

Real-time vibration-based propeller fault diagnosis for multicopters

Behnam Ghalamchi, Zheng Jia, and Mark W. Mueller

Abstract—Reliable health monitoring of mechanical components in aerial robotic systems is crucial to their safe operation. The highly constrained nature of aerial systems requires that such systems operate with a minimum of sensing and computational power. This paper proposes a method for the detection and diagnosis of motor/propeller degradation on a multicopter aerial robot. The proposed method works by monitoring the accelerometer output, and effectively correlates vibration power to the motor commands, allowing it to estimate the magnitude of a propeller’s unbalance mass. This is done directly in the time domain with a recursive implementation. The method makes low computational and memory demands, and relies only on the accelerometer almost universally present on aerial robots, so that it may be easily implemented on existing platforms as well as new designs. Experiments show reliable detection of a faulty propeller on three distinct multicopter platforms: two quadcopters whose masses differ by more than an order of magnitude, and a hexacopter; one vehicle has brushed motors, two have brushless motors. The proposed method requires only a minimum of assumptions about the vehicle’s dynamic model, and does not (for example) require knowledge of the vehicle’s center of mass or its mass moment of inertia.

I. INTRODUCTION

UNMANNED aerial vehicles (UAVs) are used in a wide range of applications including structural inspection, parcel delivery, search & rescue, geographic mapping, and increasingly also as passenger-carrying drones. As they become more widely used, issues of reliability and safety become increasingly important. The problem of reliable health monitoring on aerial systems is made difficult by such systems’ thrust and energy constraints, as they must operate with a minimum of sensors and computational power.

A common UAV morphology is the multicopter, which is popular due to its ability to hover and its mechanical simplicity, having only one moving part per actuator, and thus leading to low costs and easy maintenance. A critical point of mechanical failure for such systems is the motor/propeller pair, with potential sources of failure including mechanical wear of the motor bearing or damage to the propellers (typically due to collisions or mishandling). Typical multicopters have four propellers [1], with more safety-critical applications often requiring more propellers to ensure mechanical redundancy in the event of a failure (see e.g. [2] for a discussion on hexacopter design for safety critical applications). Less commonly, fewer than four propellers may also be used [3], [4].

Multicopter dynamics may typically be accurately captured by relatively simple models, allowing for the use of powerful model-based estimation and control strategies. In [5] a model-based approach for estimating faults is presented, using a

Thau’s observer, specifically to detect sensor faults. For faults specific to the actuators, a popular approach is to estimate the actual force produced by each propeller (e.g. using a model-based observer), and to compare this to the force that would be expected given current motor commands. One such approach is presented in [6], which investigates fault-tolerance for a hexacopter design by inverting the system dynamics for detection; in [7] a related approach is applied to an octocopter and also in [8] for a quadcopter. Similarly, in [9] a model-based approach is presented where an additive disturbance is estimated for each actuator, acting as a reduction in the produced thrust. There, each actuator is assigned a state, which is used to estimate actuator effectiveness, and may detect if an actuator produces a force significantly different from the force an undamaged actuator produces. The direct measurement of individual thrust forces is presented in [10], where a quadcopter is equipped with a strain gauge for each actuator, directly measuring the force that the propeller produces. Being a more direct approach, and not relying on the system’s dynamics for estimating the actuator forces means that such an approach should be more robust to model errors (e.g. misalignment of the vehicle’s center of mass, due to a payload); however it requires the addition of multiple sensors and specialized circuitry not typically found on multicopters.

Some control approaches have also been investigated which are capable of accommodating an actuator fault without explicit diagnosis, examples include [11], [12]. A typical approach to encoding faults is as a propeller loss-of-effectiveness, wherein the produced actuator force is some fraction of the desired force, and the goal is to estimate this fraction. An example of a (re-)planning approach for quadcopters experiencing such faults is given in [13]. Examples of systems capable of rearranging their actuator loads are hexacopters (e.g. [14]) and octocopters (e.g. [15]), while quadcopters may execute unusual control strategies to continue flight without one of their actuators [16]. The ability of an overactuated quadcopter-like design to survive failures is analyzed in [17]. A comparison of model-based and data-driven fault detection is presented in [18], specifically applied to aileron fault detection in a fixed-wing UAV.

Timely detection of a fault has obvious benefits, allowing a system to either reconfigure its actuation strategy to reduce the load on damaged actuators, or to execute an urgent landing to bring the vehicle to a safe state rather than risk future catastrophic failure. In less extreme circumstances, fault detection may be used for condition-based maintenance, allowing an operator to understand the evolution of their system health [19]. Much prior work exists on vibration-based fault diagnosis, especially for condition-based maintenance. This is often motivated by the desire to detect/predict tooth-failure in gearboxes (see, e.g., [20], [21] and references therein). A

The authors are with the High Performance Robotics Lab, University of California, Berkeley, CA 94720, USA. {behnam.ghalamchi,zjia,mwm}@berkeley.edu

broader review, including methods using microphones, is given in [19]. Timely identification of faults is especially important for helicopters, where failure of power transmission may be life-threatening. A review of approaches to identify faults in the power transmission system for helicopters may be found in [22]. Often, the goal is simply to determine whether currently observed vibration levels are indicative of a failure condition, by e.g. comparing statistical features to threshold values [20]. For gearboxes, identification is simplified because all rotating components are mechanically coupled. An approach for a gearbox with varying speed and load is given in [23], which combines hypothesis testing with a data-driven auto-regressive with exogenous variables (ARX) model for fault identification.

Herein we present a method for identifying that a failure condition is present in a multicopter UAV, and specifically which motor/propeller pair is faulty, using only already-present sensors. The method uses the power in the accelerometer signal, and is thus related to, e.g., the root-mean-square methods of [20]. Identifying which motor/propeller pair is damaged is complicated by the presence of multiple rotating sources operating at nearly the same operating points, but varying rapidly and independently, making application of existing approaches problematic. Moreover, limited memory and computation motivates the use of a recursive algorithm, that does not require collecting and analyzing batched data. The proposed approach requires a minimum of model knowledge of the system, specifically only requiring the vehicle's approximate total mass, and access to accelerometer measurements and commanded motor forces. The approach is encoded as an extended Kalman filter (EKF), as this specifically allows us to relatively easily reason about uncertain quantities in complex mechatronic systems. The EKF approximately solves the Bayesian estimation problem for a dynamic nonlinear system, given noisy measurements related to the quantities of interest [24]. The extended Kalman filter can be implemented on even low-cost modern microcontrollers, leading to it being widely used in mechatronic systems.

This expands on our prior work in [25], wherein we presented a Fourier-transform-based method for fault identification using accelerometer measurements. In that work, identification required the buffering of a sequence of accelerometer measurements while the vehicle executed carefully designed flight patterns, and the offline analysis of this data. In contrast, the algorithm presented here does not require specially designed trajectories, and can be run in real-time even on very constrained computational hardware.

Specifically, we present a Kalman-filter-based approach that estimates an unbalance for each propeller, taking as input the accelerometer measurements and the motor force commands. The approach is theoretically motivated and analyzed, and then validated in experiment. A scaling argument is presented from which follows that the estimator is likely to be (weakly) more effective on smaller vehicles, as compared to larger vehicles, as well as on vehicles with fewer propellers. Experimental validation is done over three vehicles, including two quadcopters and a hexacopter; with the largest vehicle being thirty times more massive than the smallest; and where one vehicle uses brushed motors, and two use brushless (speed

controlled) motors. Experiments also compare the results for two different trajectories – one more sedate, and the other more informative. Importantly, though the estimator has some tuning factors that must be specified during implementation, we show successful identification across all three experimental platforms with equal parameter settings.

The contribution of this paper is thus 1) the development of a novel propeller fault detection algorithm, based on accelerometer readings of vibrations due to unbalances on the propellers, 2) analysis of scaling effects for this estimator based on vehicle size and design, 3) discussion on the effect of the flight trajectory on the estimator's ability to identify a faulty propeller, and 4) experimental validation of the approach over three distinct vehicles, with multiple trajectories.

