State Estimate Recovery for Autonomous Quadcopters

Luciano Beffa, Anton Ledergerber, and Raffaello D'Andrea

Abstract— **A method for recovery from the complete loss of the state estimate is presented for autonomous quadcopters. Given an aerodynamic force model, the only measurements used to reinitialize the state estimate by means of a bank of extended Kalman filters are the angular rate and linear acceleration measurements of an IMU. The method is integrated within a complete recovery logic on a quadcopter platform and experimentally evaluated.**

I. Introduction

Over the course of the last two decades, unmanned aerial vehicles (UAVs), and particularly quadcopters, have received increasing attention. Low-cost, lightweight sensing technology paired with powerful processors enabled their widespread use as an experimental platform for research in the field of robotics, as well as in industry: for example for aerial delivery, inspection, surveillance, photography, mapping, inventory management and entertainment.

With the emergent use of quadcopters, their safety and robustness have become important requirements. Ideally, a quadcopter is completely fault-tolerant, meaning that it is engineered in such a way that a failure affecting any of the components of the system does not prevent the intended operation being completed.

An intuitive concept to create a fault-tolerant system is redundancy. By duplicating every component, a quadcopter can continue operation relying on the backup components, which almost completely eliminates the risk of a catastrophic failure. The major drawbacks of redundancy, however, are the increased cost, complexity and weight of the system. Redundancy for actuator fault-tolerance can alternatively be achieved by duplicating only a subset of components: For example, in [1], [2], [3] and [4] fault-tolerant control for hexacopters is studied. Another alternative way to achieve actuator redundancy is to introduce different means of actuation, for example by rotors that can be actively tilted [5]. In the context of sensor fault-tolerance, redundancy is typically not achieved by duplication of components, but by a sensor configuration that provides redundant information, see for example [6]. A summary of work on sensor and actuator fault-tolerance for quadcopters can be found in [7].

A less strict requirement is a fail-safe design: Such a system is designed to respond to a failure by taking an action that minimizes damage and/or harm to the equipment and its environment. For example, in [8] a control strategy is proposed that allows a quadcopter to continue controlled flight despite the loss of up to three propellers, without increasing the mechanical complexity of the system, but by controlling a reduced attitude instead of the full attitude.

Related work in sensor failure detection and isolation, as well as robust estimation allows estimation and control despite sensor failure or the presence of outliers (see for example [9], [10]). These methods may however be bound by limitations. Consider for instance the scenario in which a state estimate is lost or corrupted. Such a scenario may arise when a system using computer vision to estimate its states experiences a sudden loss of its tracked features due to aggressive maneuvers, or due to sudden illumination changes [11]. Also GPS-based state estimators can diverge due to jammed or lost GPS signal reception [12], or due to unmodeled forces from collisions or impacts. Finally, an unforeseen software or hardware issue may freeze the onboard processing unit leading to a system reset. At best, these failures temporarily corrupt the state estimate, and the estimated state converges back close to the actual state. At worst, these failures result in the complete loss of a reliable state estimate, which may in turn have catastrophic consequences.

The aim of this work is the development of a method to reinitialize the state estimate of quadcopters in such worst case scenarios. We evaluate the method by throwing the vehicle and letting it autonomously recover its state estimate without an initial attitude estimate. Since a quadcopter may only produce thrust along its thrust axis, it is essential that this axis is aligned in the opposite direction to gravity as quickly as possible to prevent the vehicle from further accelerating towards the ground when in free fall. Therefore, it is necessary that the vehicle rapidly obtains an estimate of the direction of gravity.

The problem of reinitializing the state estimate of a quadcopter has previously been addressed. For example, in [11] a method enabling a quadcopter to recover from a throw is presented. In [13] a similar method is developed. Both methods succeed in reinitializing the state estimate after aggressive maneuvers, but share a common, significant assumption: It is assumed that the direction of gravity can essentially be measured before the vehicle is launched, providing the subsequent steps with a reasonable initial estimate of the attitude. In practice, this is realized by turning the system on while it is still in the operator's hand, allowing the accelerometer to approximately measure gravity. While methods relying on this assumption are shown to work well for launching the vehicles, they do not allow for state estimate recovery mid-air or in free fall.

