MMAE/LQR Yaw Control System of a Quadrotor for Constant Unknown Inertia ⋆

Pedro Outeiro, Carlos Cardeira, Paulo Oliveira

IDMEC, Instituto Superior Técnico, Universidade de Lisboa, Portugal
(e-mail: {pedro.outeiro, carlos.cardeira, paulo.j.oliveira}
@tecnico.ulisboa.pt).

Abstract: This paper presents a methodology for angular control of a quadrotor that transports a constant unknown load, given the estimates on both inertia and angular velocity, based on measurements from an indoor multi-camera motion capture system and a gyroscope. The proposed control method is an LQR controller and the proposed estimation method is a Multi-Model Adaptive Estimator (MMAE). The control system obtained is validated both in simulation and experimentally, resorting to an off-the-shelf commercially available quadrotor.

Keywords: Autonomous mobile robots, Estimation algorithms, Parameter identification, Sensor Fusion, Embedded systems

1. INTRODUCTION

The rising interest in drones in recent years has driven a new set of applications for these unmanned aerial vehicles (UAV). Among these applications, a noteworthy one is in delivery systems. Amazon is currently developing delivery drones and systems to support these drones, such as a multi-tiered warehouse. With these applications, one major control concern is the nature of the load. The variable nature of the requested goods leads to an infinite possible set of characteristics about the transported goods. The mass, inertia and position of the item all affect the dynamics of the full system. Therefore, solutions for control and estimation that are robust to uncertain parameters are necessary.

Control methods for drones with loads can be found in Notter et al. (2016). Robust control methods have been previously considered for drones in works like Cabecinhas et al. (2016), and there is plenty of research on these methods. However, robust control methods tend to generate over conservative solutions.

An alternative to Robust Control that is also designed for handling unknown parameters is Multi-model methods. Multi-model methods for piecewise constant unknown mass of quadrotors have been considered and tested in Outeiro et al. (2018a,b). In Outeiro et al. (2018a), the method relied on a Multi-Model Adaptive Estimator (MMAE) with a bank of Integrative Kalman Filters and an LQR controller with integrative action. Because the system had multiple Kalman Filters with an integrative mechanism for gravitational force, it provided reduced state-estimation error. In Outeiro et al. (2018b), the method relied on a Multi-Model Adaptive Controller (MMAC), extending the multi-model framework from Outeiro et al. (2018a) to the control. Because of the use of a bank of combined filters and controllers, this method achieved lower settling times than Outeiro et al. (2018a).

A more common method for handling non-linear and parameter uncertainty is the Extended Kalman Filter (EKF). A derivative of the EKF, the Unscented Kalman Filter (UKF), has been used in agressive flight control for quadrotors in Loianno et al. (2017). Although the EKF and its derivative methods see widespread use, there is the possibility of divergence.

The proposed solution in this paper is an LQR controller with a Multi-Model Adaptive Estimator (MMAE) using a bank of classical Kalman Filters. The sensors used are the gyroscope, which is usually available on-board these types of platforms, and an indoor multi-camera motion capture system.

This paper is organized as follows: the problem addressed in this work is described in Section 2. The physical model considered is presented in Section 3. The behavior of the inertia is analyzed in Section 4. The solution for estimation is presented in Section 5, and in Section 6 the solution for the control problem is proposed. The verification of stability for the full proposed solution is presented in Section 7. Simulation results are presented and discussed in Section 8. In Section 9 the quadrotor model and its sensors are presented. Additionally, the implementation details are discussed in this section. The experimental results are presented and analyzed in Section 10. Finally, some concluding remarks are drawn.

2. PROBLEM STATEMENT

$$I_z \ddot{\psi} = f(u) \tag{1}$$

The yaw dynamics of a quadrotor is shown in (1), where ψ is the yaw angle, I_z is its z-axis inertia, u is the z-axis moment. Since the angular acceleration is related linearly to the provided moment, this is a simple problem to solve. However, this is not as simple for the problem of unknown

⋆ This work was supported by FCT, through IDMEC, under LAETA, project UID/EMS/50022/2019 and project LOTUS of FCT - PTDC/EEI-AUT/5048/2014.

load transportation. Taking into account the added inertia from the load I_l, results in

$$(I_z + I_l)\ddot{\psi} = f\left(u, I_l\right) \tag{2}$$

In this case, the solution for the estimation is not as immediate. Since there is a multiplicative factor that is unknown, the performance of the estimation could degrade when using classical estimation methods. As the error of the unknown parameter versus its assumed value increases, the state-estimation error would also increase. Additionally, only a lower bound for the inertia of the system can be known *a priori*, coinciding with the no-load case. Therefore, an alternative to classical estimation methods is necessary.