The paper proceeds by presenting the necessary dynamics modelling, the fault estimation algorithm, a discussion on the effect of vehicle size and the flown trajectories, experimental results, and finally a conclusion.

II. MODEL

In this section we briefly review the fundamentals of multicopter dynamics, and then develop the relationship between the vehicle's accelerometer output and the unbalances on the vehicle's propellers. We also give brief discussions on multicopter propeller speed control, which is required for the estimator, and force allocation, which is required when discussing scaling effects.

A. Multicopter dynamics

We model the multicopter as a central body (comprising a battery, sensors, payload, etc.) connected to N_p propellers with a collective mass of m_b . The position of the vehicle's center of mass, expressed in the inertial frame, is written as $\mathbf{x}$, with corresponding acceleration vector $\ddot{\mathbf{x}}$. We assume that the propellers have equal radius r_p and are arranged to rotate around the body-fixed thrust direction $\mathbf{e}_3$, with alternating handedness. Each propeller i rotates at some angular velocity ω_i relative to the vehicle body, and produces a thrust force f_i acting on the vehicle body which is related to the propeller's angular speed through the constant $\kappa_{f,i}$ as (see, e.g., [1])

$$f_i = \kappa_{f,i} \omega_i^2 \quad (1)$$

Additional external forces may act on the system, such as aerodynamic disturbances, captured by the force $\mathbf{f}_d$. The translational acceleration of the system's center of mass is then given by

$$m_b \ddot{\mathbf{x}} = \mathbf{R} \mathbf{e}_3 \sum_i f_i + m_b \mathbf{g} + \mathbf{f}_d \quad (2)$$

The rotation matrix $\mathbf{R}$ evolves as a function of the vehicle body-frame's angular velocity $\boldsymbol{\omega}$ relative to the inertial, and the corresponding angular acceleration $\boldsymbol{\alpha}$ is a function of the external moments $\boldsymbol{\tau}$ acting on the system, and the vehicle's mass moment of inertia $\mathbf{J}$ according to

$$\boldsymbol{\tau} = \mathbf{J} \boldsymbol{\alpha} + \boldsymbol{\omega} \times (\mathbf{J} \boldsymbol{\omega}) \quad (3)$$

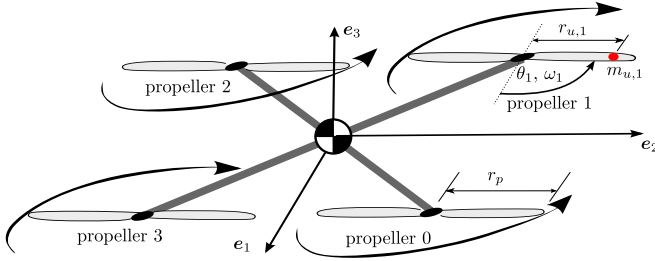


Fig. 1. A quadcopter that has a damaged propeller, indicated by the unbalance on propeller 1. The triad e_1 , e_2 , and e_3 define a body-fixed frame.

The translational dynamics are required for the discussion of the accelerometer output as discussed next, and the rotational dynamics will be used to discuss force allocation when controlling a multicopter.

B. Unbalance and accelerometer measurement

Each propeller has an unbalance of mass $m_{u,i}$ at a distance $r_{u,i}$ from its center of rotation, illustrated for the case of quadcopter in Fig. 1. Note that a damaged propeller can equivalently have a negative mass “added” where the damage is, or a positive mass on the opposite side. The angle of the unbalance with respect to the body-fixed e_1 is θ_i , with corresponding angular speed ω_i .

The rotating unbalance mass causes a radial force on the shaft and transmitted to the multicopter body, which when expressed in the body-fixed coordinate system is given below as $f_{u,i}$

$$f_{u,i} = m_{u,i} r_{u,i} \omega_i^2 \begin{bmatrix} \cos \theta_i \\ \sin \theta_i \\ 0 \end{bmatrix} = \frac{m_{u,i} r_{u,i}}{\kappa_{f,i}} f_i \begin{bmatrix} \cos \theta_i \\ \sin \theta_i \\ 0 \end{bmatrix} \quad (4)$$

As the force is internal, it does not affect the motion of the system center of mass and does not appear in (2).

The vehicle is equipped with an accelerometer, which measures the proper acceleration of the vehicle as expressed in the body-fixed frame, $\mathbf{a}$. This proper acceleration is related to the forces acting on the vehicle, and the vehicle’s total mass m_b , as

$$\mathbf{a} = \frac{1}{m_b} \left(\sum_i (\mathbf{f}_{u,i} + f_i \mathbf{e}_3) + \mathbf{f}_d \right) + \boldsymbol{\nu}_a \quad (5)$$

where $\boldsymbol{\nu}_a$ captures the accelerometer’s additive noise, and $\mathbf{f}_d$ external disturbance forces acting on the vehicle besides the propeller forces and the vehicle’s weight. The summation is to be understood as over all propellers. We assume that the accelerometer is mounted sufficiently close to the vehicle’s center of mass that there are no significant effects due to the vehicle’s angular velocity or angular acceleration. We also assume that the vehicle body is rigid so that there are no internal vibrations. Isolating now the accelerometer’s first two components (i.e. those components in the plane of the

propeller blades) gives

$$a_x = \sum_i \left(\frac{m_{u,i} r_{u,i}}{m_b \kappa_{f,i}} f_i \cos \theta_i \right) + \frac{1}{m_b} f_{d,x} + \nu_{a,x} \quad (6)$$

$$a_y = \sum_i \left(\frac{m_{u,i} r_{u,i}}{m_b \kappa_{f,i}} f_i \sin \theta_i \right) + \frac{1}{m_b} f_{d,y} + \nu_{a,y} \quad (7)$$

For the sake of brevity, we lump the additive disturbances (sensor noise and disturbance force) together into d_x , and introduce the notation Δ_i , so that

$$d_x := \frac{1}{m_b} f_{d,x} + \nu_{a,x} \quad (8)$$

$$\Delta_i := \frac{m_{u,i} r_{u,i}}{m_b \kappa_{f,i}} \quad (9)$$

$$a_x = \sum_i \Delta_i f_i \cos \theta_i + d_x \quad (10)$$

with the equation for a_y following similarly.

C. Propeller speed control

Control of the multicopter is achieved by varying the individually produced motor forces to achieve specified total force and torque requirements (see, e.g., [1]). Typically, this is achieved by computing commanded forces as a linear combination of a desired total force magnitude, and a desired three dimensional torque. These command forces $f_{\text{cmd},i}$ are then typically converted into a commanded angular speed $\omega_{\text{cmd},i}$ for each propeller i , through (1). Usually, each motor-propeller pair then attempts to track this commanded speed using a dedicated electronic speed controller. Some vehicles will instead be equipped with motors that do not offer speed control, for example when using brushed direct current motors, and may instead have some open-loop mapping from desired thrust to e.g. motor PWM command. The true angular speed of a propeller ω_i will, of course, not be exactly equal to the commanded speed in either case. This is due to the unknown dynamic response of the propeller to commands, unmodelled dynamics (e.g. aerodynamic torques acting on the propellers, or angular momentum effects due to the rotation of the body), discretization of the speed controller, etc.

As the deviation of the true angular speed from the commanded is unknown, it is practically impossible to know or predict the angle θ_i of the unbalance, even if it were known initially. One solution to this problem may be to equip the motors with additional encoders, so that their rotation may be known with more certainty; this is however not the approach followed in this work. Instead, the goal of the presented unbalance estimator is to exploit only the accelerometer measurements (6)-(7) and knowledge of the propeller force commands to infer the unbalance, without requiring any knowledge of the angles θ_i .

D. Force allocation

A typical approach to multicopter control is to reduce the system’s N_p force inputs to four kinematic inputs, consisting of the vehicle’s scalar mass-normalized proper acceleration magnitude c , and the desired angular acceleration vector $\boldsymbol{\alpha}$.

