reached, which also triggered the filters' reset. The calculated LQR integrative gain was $K_I = 0.5$ and the state gains were $K = [0.8877 \quad 1.1494]$ for velocity and height respectively. A stop condition for the integrative component of the control was set for the thrusts outside a range of 3 to 5 N.

The results of the simulation are presented in Figs. 7 and 8, where it is observed that there is a 1 second lift-off period, followed immediately by the filter reset. The settling time (5%) is at 4.8 seconds and the height stabilizes around the 6 second mark. There is no overshoot and the filters settle 0.5 seconds after resetting. At the same time that the state error settles, the 0.445 kg filter is selected, providing accurate mass estimation. However, the mass estimate can have an error in cases where the real mass does not match any of the filter masses.

5.2. MMAC with Integrative Action and Integrative Kalman Filters

The simulated load of this simulation was given a mass of 0.025 kg. For the purpose of the Kalman gain calculations, the covariance of the sensor noise was defined as 1 and 0.001 for the acceleration and height respectively, while the process noise was given a covariance of 0.5. The integrative gain of the filters was set as 4. The PPE was set to hold un-

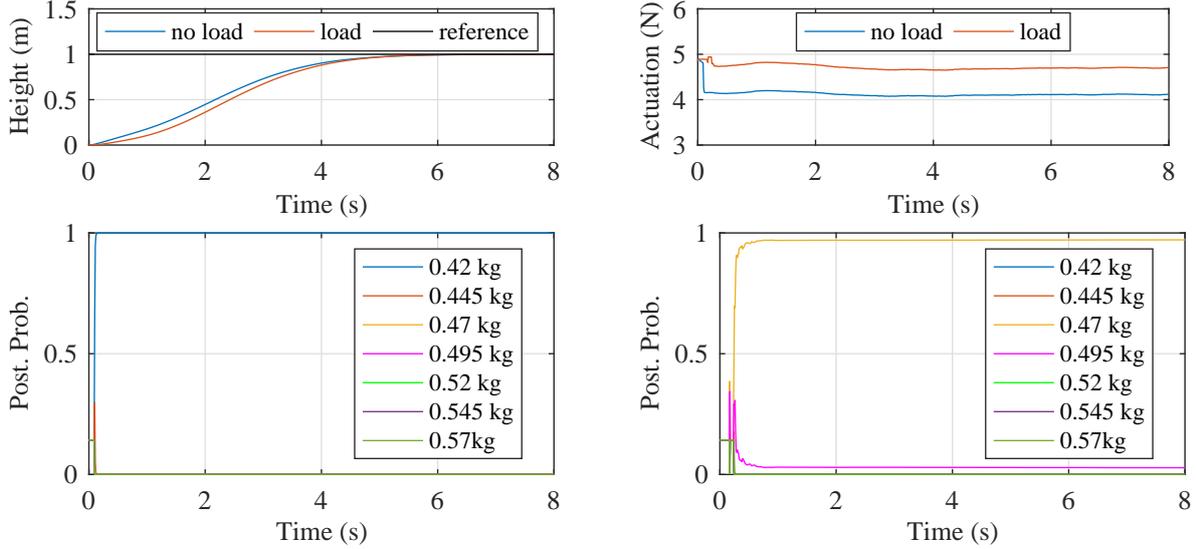


Figure 9: MMAC simulation results: upper left - (a) Height for both tests, upper right - (b) Actuation for both tests, bottom left - (c) Posterior probabilities for the no load test, bottom right - (d) Posterior probabilities for the load test.

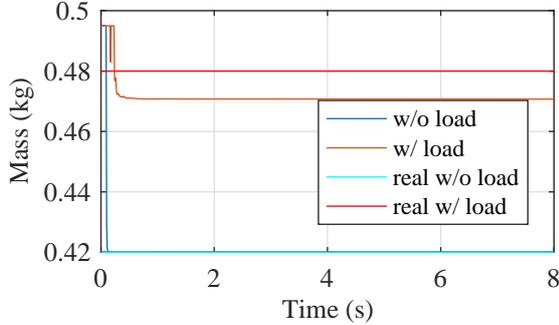


Figure 10: MMAC simulation results: Mass estimate for both tests.

til a height of 0.01 m had been reached, which also triggered the filters' reset. The LQR gains were calculated using $Q = \text{diag}([0.001 \ 5 \ 3.75])$ and $R = 15$. A stop condition for the integrative component of the control was set for the thrusts outside a range of 3 to 5 N.