3. PHYSICAL MODEL

Consider the dynamics of a quadrotor as shown in Mahony et al. (2012). Isolating the yaw dynamics, the resulting behavior can be described by:

$$I_z\ddot{\psi} = au + \left(I_x - I_y\right)\dot{\theta}\dot{\phi} \tag{3}$$

Where ψ, θ and ϕ are the yaw, pitch and roll angles, I_x, I_y and I_z are the x, y, z axis inertia in the body frame, u is the provided z-axis angular moment and a is an input parameter. It is not intended in this paper to approach the Coriolis effect ($\left(I_x - I_y\right)\dot{\theta}\dot{\phi}$) problem, and therefore it will be removed for the design of the proposed solution. The resulting dynamics are as follows:

$$I_z\ddot{\psi} = au \tag{4}$$

There are measurements available for both the angle and the angular rate, resulting in the following state-space representation for the problem:

$$\begin{aligned} \underbrace{\begin{bmatrix}\ddot{\psi}\\ \dot{\psi}\end{bmatrix}}_{\dot{\boldsymbol{x}}} &= \underbrace{\begin{bmatrix}0 & 0\\ 1 & 0\end{bmatrix}}_{\boldsymbol{A}}\underbrace{\begin{bmatrix}\dot{\psi}\\ \psi\end{bmatrix}}_{\boldsymbol{x}} + \underbrace{\begin{bmatrix}a/I_z\\ 0\end{bmatrix}}_{\boldsymbol{B}} u \\ \psi &= \underbrace{\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}}_{\boldsymbol{C}}\boldsymbol{x} \end{aligned} \tag{5}$$

That will be used in the remainder of the paper.

4. INERTIA BEHAVIOR ANALYSIS

Unlike the mass of a quadrotor, as was studied in Outeiro et al. (2018a,b), the inertia of a system depends on the relative position of its components. To perform system identification for the inertia, it cannot have high frequency variations during the test. The rotation of the rotors provides high speed variations. However, its relative center of mass remains constant. Assuming the rotor blades to be approximately flat in the z-plane, by the Parallel Axis Theorem the rotation does not affect the z-axis inertia.

The position of the load could also be an issue, but to minimize effects on the x and y rotational behavior, the load is considered as being placed above the center of mass of the quadrotor, removing the distance in the z-axis of the load from the center of mass, thus making it constant and known *a priori*.

5. ESTIMATION

In this section the estimation solution is discussed. First the multiple-model estimation framework is presented, followed by the Baram Proximity Measure (BPM), which provides a measure used in model selection.

5.1 Multiple-Model Adaptive Estimation

The Multiple-Model Adaptive Estimation (MMAE) algorithm [Fekri (2002)] is a combined state-estimation and system identification method. Its uses include providing a solution for parametric uncertainties and for non-linear state-estimation (using different liberalizations). As the name implies, it relies on multiple models for the same system, but assuming different values of the unknown parameter (or linearity points). For each of these underlying models a Kalman filter is designed, providing accurate estimates for its assumed model. The merging and processing of the information is provided by the bank of filters. In this algorithm, the Bayesian Posterior Probability Estimator (PPE) 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 higher or lower probability at start. The posterior probabilities $\boldsymbol{p}_{prob_k}(t+1)$, can be computed iteratively using the past probabilities $\boldsymbol{p}_{prob_k}(t)$ and residues of the filters $\boldsymbol{r}_i$ according to (6-8) [Chang and Athans (1978)], where h represents the number of sensors used. The residual covariance matrix of each filter ($\boldsymbol{S}_i$) is used as a weighting parameter in the calculations. $\boldsymbol{\beta}_i(t)$ is a weighting parameter based on the residual covariance and number of sensors, and $\boldsymbol{w}_i(t)$ is a quadratic weighting parameter for the residue which also uses the residual covariance.