These can then be related to a scalar desired total force magnitude $f_\Sigma = m_b c$, and a torque vector $\boldsymbol{\tau}$ through the Newton and Euler dynamic laws (2)-(3). The geometry of the vehicle defines a linear relationship relating the individual motor forces to the total force and torques, represented here by the matrix $A \in \mathbb{R}^{4 \times N_p}$, so that

$$\begin{bmatrix} c \\ \boldsymbol{\alpha} \end{bmatrix} = A \begin{bmatrix} f_1 \\ \vdots \\ f_{N_p} \end{bmatrix} \quad (11)$$

When controlling the vehicle, an inverse relationship is required to compute commanded motor forces $f_{\text{cmd},i}$, typically through a linear map characterized by the ‘mixer matrix’ $M \in \mathbb{R}^{N_p \times 4}$, so that

$$\begin{bmatrix} f_{\text{cmd},1} \\ \vdots \\ f_{\text{cmd},N_p} \end{bmatrix} = M \begin{bmatrix} c \\ \boldsymbol{\alpha} \end{bmatrix} \quad (12)$$

For quadcopters, the mixer matrix is the unique inverse of A , however for vehicles with more than four propellers a pseudo-inverse is typically employed, specifically

$$M = A^T (AA^T)^{-1} \quad (13)$$

which results in the least-norm forces to achieve the desired total force and torque (see, e.g., [26] for an in-depth discussion on hexacopter control allocation). The matrices A and M are functions of the locations of the vehicle’s propellers relative to its center of mass, and the propellers’ thrust-to-torque characteristics.

E. Propeller effectiveness

If a propeller is damaged, it may be expected that the thrust produced is less than the expected thrust – this loss of effectiveness may be captured by the a set of propeller coefficients η_i , so that $\kappa_{f,i} = \eta_i \kappa_f$, with κ_f the nominal propeller coefficient. These coefficients are thus equivalently defined as the ratio between actual propeller thrusts and nominally expected thrusts. The effectiveness may be estimated online, and could also provide an indication of a fault; they are often identified (and inverted) so that a vehicle may be controlled more accurately (see, e.g. the ‘propeller factors’ of [27]).

For near-hover flight, for a symmetric vehicle where all propellers are expected to produce equal force in hover, and assuming that the motors track the commanded velocities accurately, the factors may be estimated using the average thrust produced as below

$$\hat{\eta}_i = E \left[\frac{f_{\text{cmd},i}}{f_i} \right] \approx \frac{N_p}{m_b \|g\|} E[f_{\text{cmd},i}] \quad (14)$$

where $m_b \|g\| / N_p$ is the force that each propeller should produce on average. This uses the fact that, near hover, the vehicle has zero average translation, thus also zero average acceleration, so that the forces must average to the hover forces. This method may be compared to the proposed vibration-based estimation, and will be used in the experimental section of this paper.

III. ALGORITHM

This section presents the unbalance estimator, using the accelerometer measurements to identify the propeller unbalances. We assume access to a reasonable estimate of the propellers’ angular speeds (e.g. through the control commands), however, this is assumed sufficiently imprecise that we treat the angles θ_i as unknown – and specifically we assume no knowledge about these angles, so that they are modelled as uniformly randomly distributed. The presented estimator may thus be used on any multicopter, even if it is not equipped with complex speed control/motor angle measurement sensors.

A. Assumptions and simplifications

The goal of the estimator is to estimate the unbalances $m_{u,i} r_{u,i}$, using the accelerometer measurements a_x and a_y , but without relying on precise knowledge of the unbalance angles θ_i or the true propeller speed ω_i . The approach is to exploit the signal energy in the accelerometer measurements. Importantly, the estimator operates in the time domain, and does not require any manipulation in the frequency domain (e.g. Fourier transforms), unlike our prior work in [25]. We assume that the motors are capable of approximately tracking the commanded angular speeds, so that the commanded forces are close to the true forces.

Furthermore, we assume that the propellers rotate at high speeds, and that the unbalance angles θ_i at the time instants corresponding to the accelerometer measurements may be treated as independent sequences, each being white and uniformly distributed over $[0, 2\pi)$ (in other words, assuming total ignorance of the propeller’s rotation, and that the rotation is sufficiently fast so that it may be treated as white). Under these assumptions, we can compute the expected value of the accelerometer measurement $E[\cdot]$, as below

$$E[a_x] = \frac{1}{m_b} \bar{f}_{d,x} + \bar{\nu}_{\alpha,x} \quad (15)$$

with $\bar{f}_{d,x}$ and $\bar{\nu}_{\alpha,x}$ representing the average disturbance force and the accelerometer bias, respectively. Thus, the expected accelerometer measurement contains no information about the propeller unbalances. The signal power, on the other hand, depends on the unbalances and will be used for the estimator.

B. Estimator measurements

We define a scalar measurement at time step k , $z(k)$, as follows, which is expanded upon substitution:

$$z(k) := a_x(k)^2 + a_y(k)^2 \quad (16)$$

$$\begin{aligned} &= \sum_i \Delta_i^2 f_i(k)^2 + d_x(k)^2 + d_y(k)^2 \\ &+ \sum_i \sum_{j \neq i} \Delta_i \Delta_j f_i(k) f_j(k) \cos(\theta_i(k) - \theta_j(k)) \\ &+ \sum_i 2\Delta_i f_i(k) (d_x(k) \cos \theta_i(k) + d_y(k) \sin \theta_i(k)) \end{aligned} \quad (17)$$

The estimator uses the instantaneous signal power $z(k)$ as measurement, that is a scalar measurement for each (vector-valued) accelerometer measurement. Taking the expectation of

(17), and substituting the assumption that the propeller angles are independent and uniformly distributed yields

$$\mathbb{E}[z(k)] = \sum_i \Delta_i^2 f_i(k)^2 + \mathbb{E}[d_x^2] + \mathbb{E}[d_y^2] \quad (18)$$

This measurement is informative, as its expected value depends on the unbalances ($m_{u,i} r_{u,i}$), with the relative influence depending on the instantaneous motor forces $f_i(k)$, which will vary over the course of the flight. Intuitively, for those flight segments where propeller i produces a larger force, the corresponding unbalance's contribution to the accelerometer signal power will be larger.

The measurement uncertainty is modelled through the additive measurement noise $\nu_z(k)$, defined as

$$z(k) =: \mathbb{E}[z(k)] + \nu_z(k) \quad (19)$$

which is zero-mean by construction, and whose variance may be tediously computed as

$$\begin{aligned} \text{Var}[\nu_z(k)] &= \mathbb{E}[\nu_z(k)^2] = \mathbb{E}[z(k)^2] - \mathbb{E}[z(k)]^2 \quad (20) \\ &= \mathbb{E}[(d_x(k)^2 + d_y(k)^2)^2] - \mathbb{E}[d_x(k)^2 + d_y(k)^2]^2 \\ &\quad + 2(\mathbb{E}[d_x(k)^2] + \mathbb{E}[d_y(k)^2]) \sum_i \Delta_i^2 f_i(k)^2 \\ &\quad + \sum_i \sum_{j \neq i} \Delta_i^2 \Delta_j^2 f_i(k)^2 f_j(k)^2 \end{aligned} \quad (21)$$

Evaluating the above is difficult, as it requires information about the fourth moment of the random variables d_x and d_y . However, under the assumption that these quantities are zero-mean, independent, identically, and normally distributed, with variance σ_d^2 , (21) may be simplified to

$$\begin{aligned} \text{Var}[\nu_z(k)] &\approx 4\sigma_d^4 + \sum_i \sum_{j \neq i} \Delta_i^2 \Delta_j^2 f_i(k)^2 f_j(k)^2 \\ &\quad + 4\sigma_d^2 \sum_i \Delta_i^2 f_i(k)^2 \end{aligned} \quad (22)$$

where we've used the fact that, for a zero-mean, normally distributed quantity x , $\mathbb{E}[x^4] = 3\text{Var}[x]^2$.