The results of the simulation are presented in Figs. 9 and 10. The settling time (5%) is at 4.5 seconds and the height stabilises around the 5 second mark for both tests without overshoot. The filter selection was made in under 0.5 seconds in both tests, providing accurate mass estimation in the no load case and having a slight error in the load case, as the mass of the no load matched one of the filters and the mass of the load case did not. The MMAC did not attribute full probability to the selected filter in the load case, having a small residue for the 0.495 kg filter, resulting in a higher mass than that of the nearest filter. The actuation did not saturate and was quick to adjust to fit the case in testing, as evidenced by the quick change in actuation for the no load case. These results were promising and provided similar results independently of whether

the mass matched the assumed masses of the filters or not.

5.3. Adaptive LQR Controller with Integrative MMAE Estimation

The simulated load of this simulation was given a mass of 0.0625 kg. For the purpose of the Kalman gain calculations, the covariance of the sensor noise was defined as 1 and 0.001 for the acceleration and height respectively, while the process noise was given a covariance of 0.5. The integrative gain of the filters was set as 4. The PPE was set to hold until a height of 0.01 m had been reached, which also returns the filters to their initial condition to avoid error accumulation before lift-off. The adaptive LQR gains were calculated using a $Q = \text{diag}(2.5 \ 25)$ and $R = 10$. The adaptive mechanism gain was set to 0.07 and the reference model was a second-order transfer function with the desired height as input and parameters of $w_n = 10$ and $\zeta = 1.5$. Additionally, for low height, the reference value was changed to 0.1 to allow for a less aggressive initialization.

The results of the simulation are presented in Figs. 11 and 12. The settling time (5%) is at 4 seconds and the height stabilizes around the 4.5 second mark. There is an overshoot of 8% and the filters settle 1 second after resetting. The actuation saturates, but only during the start of flight. Immediately after lift-off the state error settles, and the 0.495 kg filter is selected. The resulting error in the mass estimate was expected, as the value of the mass doesn't match any filter. The adaptive estimate of the mass converges to the correct value and settles at 3 seconds with little overshoot.