$$\boldsymbol{p}_{prob_k}(t+1) = \frac{\boldsymbol{\beta}_k(t+1)e^{-\frac{1}{2}\boldsymbol{w}_k(t+1)}}{\sum_{j=1}^{n}\boldsymbol{\beta}_j(t+1)e^{-\frac{1}{2}\boldsymbol{w}_j(t+1)}}\boldsymbol{p}_{prob_k}(t) \tag{6}$$

$$\boldsymbol{\beta}_i(t+1) = \frac{1}{(2\pi)^{\frac{h}{2}}\sqrt{det\boldsymbol{S}_i(t+1)}} \tag{7}$$

$$\boldsymbol{w}_i(t+1) = \boldsymbol{r}'(t+1)\boldsymbol{S}_i^{-1}(t+1)\boldsymbol{r}(t+1) \tag{8}$$

Given the method for updating the probabilities, the PPE will cause the probability of the filter model with the least residue to tend to one. This means that it selects the most accurate model and uses exclusively its data for estimation. Additionally, an error in the estimation will always be incurred if the real value of the parameter does not match the value assumed by any of the models. The number of filters has to be selected taking into account the estimation error when the system does not match any of the models, while also considering the added computational weight of using more filters, as will be discussed in section 5.2.

The state estimation of the MMAE algorithm can be obtained with a switching or weighted average, see Fekri (2002) for details. In switching the state estimation matches that of the filter with the highest posterior probability. In the weighted average the state-estimation of all filters is averaged using the posterior probabilities as a weighting factor, as shown in (9). In this work, it

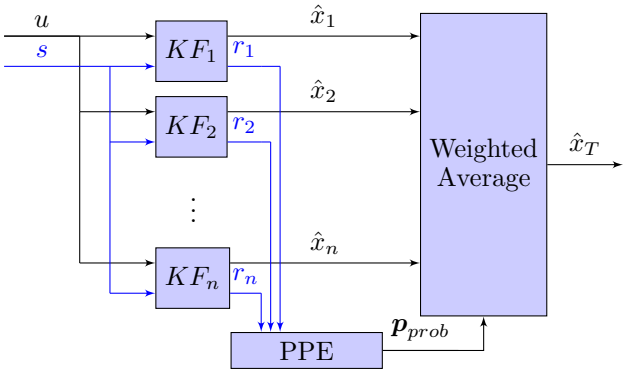

Fig. 1. MMAE Structure.

was adopted the use of the average weight, as it provides low pass filtered state estimates.

$$\hat{\boldsymbol{x}}_T = \sum_{j=1}^{n} \boldsymbol{p}_{prob_j}(t)\hat{\boldsymbol{x}}_j \tag{9}$$

The resulting structure is depicted in Fig. 1.

This method is required to select the real (or closest) value of the parameter. Therefore, a convergence condition must be met for the probability of the closest model, as shown in the following Theorem in Hassani et al. (2009):

Theorem 5.1. Let $i^* \in \{1, 2, ..., N\}$ be an index of a parameter vector in κ and $I := \{1, 2, ..., N\} \setminus i^*$ an index set. Suppose that there exists a positive constant T such that for all $t \geq 0$ and all $j \in I$ the following condition holds:

$$\frac{1}{T}\int_t^{t+T} (\omega_{i^*}(\tau) - ln\beta_j(\tau))d\tau < \frac{1}{T}\int_t^{t+T} (\omega_j(\tau) - ln\beta_{i^*}(\tau))d\tau \tag{10}$$

Then, $p_{i^*}(t)$ satisfies

$$\lim_{t\to\infty} p_{i^*}(t) = 1 \tag{11}$$

Conversely, if (11) is observed, then there exists a positive constant T such that for all $t \geq 0$ and all $j \in I$

$$\frac{1}{T}\int_t^{t+T} (\omega_{i^*}(\tau) - ln\beta_j(\tau))d\tau \leq \frac{1}{T}\int_t^{t+T} (\omega_j(\tau) - ln\beta_{i^*}(\tau))d\tau \,^1 \tag{12}$$

This theorem is presented and proven in Hassani et al. (2009). Although in this work the study was made for the CT-MMAE (continuous time), the findings still hold true for the discrete equivalent.

5.2 Baram Proximity Measure

The Baram Proximity Measure (BPM) is a measure proposed in Baram (1976) by Yoram Baram for deciding which models to use in multi-model methods. Its use has been considered in both Fekri (2002) and Gaspar et al. (2015). The BPM provides an adequate distance metric for stochastic systems between the real model and the Kalman Filter with its assumed model. The calculation

[1] Note that in (12) the inequality is not strict.