C. Estimating motor forces

From (18) it can be seen that the estimator will require knowledge of the propeller forces, or equivalently the angular speed of the propellers. Some multicopters are equipped with electronic speed controllers that return the current motor speeds to the vehicle's main control board, though often this is not the case (especially in lower-cost vehicles). When direct measurement of the motor speeds is not available, a simple method to estimate the motor's speed is to assume that they track the commanded angular speed as a first-order system (as in e.g. [28]) with time constant τ_m , so that

$$\dot{\hat{\omega}}_i = \frac{1}{\tau_m} (\omega_{\text{cmd},i} - \hat{\omega}_i) \quad (23)$$

From the estimated propeller speed, the propeller force can be recovered using (1).

D. Unbalance estimator

The relationship of (18) is at the heart of the identification algorithm. Specifically, our goal is to use the knowledge of the commanded motor forces $f_{\text{cmd},i}$ and a sample variance from the accelerometer to identify the unbalances $m_{u,i} r_{u,i}$. We estimate the unbalances by using an $(N_p + 1)$ -state extended Kalman filter, where the first N_p states x_i relate to the unbalances as follows:

$$e^{x_i} := \Delta_0^{-1} \frac{m_{u,i} r_{u,i}}{m_b k_{f,i}} \quad (24)$$

and the last state is related to the additive noise, with

$$e^{x_{N_p+1}} := \sigma_{d,0}^{-1} \sqrt{\mathbb{E}[d_x^2]} \quad (25)$$

where Δ_0 is an initial guess for all the unbalances, and $\sigma_{d,0}$ is an initial guess for the additive disturbance's standard deviation. The formulation as an exponential relationship is used to ensure that the quantities of interest remain positive (as is physically meaningful) without requiring the application of constraints in the Kalman filter (which may be problematic, see e.g. [29]). The estimator state vector is then given by $\hat{x}(k) \in \mathbb{R}^{N_p+1}$, with associated covariance matrix $P(k) \in \mathbb{R}^{(N_p+1) \times (N_p+1)}$.

The estimator measurement model is

$$z(k) = \sum_{i=1}^{N_p} f_i(k)^2 e^{2x_i} \Delta_0^2 + 2e^{2x_{N_p+1}} \sigma_{d,0}^2 + \nu_z(k) \quad (26)$$

$$=: h(x, k) + \nu_z(k) \quad (27)$$

where ν_z is the measurement noise as defined in (19).

1) *Estimator equations:* The unbalances' dynamics, as well as that of the estimated additive noise variance, is modelled as being forced by zero-mean, white, and independent noise. In addition to being simple to implement, this model allows the estimator to adapt to changing conditions (e.g. if a propeller is damaged mid-flight), as well as to "forget" bad information (e.g. due to poor initialization). This noise is quantified by its scalar standard deviation. For the unbalances, this is denoted by q_Δ , and for the additive noise variance's process noise by q_d , so that the variance prediction equation is

$$P_p(k) = P(k) + \text{diag}(q_\Delta^2, \dots, q_\Delta^2, q_d^2) \Delta t \quad (28)$$

wherein P_p and P are the variance respectively after the prediction and measurement steps, and Δt is the sampling period of the accelerometer (and thus the period of the estimator). Due to the exponential encoding of the state, the quantities q_Δ and q_d have units $\text{s}^{-\frac{1}{2}}$ and may be interpreted as multiplicative uncertainties.

The measurement Jacobian matrix $H(k)$, to be used in the filter measurement update, is given by

$$H(k) := \left. \frac{\partial h}{\partial x} \right|_{\hat{x}(k)} \in \mathbb{R}^{1 \times (N_p+1)} \quad (29)$$

$$H_i = 2f_i(k)^2 e^{2\hat{x}_i(k)} \Delta_0^2 \text{ for } i \in 1, \dots, N_p \quad (30)$$

$$H_{N_p+1} = 4e^{2\hat{x}_{N_p+1}(k)} \sigma_{d,0}^2 \quad (31)$$

so that the Kalman filter gain is computed at time k using the usual Extended Kalman filter equations (see e.g. [24], but repeated here for completeness) as

$$K(k) = P_p(k)H(k)^T (H(k)P_p(k)H(k)^T + R_z(k))^{-1} \quad (32)$$

$$\hat{x}(k+1) = \hat{x}(k) + K(k) (z(k) - h(\hat{x}(k), k)) \quad (33)$$

$$P(k+1) = (I - K(k)H(k)) P_p(k) \quad (34)$$

wherein

$$R_z(k) = \text{Var}[\nu_z(k)] + r_0 \quad (35)$$

with $\text{Var}[\nu_z(k)]$ computed according to (22) and substituting the estimate $\hat{x}(k)$, and where r_0 is an additional tuning term used to decrease the estimator's reliance on the measurements.

2) *Implementation assumptions*: Implementation of the above estimator requires that a designer specify the estimator initialization, $P(0)$ and the initial beliefs for the unbalances Δ_0 and the accelerometer noise $\sigma_{d,0}$; and the process noise magnitudes q_Δ and q_d , and the additional measurement noise variance r_0 which are effectively tuning parameters. Further discussion on choosing the tuning parameters is given in the experimental validation section. Note that the estimator state $\hat{x}$ is always initialized to zero.

The implementation also assumes that the disturbances (external force and accelerometer noise) are zero-mean, independent, identically normally distributed. The zero-mean assumption can be approximated well in practice by applying a high-pass filter to the accelerometer output.

Notable is that the estimator does not require any knowledge about most vehicle physical parameters, such as its mass moment of inertia, or the location of the vehicle's center of mass. Moreover, an error in the belief of the vehicle mass m_b will simply cause a proportional error in the estimate of the unbalances; however the relative values of the unbalances should be mostly unaffected. Furthermore, no detailed knowledge of the disturbance forces f_d is required. This implies that the estimator may be easily used in situations where these parameters are not (well) known, e.g. in uses such as package delivery where the payload often changes, and is hard to characterize accurately. This may be compared to model-based loss-of-effectiveness fault detection approaches, which rely on comparison of the force command to the observed dynamics of the vehicle – in such cases the estimate will be very sensitive to model mismatch, such as an error in the belief of the vehicle's center of mass causing an apparent torque imbalance in the vehicle.

IV. SCALING EFFECTS AND INFORMATIVE TRAJECTORIES

In this section we reason about the utility of the proposed algorithm for vehicles of different sizes and with varying numbers of propellers, as well as for different trajectories executed by a vehicle. We make some assumptions to deduce that the algorithm should be more effective for vehicles with fewer propellers. Moreover, the method is expected to be more effective for smaller vehicles, though the scaling relationship may be weak.

A. Informative motions

Estimating the propeller unbalances relies on the individual forces f_i for the different propellers varying over time relative to one another (in a sufficiently distinct manner), so that the effects of the propellers' unbalances are distinct. It is obvious that, if all commanded forces are equal, the estimator will not be able to distinguish between propellers – this may occur for a vehicle at hover in a substantially disturbance-free environment, for example, or when tracking a constant velocity trajectory. Similarly, if all forces vary in unison, such as when performing motions that are primarily along the vertical axis, the estimator will also be unable to distinguish the effects of the various unbalances.

However, a vehicle maneuvering through space, specifically including horizontal motions that require force differences to produce torques, would have more informative forces and allow for clearer identification of the unbalances. Indeed, a vehicle attempting to maintain a constant position while rejecting external disturbances will provide similar information as a vehicle maneuvering without disturbances, and thus permit identification. A more detailed, quantitative analysis of the informativeness of different trajectories is beyond the scope of this work.

B. Scaling effects

1) *Vehicle mass*: The amplitude of the unbalance's contribution to the accelerometer signal scales with $(m_{u,i}r_{u,i}\omega_i^2)/m_b$, from (4). Under the assumption that the propeller's mass scales with the mass of the vehicle, it may be that the mass of a typical unbalance of concern also scales with the mass of the vehicle, so that the unbalance signal scales like $r_{u,i}\omega_i^2$. We assume that the unbalance is at the propeller tip, so that $r_{u,i}$ is simply the propeller radius, which is assumed to scale proportionally to the vehicle length scale, and thus cubically to the vehicle mass, so that $r_{u,i}^3 \sim m_b$.