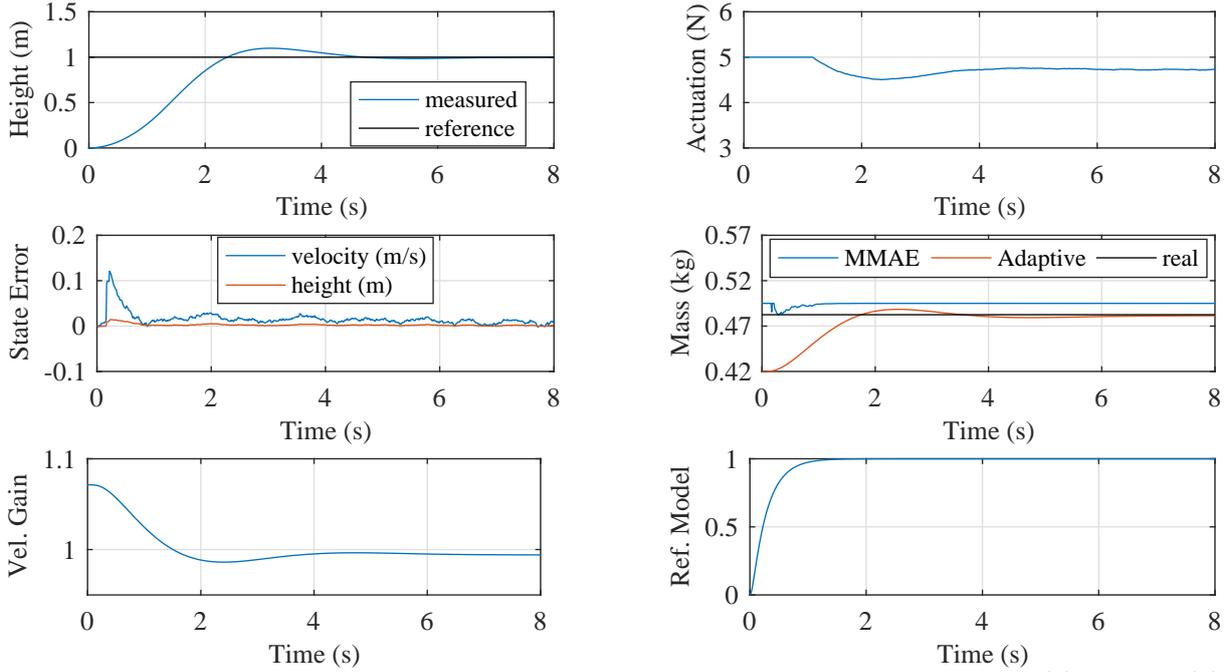


Figure 11: Adaptive control simulation results. From left to right, top to bottom: (a) Height, (b) Actuation, (c) State error, (d) Mass estimates, (e) Velocity LQR gain, (f) Reference model.

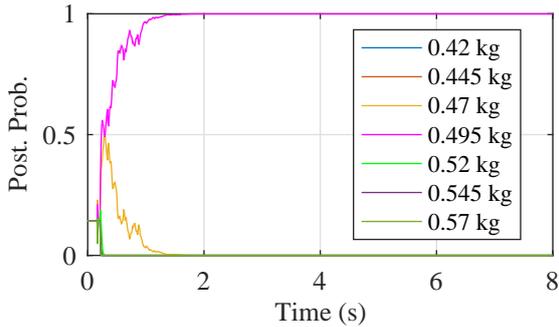


Figure 12: Adaptive control simulation results: Posterior probabilities.

6. Experimental Results

In this section, the experimental set-up is presented. Then, the implementation platform for the solutions and the additional components required for the full control of the quadrotor will be discussed. Finally, the experimental results of the proposed solution will be presented and analyzed.

6.1. Implementation

For the purpose of implementing the control, the AR.Drone 2.0 Quadcopter Embedded Coder [10] was used, as it provides a Simulink based environment for development of software to run in the quadrotor, and allows direct access to its sensors and actuators.

The control and estimation approaches were transferred onto the Embedded Coder Simulink environment. The commands for the four rotors were defined by first calculating the necessary thrust from each rotor, from the desired overall thrust and the desired torques, and converting them into their

equivalent PWM commands.

The total mass of the load used was 0.057 kg. Additionally, the mass of the quadrotor was higher than expected having an added 0.05 kg without using a load.

6.2. LQR with Iterative Action with Integrative MMAE Estimation

The results obtained with the set-up in Section 6.1 are presented in Figs. 13 and 14. The thrust gain a had to be changed to 1.025 to ensure accurate mass estimation. It is observed in Fig. 13a that the settling times (5%) are 5 and 7 seconds for the no load and with load cases respectively. The one meter height is reached at 6 and 7.5 seconds respectively. The height estimate is smoother and follows the measurement very closely for both cases, as seen in Fig. 13c and 13d. The estimated velocity is smooth despite the sensitivity of the accelerometer and seems coherent with the height data. In the initial stage of flight in Fig. 13e, the filter that was given more probability was the one with 0.495 kg, but the filter with the correct mass was selected in the end. In Fig. 13f the selection of the filter matching the closest mass was also observed and always had the highest probability. Additionally, the selection of the mass with no load settled in about 2.5 seconds, while for the load case it settled in approximately 2 second. The settling time of the probabilities is further corroborated in Fig. 14, where it can also be observed the accurate estimation of the mass in the no load test and an error of 0.007 kg in the load test, due to it not matching the filters.