The main contribution of this work is the development of a method that allows the vehicle to recover from the com-

The authors are with the Institute for Dynamic Systems and Control, ETH Zurich, Switzerland. The contact author is Luciano Beffa luciano.danilo.beffa@alumni.ethz.ch.

plete loss of its state estimate. More precisely, the method described here relies solely on measurements provided by an IMU comprised of an accelerometer and a gyroscope to determine the direction of gravity and velocity of the vehicle whilst in free fall. It is shown that this may be approached by formulating a Bayesian tracking problem, and approximately solving it using a bank of extended Kalman filters (bank of EKFs).

The remainder of this paper is organized as follows: In Section II, the mathematical model of a quadcopter platform is briefly summarized. The aerodynamic force model used therein is discussed in detail in Section III. Section IV explains how these models are used in a bank of EKFs framework to find the attitude and velocity of the quadcopter platform. Integration in a complete recovery logic is presented in Section V. Finally, Section VI discusses the experimental evaluation of the suggested method.

II. System Model

In this section, the equations describing the motion and measurements of a quadcopter platform are recalled and the necessary variables for the following sections are introduced. For notational convenience, vectors are denoted as n-tuples $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ with dimensions and stacking clear from the context.

A. Equations of Motion

Let I denote an inertially fixed Cartesian coordinate frame, with its z-axis pointing in the opposite direction to gravity, and B denote a Cartesian body-fixed frame, with its z-axis pointing along the vehicle thrust axis. The other axes of both frames are chosen to satisfy the right-hand rule.

The quadcopter is modeled as a single rigid body. The state of the vehicle is described by the state variable

$$\boldsymbol{x} = (\boldsymbol{p}, \boldsymbol{v}, R, \boldsymbol{\omega}), \tag{1}$$

where $\boldsymbol{p} \in \mathbb{R}^3$ is the position, and $\boldsymbol{v} \in \mathbb{R}^3$ is the linear velocity of the center of mass of the vehicle, $R \in SO(3)$ is the attitude, and $\boldsymbol{\omega} \in \mathbb{R}^3$ is the angular velocity of the vehicle. The attitude R represents a coordinate transformation from the inertial frame I to the body-fixed frame B, i.e. for a vector $\boldsymbol{\xi}$ it satisfies the equation

$${}_B\boldsymbol{\xi} = R\,{}_I\boldsymbol{\xi}, \tag{2}$$

where the prefixes denote the frames in which $\boldsymbol{\xi}$ is expressed.

The kinematic relations of the position and attitude can be written as

$${}_I\dot{\boldsymbol{p}} = {}_I\boldsymbol{v} \tag{3}$$

$$\dot{R} = -[\![{}_B\boldsymbol{\omega}]\!]_\times R, \tag{4}$$

where $[\![{}_B\boldsymbol{\omega}]\!]_\times$ is the skew-symmetric matrix notation of the crossproduct defined as

$$[\![{}_B\boldsymbol{\omega}]\!]_\times = \begin{bmatrix} 0 & -{}_B\omega_z & {}_B\omega_y \\ {}_B\omega_z & 0 & -{}_B\omega_x \\ -{}_B\omega_y & {}_B\omega_x & 0 \end{bmatrix}.$$