Two multicopter scaling laws are suggested in [30], which may be used to gain insight into the overall scaling effect. The first is Mach scaling, which assumes that the propeller tip speed $r_p\omega_i$ is constant – this implies that the accelerometer signal due to the unbalance will scale like $1/r_p$, or proportionally to $m_b^{-1/3}$. The second approach is Froude scaling, which effectively assumes that the product $r_p\omega_i^2$ is constant, so that the effect is independent of the vehicle size. Thus, in practice, the unbalance signal will scale either independently of the vehicle size, or inversely proportionally to the vehicle scale.

2) *Number of propellers*: There are a few considerations when analyzing the effect of varying the number of propellers on the estimator's efficacy. If the vehicle mass and propeller size are held constant, but the number of propellers is varied, the unbalance's effect scales inversely proportionally to the number of propellers – this is due to the factor ω_i^2 scaling like the individual force, which scales inversely proportionally to the number of propellers. Therefore, a vehicle with more propellers will experience lower vibrations due to an unbalance, all else being equal. Note that this is a very approximate



Fig. 2. The vehicles used in the experiments, with physical parameters given in Table I.

relationship: adding more motors typically adds a substantial mass to the vehicle, with the exact relationship depending on the vehicle structure, payload, etc.

Furthermore, the greater the number of propellers (and thus larger estimator state-space), the larger the total uncertainty in the system. The estimator may thus be expected to converge more slowly for vehicles with more propellers, simply due to the space of the uncertainty being larger.

Finally, assuming that a vehicle follows a fixed kinematic trajectory, so that the angular acceleration and proper acceleration are fixed, the value of the individual motor forces are affected by the mixer matrix M , as defined in (13). Noting that the estimator relies on variations over time of the motor forces to identify unbalances, the variance of the motor forces may be used as an approximation of their informativeness. As the forces are linearly related to the kinematic quantities of total acceleration and angular acceleration (neglecting the quadratic angular velocity term in (3)), the singular values of the mixer matrix give some indication to the ‘gain’ of the variance from the kinematic quantities to the commanded motor forces. Tedious computation shows that a vehicle with N_p propellers located symmetrically about the vehicle’s center of mass at a distance l has mixer matrix with non-zero singular values $\left\{ \frac{m_b}{\sqrt{N_p}}, \frac{2J_{xx}}{\sqrt{N_p}l}, \frac{J_{zz}}{\sqrt{N_p}\kappa_\tau} \right\}$, where κ_τ is the scalar relating the propellers’ force to the torque about their rotational axes. The terms J_{xx} and J_{zz} are the two distinct components of the mass moment of inertia, corresponding to the mass moment of inertia about an axis perpendicular to the propellers’ thrust direction for J_{xx} , and parallel to the propellers’ thrust direction for J_{zz} . This relationship suggests that, all else being equal, a vehicle with more propellers flying the same trajectory will have more difficulty in identifying a propeller unbalance, as the force’s individual variability will be lower. The presence of co-axial rotors pairs does not fundamentally alter the above considerations, as long as the propellers rotate at independent speeds.

V. EXPERIMENTAL VALIDATION

In this section the estimator is experimentally evaluated, and shown to be able to identify a damaged motor/propeller pair, and also accurately estimate the magnitude of the unbalance on the damaged propeller. Experiments are done on three different vehicle types, spanning a thirty-fold change in total vehicle mass, and having either four or six propellers. Notably, furthermore, the smallest scale vehicle features brushed motors

TABLE I
PHYSICAL PARAMETERS OF EXPERIMENTAL VEHICLES

	Small	Medium	Large
propeller count N_p	4	4	6
mass m_b , g	39	690	1170
propeller radius r_p , mm	32	100	100
thrust const. $\kappa_{f,i}$, $\text{Ns}^2/\text{rad}^2 \times 10^{-8}$	4.14	764	764



Fig. 3. The damaged propellers used in the experiments, with the removed propeller sections indicated by dashed outlines, on the right-hand side. The damage to the small propeller is approximately 5mg, and that to the large propeller is approximately 40mg.

which do not have any closed-loop speed control, unlike the two larger vehicles having electronic speed controllers. For the smallest vehicle the actual forces produced are likely to vary more from the truth than those of the larger vehicles.

The experiments show the estimator applied to vehicles with nominal propellers, and with a damaged propeller. Two different trajectories are used: one where the vehicle has to move between randomly chosen setpoints with sharply varying motor forces; and a second trajectory where the vehicle flies along a horizontal circle, with comparatively smoother motor forces along the trajectory.

A. Experimental setup

Experiments are conducted in an in-door laboratory environment featuring a motion capture system used for state position/attitude tracking of the vehicles. All vehicles are equipped with a Crazyflie 2.0 flight controller [31] (featuring an STM32F4 microcontroller and an MPU9250 inertial measurement unit). The unbalance estimator, IMU, and motor commands are updated at 500Hz, running on this microcontroller. The flight control stack is based on a modified PX4-stack [32], and off-board code uses ROS [33]. Note that the use of the motion capture system should have no effect on the evaluation of the estimator; as the estimator relies only on onboard information, the results will be the same if the vehicle flies using GPS, or is remotely piloted.

Three different vehicles are used, as shown in Fig. 2, and their relevant physical parameters are given in Table I. Experiments were done with all nominal propellers, or with a single damaged propeller. Propellers were manually damaged by cutting off a part of the propeller tip: the damaged propellers are shown in Fig. 3.

B. Estimator implementation

The estimator was implemented with the following parameters, identical across all three vehicle types unless otherwise noted. The time constant with which the motors are assumed to track the commands is set to $\tau_m = 15\text{ms}$. The additional measurement noise was set to $r_0 = 100\text{m}^2/\text{s}^4$ – a value that

was experimentally identified as providing acceptable results for all platforms. The initial belief of the accelerometer noise standard deviation was set to $\sigma_{d,0} = 1\text{ms}^{-2}$. The process noise for the accelerometer noise state was chosen to be $q_d = \frac{0.05}{\sqrt{60}}/\sqrt{s}$, and that for the unbalances as $q_\Delta = \frac{0.2}{\sqrt{60}}/\sqrt{s}$. These values were chosen based on the intuition that the accelerometer noise will change slowly (in this case, the standard deviation of one minute's drift corresponds to $e^{0.05} \approx 5\%$ change), while the unbalance masses may change more quickly (corresponding to approximately 22% magnitude change per minute).

The initial state uncertainty was set to be diagonal, with the standard deviation of the unbalance states set to 0.1, and for the accelerometer noise state set to 0.05. The accelerometer bias was filtered out with a first-order high-pass filter with cutoff frequency set to 1Hz. The medium quadcopter and large hexacopter unbalances in the estimator state were initialized at the equivalent of 20mg, and the small quadcopter with 2mg, at the tips of their respective propellers. Though the estimator state encodes the variables Δ , they are transformed for all plots and discussions to the equivalent positive unbalance masses at the tip of the vehicle propeller through (24). The estimator is not overly sensitive to the selection of the tuning parameters: identical estimator parameters (including tuning parameters) were used in all experiments, across the three different vehicle types, with the only differences being the number of propellers, the propeller radii, and the initial guess for the propeller unbalance.

C. Results

1) *Small quadcopter*: Three experiments are shown in Fig. 4, which demonstrates the efficacy of the proposed estimator on a very small quadcopter with brushed motors. When the vehicle has no damaged propellers, the estimated unbalances quickly decay to values close to zero, and remain there. In the case of having a damaged propeller, the estimator can quickly identify which motor/propeller is problematic, and is able to estimate the magnitude of the unbalance to the correct order of magnitude. Estimator convergence is shown for two different trajectories: either the vehicle flies between randomly generated position setpoints, or the vehicle tracks a horizontal circle at constant speed. When tracking between randomly assigned points, the motor forces have a much larger variance, providing more information for the estimator to identify a damaged propeller. For this vehicle, and this damaged propeller, the estimator however performs similarly for both trajectories, and rapidly identifies the damaged propeller with consistent estimates of the unbalance. Also visible on the graph is the effect on the accelerometer measurements when a damaged propeller is present, showing an approximately three-fold increase in amplitude of the accelerometer measurements. In each experiment, the estimator converges after approximately 15s.