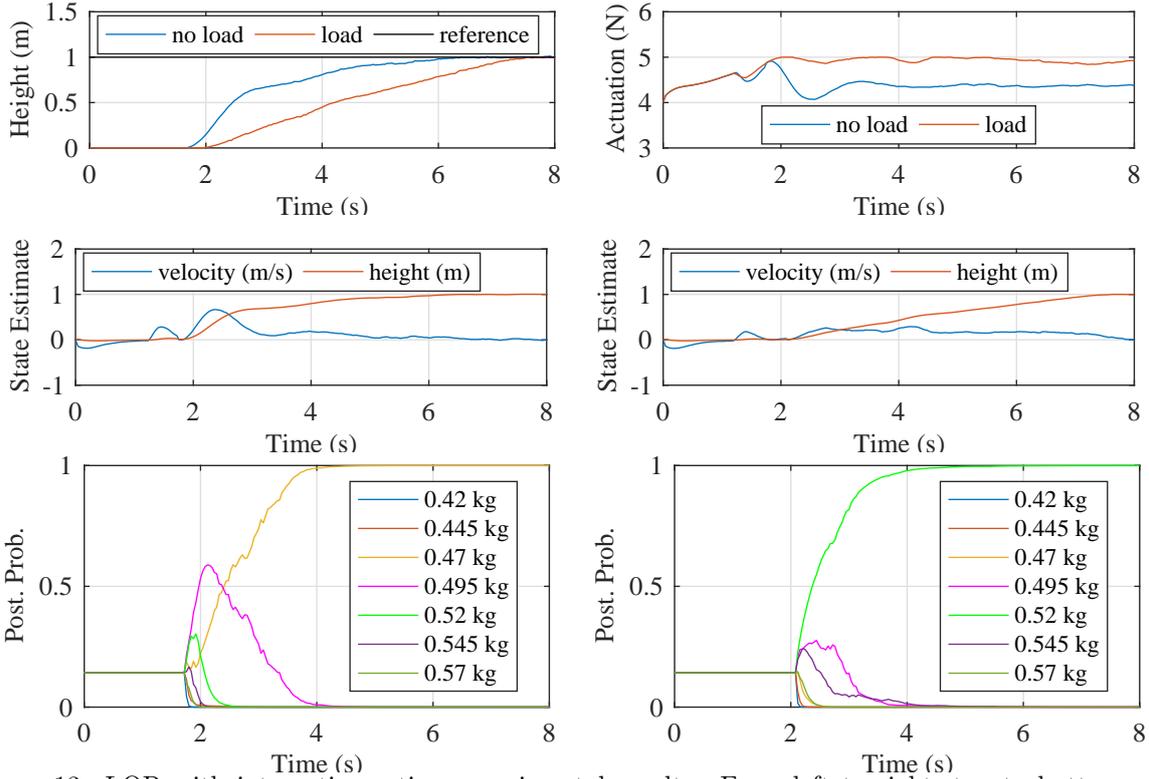


Figure 13: LQR with integrative action experimental results. From left to right, top to bottom: (a) Height for both tests, (b) Actuation for both tests, (c) State estimates for the no load test, (d) State estimates for the load test, (e) Posterior probabilities for the no load test, (f) Posterior probabilities for the load test

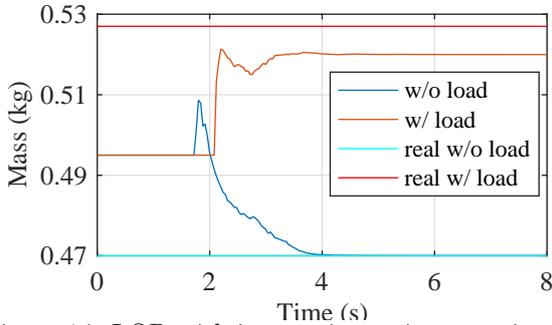


Figure 14: LQR with integrative action experimental results: Mass estimate of both tests.