The dynamics of the vehicle can be described by the Newton-Euler equations

$$m\,{}_I\dot{\boldsymbol{v}} = R^\mathsf{T}\,{}_B\boldsymbol{f}_a(\boldsymbol{v}, \boldsymbol{\omega}) + R^\mathsf{T}\,{}_B\boldsymbol{f} + m\boldsymbol{g} \tag{5}$$

$$J\,{}_B\dot{\boldsymbol{\omega}} = -{}_B\boldsymbol{\omega} \times (J\,{}_B\boldsymbol{\omega}) + {}_B\boldsymbol{\tau}, \tag{6}$$

where $m \in \mathbb{R}$ is the vehicle mass, $J \in \mathbb{R}^{3\times3}$ is the mass moment of inertia matrix, ${}_B\boldsymbol{f} \in \mathbb{R}^3$ is the collective thrust vector produced by the propellers, ${}_B\boldsymbol{\tau} \in \mathbb{R}^3$ is the torque vector resulting from the differential thrust of the propellers, and ${}_B\boldsymbol{f}_a(\boldsymbol{v}, \boldsymbol{\omega}) \in \mathbb{R}^3$ is an aerodynamic force vector.

B. Measurement Models

The measurement models for the accelerometer and gyroscope are presented. Considering (5), we assume that the accelerometer can be modeled as follows

$$\begin{aligned} {}_B\boldsymbol{z}_{\text{acc}} &= \frac{1}{m}\left({}_B\boldsymbol{f}_a(\boldsymbol{v}, \boldsymbol{\omega}) + {}_B\boldsymbol{f}\right) + \boldsymbol{\eta}_{\text{acc}} \\ \boldsymbol{\eta}_{\text{acc}} &\sim \mathcal{N}(\boldsymbol{0}, \Sigma_{\text{acc}}), \end{aligned} \tag{7}$$

where $\boldsymbol{\eta}_{\text{acc}} \in \mathbb{R}^3$ is additive zero-mean Gaussian noise with a known covariance matrix Σ_{acc}. Furthermore, we assume that the gyroscope can be modeled as

$$\begin{aligned} {}_B\boldsymbol{z}_{\text{gyro}} &= {}_B\boldsymbol{\omega} + \boldsymbol{\eta}_{\text{gyro}} \\ \boldsymbol{\eta}_{\text{gyro}} &\sim \mathcal{N}(\boldsymbol{0}, \Sigma_{\text{gyro}}), \end{aligned} \tag{8}$$

where $\boldsymbol{\eta}_{\text{gyro}} \in \mathbb{R}^3$ is additive zero-mean Gaussian noise with a known covariance matrix Σ_{gyro}.

Note that we assume for both accelerometer and gyroscope, in contrast to other sensor models, cf. [13], that the measurements are unbiased. This is generally not the case for most IMU sensors used on quadcopters, but may for example be achieved by regular calibration. Alternatively, different sensors may be used in addition to determine the IMU biases by means of sensor fusion.

III. Aerodynamic Forces

In literature, it is often assumed that the dominant contribution to aerodynamic drag forces is the aerodynamic interaction between the air and the propellers, see for example [14]. A derivation of the aerodynamic forces and torques using blade element theory for a propeller can be found, for example, in [15]. These models were shown to be a reasonable approximation to the true drag forces in certain flight regimes. Unfortunately, if a vehicle is thrown by hand, it is generally far away from typical flight regimes: The vehicle may be translating and rotating at very fast rates. Therefore, we take a data-driven approach, and learn the aerodynamic model from experimental data.

To reduce the complexity of the aerodynamic model, the motors are commanded to spin at a constant, predefined speed, both while collecting data to determine an aerodynamic model, as well as in the experimental evaluation of our recovery method.