2) *Medium quadcopter*: For the medium quadcopter, the estimator can also successfully and reliably identify a damaged propeller, as shown in Fig. 5. The figure compares the estimator performance when flying two different trajectories: one

with randomly chosen position setpoints, and another where the vehicle flies a steady horizontal circle. In both cases, the estimator correctly identifies the faulty propeller, but notably converges slower than with the small quadcopter (requiring on the order of 100s to converge). The slower convergence corresponds to the predictions made in Section IV, and convergence time may be reduced by changing the tuning of the estimator (recall that all experiments used identical parameters).

The commanded motor forces vary much more when the vehicle flies between randomly assigned points, than when flying along the circle, therefore the estimated accelerometer noise is higher for the 'noisier' trajectory due to the estimated propeller speeds being farther from the true speeds over large force ranges (due to e.g. motor torque limits).

3) *Large hexacopter*: For the large hexacopter, the estimator can also successfully and reliably identify a damaged propeller, as is shown in Fig. 6. As for the small quadcopter trajectories, three cases are shown: an undamaged vehicle flying to randomly assigned points, a damaged vehicle flying to randomly assigned points, and a damaged vehicle flying a horizontal circle. Again, the estimator is capable of identifying whether a damaged propeller is present, identifying the specific damaged propeller, and estimating the unbalance to the correct order of magnitude. Convergence with the hexacopter is slower than with the two quadcopters, and the circular trajectory is slightly less informative than the random points trajectory.

Notable is that the estimator converges to an unbalance value in the range 100 to 120mg on the hexacopter, and 40 to 50mg on the medium quadcopter, though the same damaged propeller was used. There are two likely causes for this discrepancy. The first is that the vehicles' structures are not perfectly rigid, as was assumed in the modelling section. A second cause is the non-linear nature of the estimator where the states are encoded through the exponential function. Due to its convexity, a higher variance will lead to an over-estimate of the states, which can be affected by the choice of estimator tuning parameters.

4) *Comparison to loss-of-effectiveness*: Due to the propeller damage, the propeller may be expected to produce less thrust at a given angular velocity than an undamaged propeller. We quantify this using the estimated propeller effectiveness factors as defined in Section II-E, specifically (14).

For the small quadcopter experiments, as shown in Fig. 4, we compute the propeller factors during the flight, substituting the sample average for the expectation of (14), and specifically restricting the sample to a 20s interval, starting 5s after the vehicle takes off. The propeller effectiveness for the flights are below:

Flight condition	$\hat{\eta}_0$	$\hat{\eta}_1$	$\hat{\eta}_2$	$\hat{\eta}_3$
No damage, random points	1.17	1.08	1.10	1.01
Damage at 0, random points	1.09	1.06	1.06	1.01
Damage at 0, circle	1.18	1.03	1.08	1.00

Recall that a value of 1.0 indicates a nominal propeller. The data shows that there is no strong correlation between the damage applied to propeller 0, and the propeller factors. Thus, for this system, any loss of effectiveness that may be perceived is completely masked by variability in the performance of the brushed motors. This is in contrast to the presented approach,

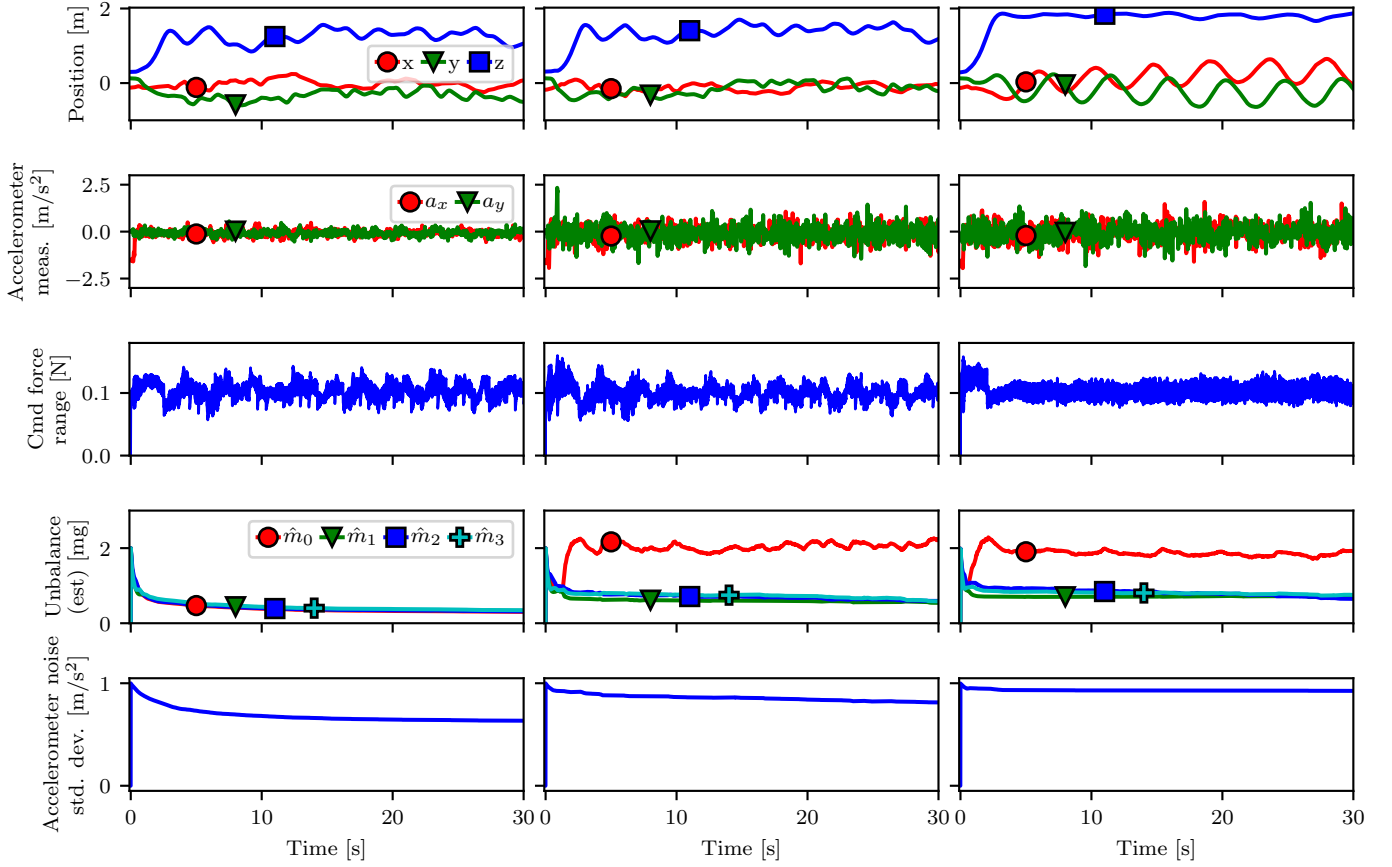


Fig. 4. Three experiments done with the small quadcopter. The left-hand column shows a vehicle with no damaged propellers flying between randomly assigned positions, the middle column shows the same trajectory but with a damaged propeller located at motor 0. The right-hand column also has a damaged propeller at motor 0, but the vehicle instead flies a circular horizontal trajectory of radius 1m with velocity 1 m s^{-1} .

which was able to rapidly and accurately identify the damaged propeller.

VI. CONCLUSION

We have presented a propeller fault detector that uses accelerometer measurements to recursively estimate unbalances present in a multicopter's rotors. The intuition behind the estimator is to correlate changes in the accelerometer noise power to changes in the individual propeller speeds or motor force commands, allowing the estimator to identify a damaged propeller, and the magnitude of the damage. The estimator is shown to work reliably on three diverse platforms, with substantially different total masses, and different actuators (brushed/brushless motors).

The algorithm may be useful to small, low cost vehicles, where adding complex additional sensors for fault detection is prohibitively expensive or heavy. For larger systems, which may already include additional sensors, the algorithm provides an inexpensive, independent method for identifying potential drive-train issues.