6.3. MMAC with Integrative Action and Integrative Kalman Filters

The experimental results are presented in Figs. 15 and 16. The thrust gain a had to be changed to 1.025 to ensure accurate mass estimation. It is observed in Fig. 15a that the settling times (5%) are 4 and 6.5 seconds for the no load and with load cases respectively. The one meter height is reached at 4.5 and 6.5 seconds respectively. There was a 4% overshoot in the no load test. The actuation saturates only momentarily and stabilizes with limited variation, as seen in Fig. 15b. In the initial stage of flight in Fig. 15c, the filter that was given more probability was the one with a mass of 0.495 kg, but the filter with the correct mass was selected in the end. In Fig. 15d the selection of the filter

matching the closest mass was also observed and always had the highest probability. Additionally, the selection of the mass for with no load settled in about 2.5 seconds, while for the load case it settled in approximately 2 seconds. The settling time of the probabilities is further corroborated in Fig. 16, where it can also be observed the accurate estimation of the mass in the no load test and an error of 0.007 kg in the load test, due to it not matching the filters.

6.4. Adaptive LQR Controller with Integrative MMAE Estimation

The experimental results are presented in Fig. 17. For the estimator, the thrust gain a had to be changed to 1.025 in order to ensure accurate mass estimation, while for the control it was changed to 1.05. The settling times (5%) are 6 and 6.5 seconds for the no load and with load cases respectively. The static one meter height is reached at 7 seconds in both cases. The height estimate is smoother and follows the measurement very closely for both cases. The estimated velocity is smooth despite the sensitivity of the accelerometer and seems coherent with the height data. In the initial stage of flight, the filter that was given more probability in the no load test was the one with a mass of 0.495 kg, but the filter with the correct mass was selected in the end. For the load test, the selection of the filter match-

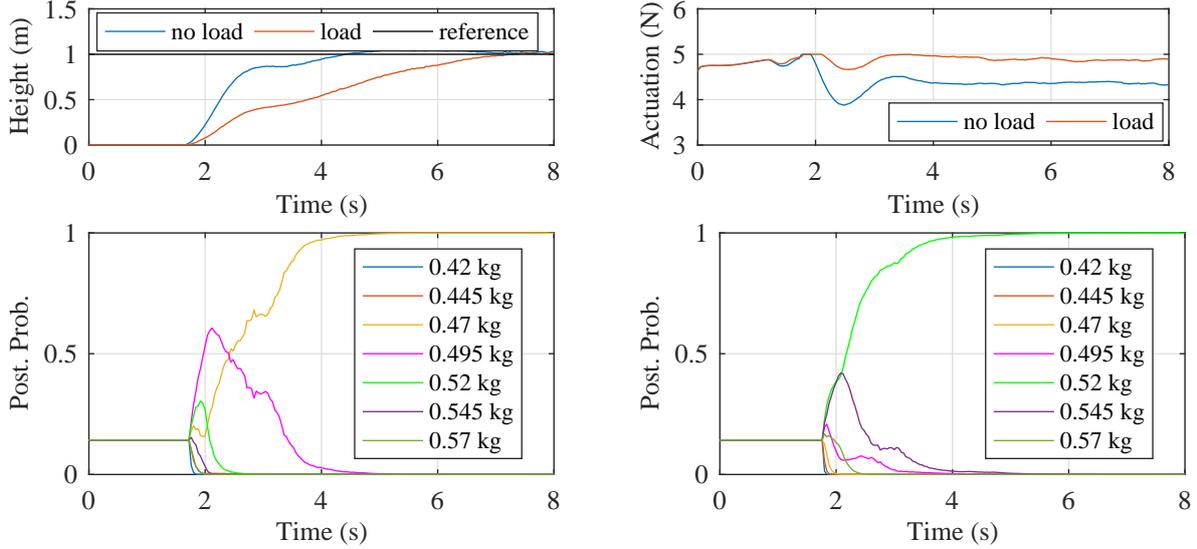


Figure 15: MMAC experimental results: upper left - (a) Height for both tests, upper right - (b) Actuation for both tests, bottom left - (c) Posterior probability no load test, bottom right - (d) Posterior probability load test.