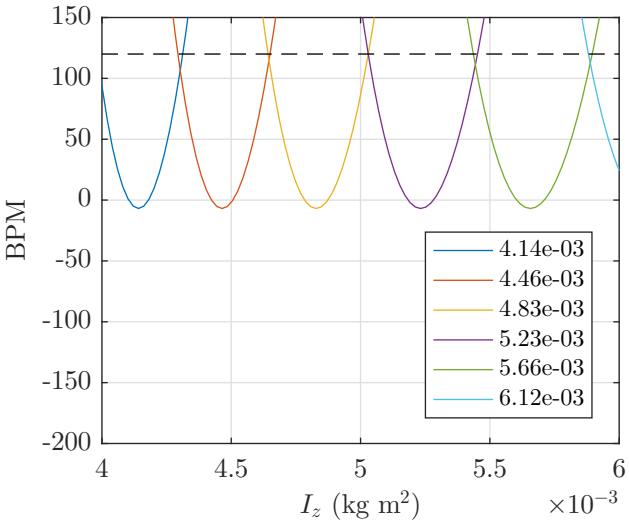

Fig. 2. BPM Curves

of the BPM is presented in (13), where S is the residual covariance matrix of the filter and r is the residue obtained from using the filter.

$$BPM = log(|S|) + tr(S^{-1} \times \gamma)$$
$$\gamma = \frac{r' * r}{length(r)^2} \tag{13}$$

To perform the model selection using this measure, it is necessary to set the desired range for the unknown parameter and the maximum allowed value of BPM. The first model can be selected by increasing the parameter until the BPM of the minimum value of the range does not exceed the maximum. Afterwards, the following models will be required to ensure that the minimum BPM curve of the current and previous models does not exceed the maximum. Finally, models will be added until the minimum BPM curve of all models does not exceed the maximum.

Performing this analysis for the problem addressed in this paper with only a yaw measurement yields the results in Fig. 2. Here, it can be observed that, for a maximum BPM of 120 and for the range of values presented, the obtained number of models is six and the inertias of the models are as presented in the legend of the figure.

6. CONTROL

In this section the proposed controller is presented and the requirement of zero steady-state error is addressed. The proposed control solution is an LQR controller. In the absence of exterior perturbations it should suffice for the purpose of control.

A major requirements for the control is ensuring zero static error, and in this case a classical LQR controller achieves this condition. Given the closed loop transfer function for the system with a classical LQR controller with sample time T, and gains K_ψ and $K_{\dot{\psi}}$ in (14), the steady-state gain is one, and the controller provides zero static error.

$$\frac{\psi}{\psi_{\text{ref}}} = \frac{aT^2K_\psi}{\left[I_z\,(z-1) + aTK_{\dot{\psi}}\right](z-1) + aT^2K_\psi} \tag{14}$$

7. STABILITY VERIFICATION

The stability of the proposed control system architecture is central to the operation of autonomous aerial vehicles in the presence of unknown constant parameters. To analyse this issue, the following lemma from Fekri (2002) is instrumental:

Lemma 7.1. In the case where the real system has an unknown constant parameter that matches the underlying model of one of the filters in the filter bank, the corresponding posterior probability will tend to one and all the other probabilities will go to zero.

Thus the following lemma can be stated:

Lemma 7.2. For a system in the conditions of the previous lemma and based on the Separation Theorem, there is a finite time instant T_{fin} such that, for $t > T_{fin}$, all variables of the closed loop control system are bounded and the yaw rate error converges to zero.

The sketch of the lemma is based on the assumptions previously outlined, namely assuming that one Kalman filter is based on an underlying model with the correct parameter and the eigenvalues of the controller and the estimator are recovered. The resulting A matrix of the system is presented in (15), where T is the sample time, I_m is the model inertia, I_r is the real inertia, L are the Kalman gains and K are the controller gains. In the present case, the design inertia was 5.1×10^{-3} kg m^2, and the real inertia was 4.8×10^{-3} kg m^2. The obtained eigenvalues were $eig = \{0.9902,\ 0.9445 \pm 0.0569i,\ 0.2114\}$. As expected, all have absolute value under one and thus the overall closed loop system is stable.