A. Model

We propose the following model for the aerodynamic force

$$\boldsymbol{f}_a(\boldsymbol{v}, \boldsymbol{\omega}) + \boldsymbol{f} = A\boldsymbol{v} + B\|\boldsymbol{v}\|\boldsymbol{v} + C\boldsymbol{\psi}(\boldsymbol{\omega}), \tag{9}$$

where $A \in \mathbb{R}^{3\times3}$, $B \in \mathbb{R}^{3\times3}$, and $C \in \mathbb{R}^{3\times10}$ are parameter matrices, and $\boldsymbol{\psi}(\boldsymbol{\omega}) \in \mathbb{R}^{10}$ is defined as

$$\boldsymbol{\psi}(\boldsymbol{\omega}) := (\omega_x, \omega_y, \omega_z, \omega_x^2, \omega_y^2, \omega_z^2, \omega_x\omega_y, \omega_y\omega_z, \omega_z\omega_x, 1). \tag{10}$$

Note that this model assumes constant propeller speeds, and thus incorporates a constant collective thrust $\boldsymbol{f}$ in the term $C\boldsymbol{\psi}(\boldsymbol{\omega})$. Furthermore, the model assumes absence of aerodynamic disturbances such as changing wind. The model consists of linear and quadratic terms in the vehicle's linear and angular velocity. Subsequently, we will use the following definition of the combined aerodynamic force

$$\boldsymbol{f}_c(\boldsymbol{v}, \boldsymbol{\omega}) := \boldsymbol{f}_a(\boldsymbol{v}, \boldsymbol{\omega}) + \boldsymbol{f}. \tag{11}$$

B. Model Learning

The parameter matrices A, B, and C are estimated using a least-squares approach. By defining an input feature vector

$$\boldsymbol{\phi}(\boldsymbol{v}, \boldsymbol{\omega}) := \begin{bmatrix}\boldsymbol{v}^\mathsf{T} & \|\boldsymbol{v}\|\boldsymbol{v}^\mathsf{T} & \boldsymbol{\psi}(\boldsymbol{\omega})^\mathsf{T}\end{bmatrix}^\mathsf{T}, \tag{12}$$

and by using (11), we can rewrite the aerodynamic model described in (9) to be linear in the feature vector according to

$$\boldsymbol{f}_c(\boldsymbol{v}, \boldsymbol{\omega}) = \Lambda\boldsymbol{\phi}(\boldsymbol{v}, \boldsymbol{\omega}), \tag{13}$$

where $\Lambda \in \mathbb{R}^{3\times16}$ is the combined parameter matrix defined as

$$\Lambda := \begin{bmatrix}A & B & C\end{bmatrix}.$$

Assuming that $\boldsymbol{v}$, $\boldsymbol{\omega}$, and $\boldsymbol{z}_{\text{acc}}$ can be experimentally obtained, an estimate for Λ can be found as the least-squares solution

$$\Lambda^* = \arg\min_{\Lambda} \sum_{i=1}^{n} (m\boldsymbol{z}_{\text{acc}}^i - \Lambda\boldsymbol{\phi}^i)^\mathsf{T}(m\boldsymbol{z}_{\text{acc}}^i - \Lambda\boldsymbol{\phi}^i), \tag{14}$$

where the superscript i indicates the training sample number, and n is the total number of training samples.

C. Experimental Data

In order to collect data to fit the aerodynamic model (13), the quadcopter was thrown in the Flying Machine Arena [16], a $10\,\text{m} \times 10\,\text{m} \times 10\,\text{m}$ flight space equipped with a Vicon motion capture system. During all experiments, all propellers were commanded to rotate at a constant idle speed. The result is a dataset consisting of measurements of the position and attitude of the vehicle, provided by the motion capture system at $200\,\text{Hz}$, and gyroscope and accelerometer measurements at $1000\,\text{Hz}$, along multiple parabolic trajectories. The position measurements were low-pass filtered, and numerically differentiated to obtain the velocity of the vehicle. Furthermore, both the velocity and attitude were linearly interpolated to match the sampling time of the IMU measurements. Using the attitude measurements, the velocity was then rotated to the body-fixed frame B. The dataset contains around $n = 20\,000$ samples.

IV. Gravity Estimation

This section formulates a Bayesian tracking problem and shows how to approximately solve it using a bank of EKFs.