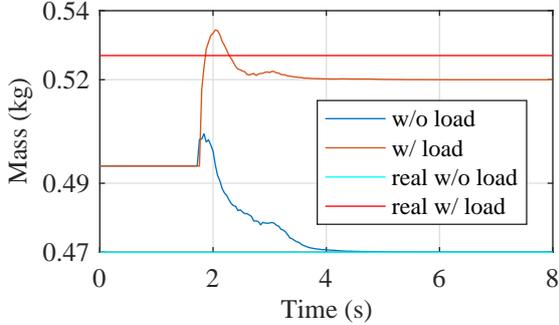


Figure 16: MMAC experimental results: Mass estimates for both tests.

ing the closest mass was also observed and always had the highest probability. Additionally, the selection of the mass with the MMAE settled in about 2 seconds in both cases. The settling time of the probabilities is further corroborated in Fig. 17e, where it can also be observed the accurate estimation of the mass in the no load test and an error of 0.007 kg in the load test, due to it not matching the filters. The adaptive estimate of the mass was accurate for the no load case, but had an error of 0.015 kg on the load case. The height gain of the LQR controller was constant, but the velocity gain decreased slightly with the increasing mass estimate, allowing for more force to be requested in higher mass cases. The actuation saturates at the lift-off stage in both tests and only once more in the load test.

7. Conclusions

The purpose of this thesis was to provide a height control system for an off-the-shelf drone using on-board sensors. Multiple estimation and control solutions were studied, including non-linear control

solutions, which was a requirement of the selected topic. All solutions were tested in simulation and experimentally.

A new method for adjusting Kalman filters was suggested: the Kalman filter with integrative component. In this method, the filter was adjusted with a feedback loop with integration from the residue of a sensor (the height sensor in this case).

The proposed solutions were the LQR with integrative action using an integrative MMAE estimator, the MMAC using LQR with integrative action and Kalman filters with integrative component, and the adaptive controller using an integrative MMAE estimator. All solutions had a settling time equal or under 7 seconds and the maximum overshoot was of 4%. All multi-model methods were capable of selecting the model closest to the real load and the adaptive identification provided accurate estimation on simulation, but had some error on the experimental estimate for the load test.

To finalize the work produced here, it would be relevant to add horizontal position control for full position control, define a suitable Lyapunov function for the unknown load problem, and define Lyapunov based estimation and control approaches. Two other problems to explore could also be the control of a quadrotor with an unknown cable-suspended load and the control of a quadrotor for dropping a load while moving.

Acknowledgements

This work was supported by FCT, through IDMEC, under LAETA, project UID/EMS/50022/2013.