$$\begin{bmatrix} 1 & 0 & -\frac{K_\psi T a}{I_r} & -\frac{K_\psi T a}{I_r} \\ T & 1 & -\frac{K_\psi T^2 a}{2I_r} & -\frac{K_\psi T^2 a}{2I_r} \\ L_{11} & L_{12} & 1 - \frac{K_\psi T a}{I_m} - L_{11} & -L_{12} - \frac{K_\psi T a}{I_m} \\ L_{21} & L_{22} & T - L_{21} - \frac{K_\psi T^2 a}{2I_m} & 1 - \frac{K_\psi T^2 a}{2I_m} - L_{22} \end{bmatrix} \tag{15}$$

8. SIMULATION RESULTS

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

To validate if the proposed solution could work, a simulation was prepared. The parameters of the drone used for its setup are an inertia I_z of 4.8334×10^{-3} kg m^2 and an input factor a of 1. The inertia after adding a load was 5.9663×10^{-4} kg m^2. The models used were the ones presented in Section 5.2. For the purpose of the Kalman gain calculations, the covariance of the sensor noise was defined as 1 for the yaw rate, and as 10^{-5} for the yaw, while the process noise was given a covariance of 10^{-6}. The LQR gains were calculated using a $Q = [1\ 0;\ 0\ 1]$ and $R = 1$.

The results of the simulation are presented in Fig. 4 and 5. The settling times (5%) with 360º rotations are at 3.5 seconds in both tests. There was no overshoot in both tests. The actuation only saturated during the start of each rotation. The estimation error zeroes after 2 seconds in the no load case and 3 seconds in the load case. These results are expected as the load case is a non-matching case (a case where the real load does not match a model load) and the no load case is a matching case. The probabilities only evolved during the rotations and the maximum probability was attributed to the closest model. The inertia estimate for the no load case converged to the correct value, because it is a matching case. In the load case, the inertia had an error, as it is a non-matching case.

9. IMPLEMENTATION

In this section the experimental set-up is presented.

The model that was used for testing is the Parrot Ar.Drone 2.0 by Parrot SA. It is an off-the-shelf commercially available, general purpose drone designed for users without any drone piloting skills. It is capable of recording video and is a good starter drone, due to its stability.

This model is equipped with an Inertial Measurement Unit (IMU) composed of a three-axis accelerometer, a three-axis gyroscope and a three-axis magnetometer. Other sensors are available on board such as an ultrasound sensor, a barometer and two cameras. One normal camera on the front and an optical-flow camera on the bottom.

For the purpose of implementing the control system, the AR.Drone 2.0 Quadcopter Embedded Coder [Lee (2017)] was used, as it provided a Simulink based environment for development of software to run in the quadrotor and allows direct access to its sensors and actuators.

The controller and estimator from Section 8 were transferred onto the Embedded Coder Simulink environment.

To ensure the stabilization of the quadrotor, the X and Y position and angles were regulated using a cascading PID control system for defining the necessary torques. Additionally, the height was also controlled using a PID controller. The sensors used were an indoor multi-camera motion capture system, that provides the ground truth for the position and orientation, and the gyroscope, that provides the angular rates. The setup of the experiment is presented in Fig. 3.

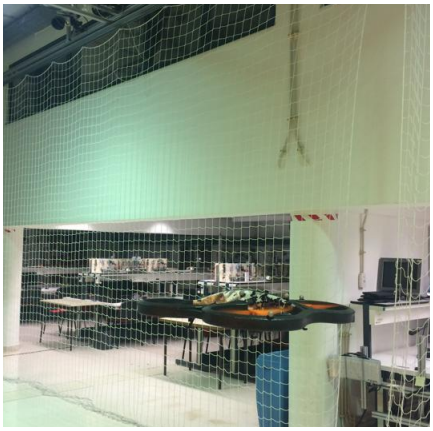

Fig. 3. Experimental setup.