To use the algorithm in a system, a designer would need to extend what is presented in order to make decisions based on the estimates; one example would be to define a threshold for tolerable propeller unbalances – if a vehicle exceeds these thresholds, a maintenance event is triggered. Alternatively, a

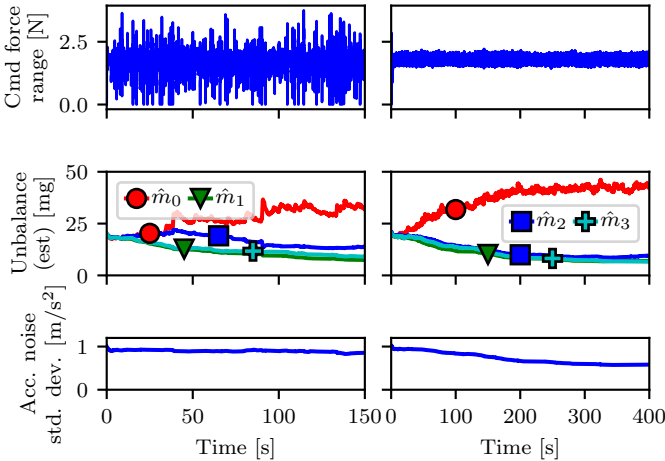


Fig. 5. Performance of the estimator on the medium quadcopter, with a damaged propeller on motor 0. The left-hand column shows the vehicle flying a trajectory between randomly assigned position setpoints (and thus having rapidly varying motor forces), and the right-hand column of graphs shows the vehicle flying a horizontal circular trajectory of radius 1m at a speed of 1 m s^{-1} . The estimator correctly identifies the damaged propeller, and the estimate converges to the same value in both cases.

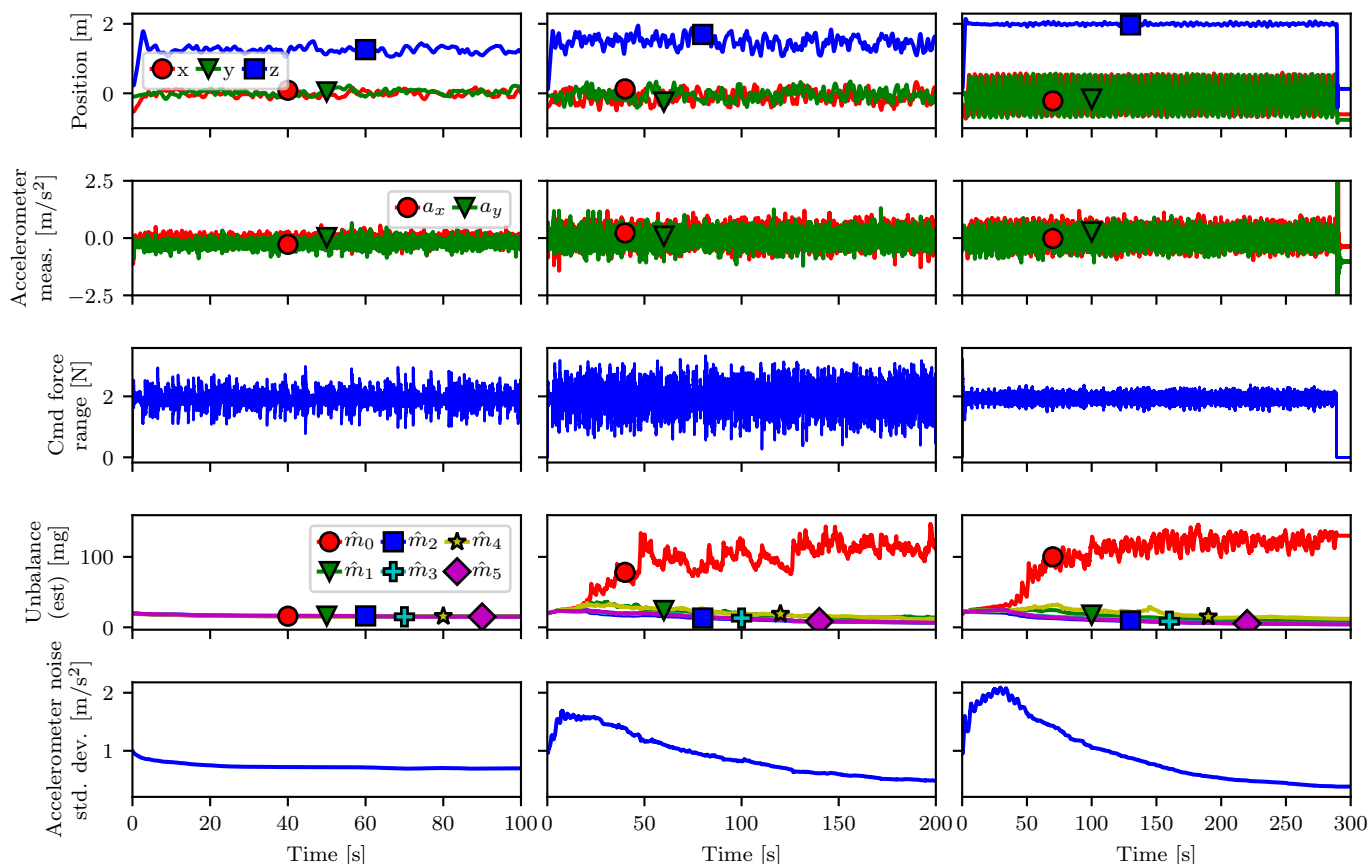


Fig. 6. Performance of the estimator on the large hexacopter, with a damaged propeller on motor 0. The columns, from left to right, show the vehicle flying a trajectory between randomly generated points without propeller damage, flying the same trajectory with a damaged propeller at motor 0, and flying a circular trajectory with a damaged propeller at motor 0. The third row plots the range of the forces commanded, i.e. the minimum and maximum motor force commanded at each time instant.

health monitoring system may look at the change in unbalances, rather than their absolute value, to flag a maintenance event, or trigger an emergency landing. Because the estimator is able to identify the specific damaged motor/propeller pair, maintenance may be substantially simplified (no expert visual inspection of all propellers required, for example).

The algorithm may also be readily adapted to other aerial systems with propellers, such as tailsitter systems (e.g. [34]), or other hybrid designs (e.g. [35]).

ACKNOWLEDGEMENTS

This work was supported by the Department of Mechanical Engineering, UC Berkeley, and the Finnish Cultural Foundation. The experimental test bed at the HiPeRLab is the result of contributions of many people, a full list of which can be found at hiperlab.berkeley.edu/members/.

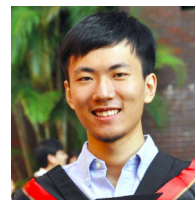
REFERENCES

- [1] R. Mahony, V. Kumar, and P. Corke, “Aerial vehicles: Modeling, estimation, and control of quadrotor,” *IEEE Robotics & Automation Magazine*, vol. 19, no. 3, pp. 20–32, 2012.
- [2] J. I. Giribet, R. S. Sanchez-Pena, and A. S. Ghersin, “Analysis and design of a tilted rotor hexacopter for fault tolerance,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, no. 4, pp. 1555–1567, 2016.
- [3] W. Zhang, M. W. Mueller, and R. D’Andrea, “A controllable flying vehicle with a single moving part,” in *IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2016, pp. 3275–3281.
- [4] M. Piccoli and M. Yim, “Piccolissimo: The smallest micro aerial vehicle,” in *IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2017, pp. 3328–3333.
- [5] A. Freddi, S. Longhi, and A. Monteriu, “A model-based fault diagnosis system for a mini-quadrotor,” in *7th workshop on Advanced Control and Diagnosis*, 2009, pp. 19–20.
- [6] C. T. Raabe and S. Suzuki, “Adaptive, failure-tolerant control for hexacopters,” in *AIAA Infotech@ Aerospace (I@A) Conference*, 2013, pp. 1–8.
- [7] M. Saied, B. Lussier, I. Fantoni, C. Francis, H. Shraim, and G. Sanahuja, “Fault diagnosis and fault-tolerant control strategy for rotor failure in an octorotor,” in *IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2015, pp. 5266–5271.
- [8] R. C. Avram, X. Zhang, and J. Muse, “Quadrotor actuator fault diagnosis and accommodation using nonlinear adaptive estimators,” *IEEE Transactions on Control Systems Technology*, vol. 25, no. 6, pp. 2219–2226, 2017.
- [9] A. Hasan and T. A. Johansen, “Model-based actuator fault diagnosis in multirotor UAVs,” in *International Conference on Unmanned Aircraft Systems*, IEEE, 2018, pp. 1017–1024.
- [10] J. Bazin, T. Fields, and A. J. Smith, “Feasibility of in-flight quadrotor individual motor thrust measurements,” in *AIAA Atmospheric Flight Mechanics Conference*, 2016, p. 1760.
- [11] M. Saied, B. Lussier, I. Fantoni, H. Shraim, and C. Francis, “Passive fault-tolerant control of an octorotor using super-