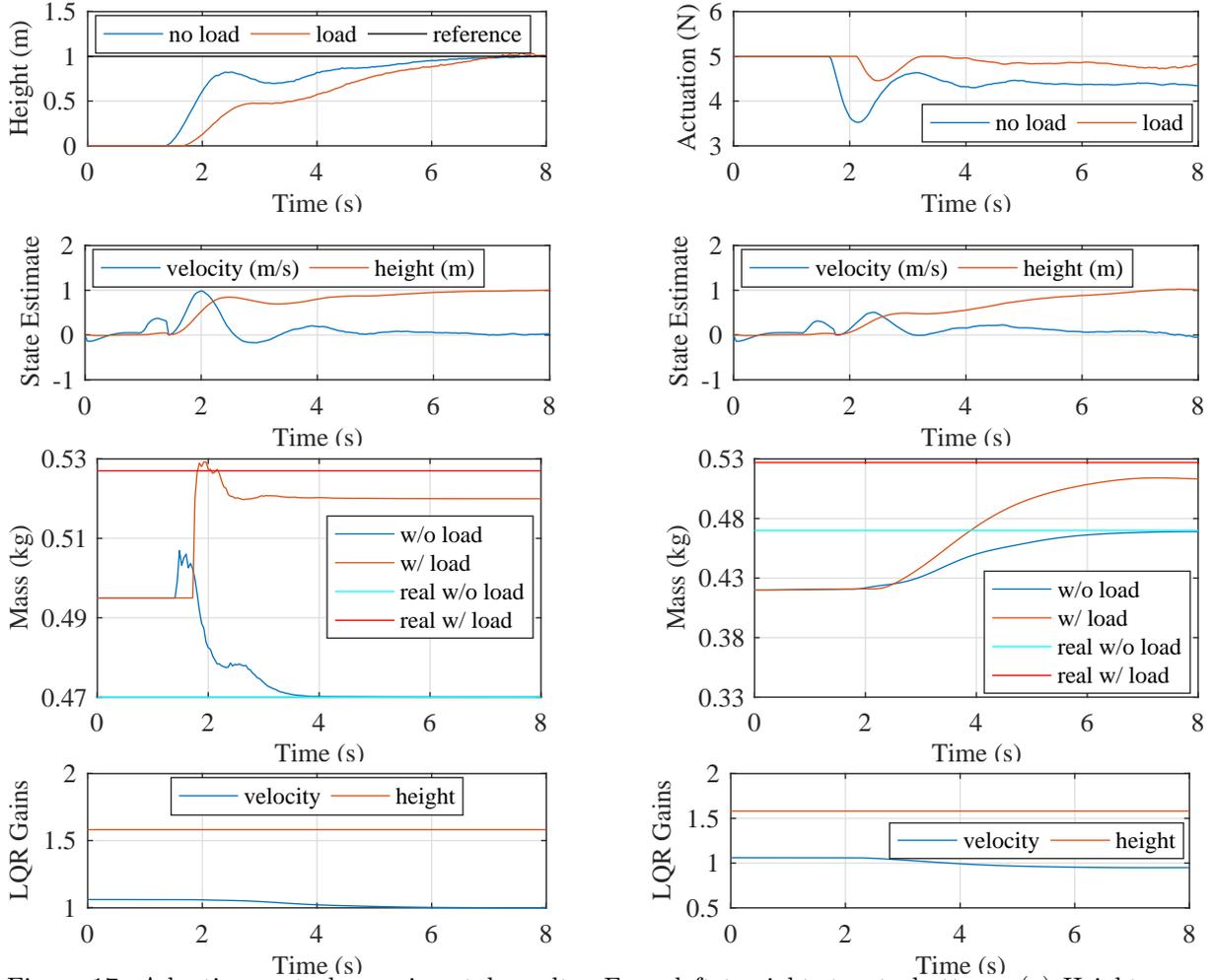


Figure 17: Adaptive control experimental results. From left to right, top to bottom: (a) Height measurement from both tests, (b) Actuation of both tests, (c) State estimates for the no load test, (d) State estimates for the load test, (e) MMAE mass estimate from both tests, (f) Adaptive mass estimate from both tests, (g) LQR gains for the no load test, (h) LQR gains for the load test

References

- [1] S. Dai, T. Lee, and D. S. Bernstein. Adaptive control of a quadrotor uav transporting a cable-suspended load with unknown mass. In *53rd IEEE Conference on Decision and Control*, pages 6149–6154, Dec 2014.
- [2] Sarah Tang and Vijay Kumar. Mixed integer quadratic program trajectory generation for a quadrotor with a cable-suspended payload. In *IEEE International Conference on Robotics and Automation*, 2015.
- [3] Rafael Jos Figueiras dos Santos. Load transportation using rotary-wing uavs. Master’s thesis, Instituto Superior Tcnico, 2015.
- [4] Koushil Sreenath and Vijay Kumar. Dynamics, control and planning for cooperative manipulation of payloads suspended by cables from multiple quadrotor robots. In *Robotics: Science and Systems (RSS)*, 2013.
- [5] R. Mahony, V. Kumar, and P. Corke. Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor. *Robotics Automation Magazine, IEEE*, PP(99):1, 2012.
- [6] N.L.M. Jeurgens. Implementing a simlink controller in an ar.drone 2.0. Master’s thesis, Eindhoven University of Technology, 2016.
- [7] Sajjad Fekri. *Robust Adaptive MIMO Control Using Multiple-Model Hypothesis Testing and Mixed- μ Synthesis*. PhD thesis, Instituto Superior Tcnico, 2002.
- [8] Michael Athans. Multiple-model adaptive estimation (mmae). Slides, 2001.
- [9] Shankar Sastry and Marc Bodson. *Adaptive Control: Stability, Convergence and Robustness*. Prentice-Hall, 1989.
- [10] Darenlee. Ar.drone 2.0 support from embedded coder. <https://www.mathworks.com/hardware-support/ar-drone.html>. Accessed: 30-09-2017.