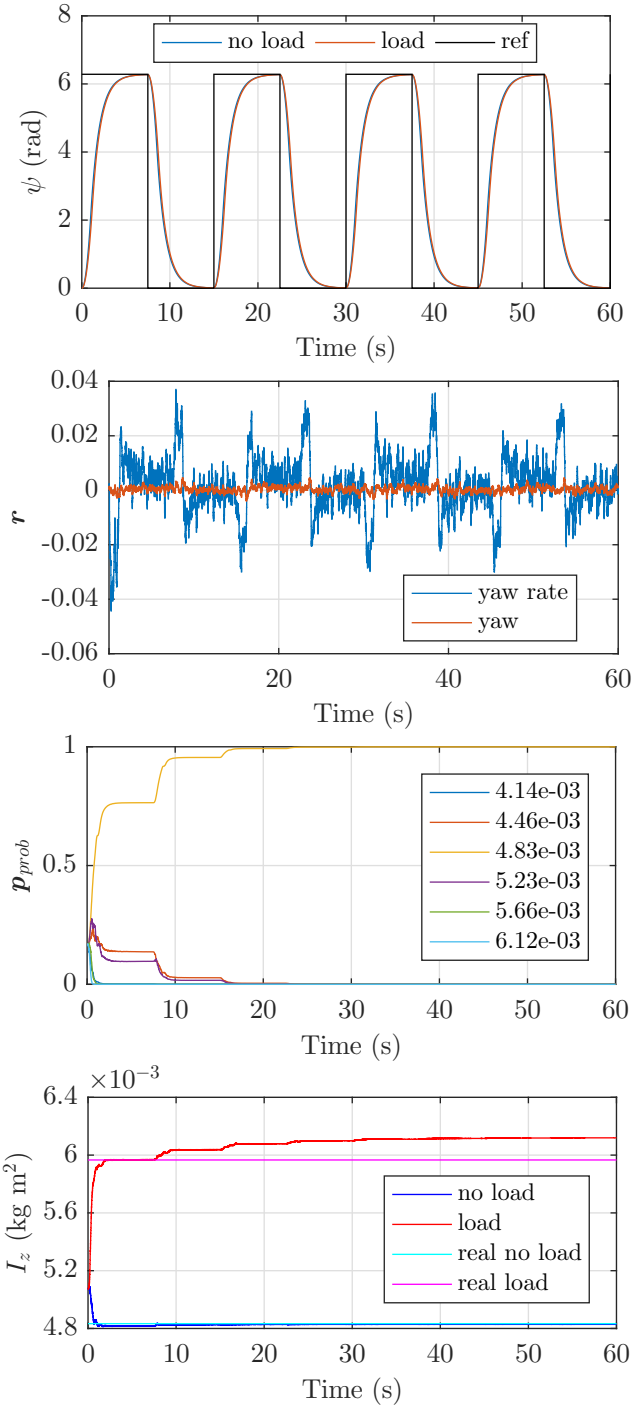

Fig. 4. Simulation results: first - (a) yaw for both tests, second - (b) state error for the no load test, third - (c) posterior probabilities for the no load test, fourth - (d) inertia estimate for both tests.

10. EXPERIMENTAL RESULTS

In this section, the analysis of the results of the experiment are presented.

To validate if the proposed solution worked, an experimental test was prepared. The real inertia I_z of the quadrotor is 4.8334×10^{-3} kg m^2. No test was performed with a load, as it was difficult to increase significantly the inertia without affecting the remainder of the dynamics. The models used were the ones presented in section 5.2.