- twisting algorithm: Theory and experiments,” in *Control and Fault-Tolerant Systems (SysTol)*, IEEE, 2016, pp. 361–366.
- [12] R. C. Avram, X. Zhang, and J. Muse, “Nonlinear adaptive fault-tolerant quadrotor altitude and attitude tracking with multiple actuator faults,” *IEEE Transactions on Control Systems Technology*, vol. 26, no. 2, pp. 701–707, 2018.
- [13] A. Chamseddine, Y. Zhang, C. A. Rabbath, C. Join, and D. Theilliol, “Flatness-based trajectory planning/replanning for a quadrotor unmanned aerial vehicle,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 4, pp. 2832–2848, 2012.
- [14] D. Scaramuzza, M. Achtelik, L. Doitsidis, F. Friedrich, E. Kosmatopoulos, A. Martinelli, M. Achtelik, M. Chli, S. Chatzichristofis, L. Kneip, D. Gurdan, L. Heng, G. H. Lee, S. Lynen, M. Pollefeys, A. Renzaglia, R. Siegwart, J. Stumpf, P. Tanskanen, C. Troiani, S. Weiss, and L. Meier, “Vision-controlled micro flying robots: From system design to autonomous navigation and mapping in GPS-denied environments,” *IEEE Robotics Automation Magazine*, vol. 21, no. 3, pp. 26–40, Sep. 2014.
- [15] A. Marks, J. F. Whidborne, and I. Yamamoto, “Control allocation for fault tolerant control of a VTOL octorotor,” in *UKACC International Conference on Control*, IEEE, 2012, pp. 357–362.
- [16] M. W. Mueller and R. D’Andrea, “Relaxed hover solutions for multicopters: Application to algorithmic redundancy and novel vehicles,” *The International Journal of Robotics Research*, vol. 35, no. 8, pp. 873–889, 2016.
- [17] A. Nemati, R. Kumar, and M. Kumar, “Stabilizing and control of tilting-rotor quadcopter in case of a propeller failure,” in *ASME 2016 Dynamic Systems and Control Conference*, American Society of Mechanical Engineers, 2016, V001T05A00.
- [18] P. Freeman, R. Pandita, N. Srivastava, and G. J. Balas, “Model-based and data-driven fault detection performance for a small UAV,” *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 4, pp. 1300–1309, 2013.
- [19] P. Henriquez, J. B. Alonso, M. A. Ferrer, and C. M. Travieso, “Review of automatic fault diagnosis systems using audio and vibration signals,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 44, no. 5, pp. 642–652, 2013.
- [20] M. Lebold, K. McClintic, R. Campbell, C. Byington, and K. Maynard, “Review of vibration analysis methods for gearbox diagnostics and prognostics,” in *Proceedings of the 54th meeting of the society for machinery failure prevention technology*, 2000, pp. 623–634.
- [21] A. S. Sait and Y. I. Sharaf-Eldeen, “A review of gearbox condition monitoring based on vibration analysis techniques diagnostics and prognostics,” in *Rotating Machinery, Structural Health Monitoring, Shock and Vibration, Volume 5*, Springer, 2011, pp. 307–324.
- [22] P. D. Samuel and D. J. Pines, “A review of vibration-based techniques for helicopter transmission diagnostics,” *Journal of sound and vibration*, vol. 282, no. 1-2, pp. 475–508, 2005.
- [23] M. Yang and V. Makis, “ARX model-based gearbox fault detection and localization under varying load conditions,” *Journal of Sound and Vibration*, vol. 329, no. 24, pp. 5209–5221, 2010.
- [24] D. Simon, *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons, 2006.
- [25] B. Ghalamchi and M. W. Mueller, “Vibration based propeller fault diagnosis for multicopters,” in *International Conference on Unmanned Aerial Systems*, IEEE, 2018, pp. 1041–1047.
- [26] G. Ducard and M.-D. Hua, “Discussion and practical aspects on control allocation for a multi-rotor helicopter,” in *Conference on Unmanned Aerial Vehicle in Geomatics*, 2011, pp. 1–6.
- [27] S. Lupashin, M. Hehn, M. W. Mueller, A. P. Schoellig, M. Sherback, and R. D’Andrea, “A platform for aerial robotics research and demonstration: The flying machine arena,” *Mechatronics*, vol. 24, no. 1, pp. 41–54, 2014.
- [28] M. W. Mueller and R. D’Andrea, “Stability and control of a quadcopter despite the complete loss of one, two, or three propellers,” in *IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2014, pp. 45–52.
- [29] S. J. Julier and J. J. LaViola Jr, “On Kalman filtering with nonlinear equality constraints,” *IEEE Transactions on Signal Processing*, vol. 55, no. 6, pp. 2774–2784, 2007.
- [30] A. Kushleyev, D. Mellinger, C. Powers, and V. Kumar, “Towards a swarm of agile micro quadrotors,” *Autonomous Robots*, vol. 35, no. 4, pp. 287–300, 2013.
- [31] Bitcraze, *Crazyflie 2.0*, www.bitcraze.io/crazyflie-2/, Accessed: 2018-10-03.
- [32] L. Meier, P. Tanskanen, L. Heng, G. Lee, F. Fraundorfer, and M. Pollefeys, “Pixhawk: A micro aerial vehicle design for autonomous flight using onboard computer vision,” *English, Autonomous Robots*, vol. 33, no. 1-2, pp. 21–39, 2012.
- [33] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, “ROS: An open-source robot operating system,” in *ICRA Workshop on Open Source Software*, 2009.
- [34] Y. Ke, K. Wang, and B. M. Chen, “Design and implementation of a hybrid UAV with model-based flight capabilities,” *IEEE/ASME Transactions on Mechatronics*, 2018.
- [35] J. Apkarian, “Attitude control of pitch-decoupled VTOL fixed wing tiltrotor,” in *International Conference on Unmanned Aircraft Systems*, IEEE, 2018, pp. 195–201.



Behnam Ghalamchi is lecturer at California Polytechnic State University - San Luis Obispo . He obtained his doctoral degree from Lappeenranta University of Technology in 2014. From 2017-2019 he was post doctoral researcher (funded by Finnish Cultural Foundation) at HiPeRLab, UC Berkeley, CA.



Zheng Jia obtained his bachelors degree in Electronic and Information Engineering from The Hong Kong Polytechnic University in 2013 and a masters in Robotics, Systems and Control from ETH Zurich in 2017. He joined the HiPeR Lab of UC Berkeley as a short-term visiting scholar starting from June 2018 to December 2018. He is now a visiting scholar at Intelligent Control Systems of Max Planck Institute for Intelligent Systems. His main research interest is distributed control and coordination of multi-robot systems.



Mark W. Mueller is an assistant professor in the Mechanical Engineering Department at the University of California, Berkeley. He received a bachelors degree from the University of Pretoria, a masters from ETH Zurich in 2011, and a DrSc from ETH Zurich in 2015, all in Mechanical Engineering. Current research focuses on control and design of aerial robots for challenging environments.