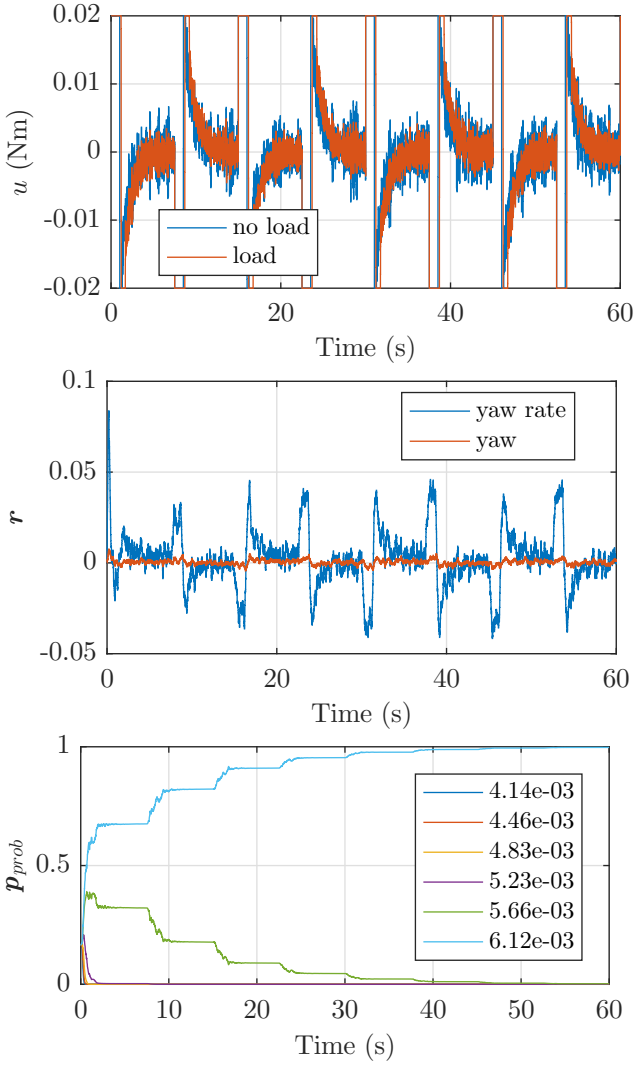

Fig. 5. Simulation results: first - (a) actuation for both tests, second - (b) state error for the load test, third - (c) posterior probabilities for the load test.

For the purpose of the Kalman gain calculations, the covariance of the sensor noise was defined as 10^{-5} for the yaw and as 1 for the yaw rate, while the process noise was given a covariance of 10^{-5}. The LQR gains were calculated using $Q = [1\ 0;\ 0\ 1]$ and $R = 400$.

The results of the test are presented in Fig. 6. The settling time (5%) with 180° rotations is at 5 seconds, but there is a small static error in some of the rotations, and a larger error (0.14 rad) in the last rotation. The actuation only saturated during the start of each rotation. The probabilities evolved in a similar fashion to what was observed in simulation, changing mostly during the rotations. The highest probability was attributed to the closest model. There were peaks in the probabilities of the other models at the start of the rotations, but this did not seem to affect the selection of the closest model. The inertia estimate had a low error, as it is a matching case.

CONCLUSION

This paper proposed and studied the application of Multiple-Model Kalman filtering and LQR control for transportation of unknown loads with quadrotors. The

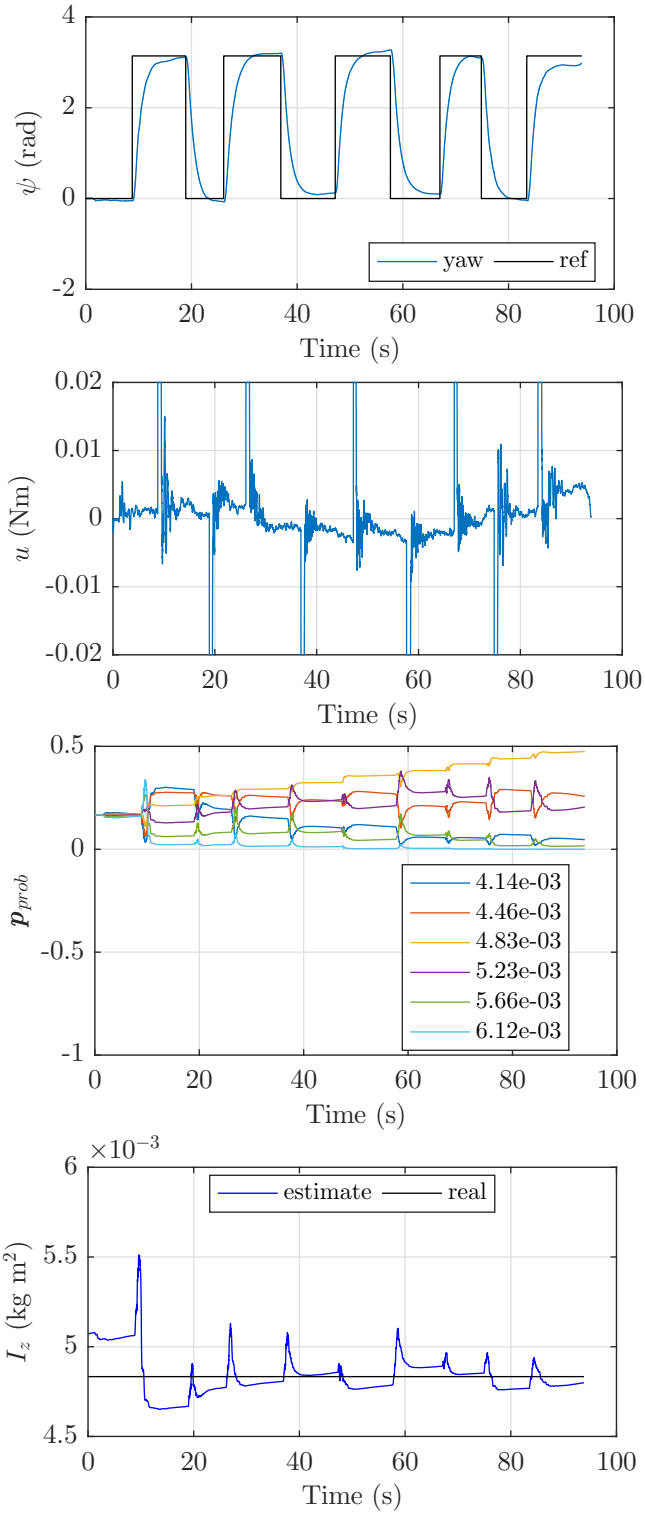

Fig. 6. Experimental results: first - (a) yaw measurement, second - (b) actuation, third - (c) posterior probabilities, fourth - (d) inertia estimate.

sensors used for the proposed solution were an indoor camera indoor multi-camera motion capture system and a gyroscope. The solution was studied in a simulation and experimentally using the Parrot Ar.Drone 2.0. The control and estimation systems provided very low settling times in simulation, while in testing it was much slower to ensure that the x, y and z controllers would be capable of accompanying the rotation. The controller was capable of converging to the desired angle in simulation, while in testing there was static error. The Inertia estimation from the MMAE was capable of selecting the closest mass in the filter bank, and was mostly driven during the rotations. The inertia estimates had only an error in non-matching cases.

REFERENCES

Baram, Y. (1976). *Information, Consistent Information and Dynamic System Identification*. Ph.D. thesis, Massachusetts Institute of Technology.

Cabecinhas, D., Naldi, R., Silvestre, C., Cunha, R., and Marconi, L. (2016). Robust landing and sliding maneuver hybrid controller for a quadrotor vehicle. *IEEE Transactions on Control Systems Technology*, 24(2), 400–412. doi:10.1109/TCST.2015.2454445.

Chang, C.B. and Athans, M. (1978). State estimation for discrete systems with switching parameters. *IEEE Transactions on Aerospace and Electronic Systems*, AES-14(3), 418–425. doi:10.1109/TAES.1978.308603.

Fekri, S. (2002). *Robust Adaptive MIMO Control Using Multiple-Model Hypothesis Testing and Mixed-μ Synthesis*. Ph.D. thesis, Instituto Superior Técnico.

Gaspar, T., Oliveira, P., and Silvestre, C. (2015). Model-based filters for 3-d positioning of marine mammals using ahrs- and gps-equipped uavs. *IEEE Transactions on Aerospace and Electronic Systems*, 51(4), 3307–3320. doi:10.1109/TAES.2015.140748.

Hassani, V., Pedro Aguiar, A., Pascoal, A.M., and Athans, M. (2009). Further results on plant parameter identification using continuous-time multiple-model adaptive estimators. In *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, 7261–7266. doi:10.1109/CDC.2009.5400434.

Lee, D. (2017). Ar.drone 2.0 support from embedded coder. https://www.mathworks.com/hardware-support/ar-drone.html.

Loianno, G., Brunner, C., McGrath, G., and Kumar, V. (2017). Estimation, control, and planning for aggressive flight with a small quadrotor with a single camera and imu. *IEEE Robotics and Automation Letters*, 2(2), 404–411.

Mahony, R., Kumar, V., and Corke, P. (2012). Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor. *IEEE Robotics Automation Magazine*, 19(3), 20–32. doi:10.1109/MRA.2012.2206474.

Notter, S., Heckmann, A., Mcfadyen, A., and Gonzalez, L. (2016). Modelling, simulation and flight test of a model predictive controlled multirotor with heavy slung load. volume 49. doi:10.1016/j.ifacol.2016.09.032.

Outeiro, P., Cardeira, C., and Oliveira, P. (2018a). Lqr/mmae height control system of a quadrotor for constant unknown load transportation. In *2018 13th APCA International Conference on Automatic Control and Soft Computing (CONTROLO)*, 389–394. doi: 10.1109/CONTROLO.2018.8514545.

Outeiro, P., Cardeira, C., and Oliveira, P. (2018b). Mmac height control system of a quadrotor for constant unknown load transportation. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 4192–4197. doi: 10.1109/IROS.2018.8594215.