A. Problem Formulation

Given a quadcopter in free fall with an arbitrary unknown initial attitude and an arbitrary unknown initial velocity, the goal is to determine an estimate of the direction of gravity at every time step k.

Under the assumption that the magnitude of the gravitational acceleration $g = \|\boldsymbol{g}\|$ is known, but its direction expressed in the coordinate frame B is a priori unknown, the gravity vector ${}_B\boldsymbol{g}_k$ can be modeled as a sample of a continuous random variable $\boldsymbol{G}$ with a uniform initial distribution on its sample space, i.e. a sphere $\mathcal{S}$ with radius g. As measurements $\boldsymbol{z}_k = (\boldsymbol{z}_{\text{acc},k}, \boldsymbol{z}_{gyro,k})$ become available, information about the distribution of $\boldsymbol{G}$ is obtained. The goal is to extract an estimate ${}_B\hat{\boldsymbol{g}}_k$ from the conditional distribution

$$p({}_B\boldsymbol{g}_k | \boldsymbol{z}_{1:k}), \tag{15}$$

where $\boldsymbol{z}_{1:k} = (\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_k)$ is the sequence of measurements from time step 1 up to time step k.

B. Bank of Extended Kalman Filters

In order to solve the Bayesian tracking problem recursively, we discretize the sample space of $\boldsymbol{G}$ by a set of finite realizations ${}_B\boldsymbol{g}_k^i$, allowing us to implement an EKF for every realization or hypothesis i, where every hypothesis is initially assigned a probability mass

$$p(i) = \frac{1}{N} \quad \forall i \in 1, 2, \ldots, N, \tag{16}$$

where N is the number of hypotheses. Note that this assumes that all possible realizations are evenly spread on the sphere $\mathcal{S}$. This may, for example, be achieved by choosing ${}_B\boldsymbol{g}_0^i$ to be the vertices of a platonic solid. In practice, for an arbitrary number of hypotheses N, we use numerical solutions to the so-called *spherical codes problem*, provided in [17].

C. Attitude and Velocity Estimation

Every one of the N EKF estimates the velocity and attitude of the vehicle. Attitude estimation is realized as described in [18], i.e. the attitude is described by a deterministic reference attitude $R_{\text{ref}}^i \in SO(3)$, and a stochastic error rotation vector $\boldsymbol{\delta}^i \in \mathbb{R}^3$ as

$$R^i = \exp([\![-\boldsymbol{\delta}^i]\!]_\times) R_{\text{ref}}^i, \tag{17}$$

where $\exp(\cdot)$ is the matrix exponential. The reference attitude at time $k = 0$ is initialized such that it satisfies the equality

$${}_B\boldsymbol{g}_0^i = R_{\text{ref},0}^i {}_I\boldsymbol{g}. \tag{18}$$

This may for example be done by computing the shortest-arc rotation between ${}_B\boldsymbol{g}_0^i$ and ${}_I\boldsymbol{g}$ as described, for example, in [19].

The aerodynamic model contained in the measurement model of the accelerometer (7) renders the velocities of the vehicle with respect to the surrounding air observable, and by estimating the velocity of the vehicle, the direction of

gravity can be inferred through the system dynamics (5). Note however, that the orientation of the vehicle around the direction of gravity is not observable, and thus only two components of the error rotation vector can be tracked. We therefore choose the third component of the rotation vector to be zero, such that

$$\boldsymbol{\delta}^i = (\delta_x^i, \delta_y^i, 0). \tag{19}$$

The stochastic state $\boldsymbol{x}_k^i \in \mathbb{R}^5$ of the i-th EKF at time step k consists of the velocity ${}_B\boldsymbol{v}_k^i$ of the vehicle expressed in the body-fixed coordinate frame, and the two components $\delta_{x,k}^i$ and $\delta_{y,k}^i$ of the error rotation vector $\boldsymbol{\delta}_k^i$. Furthermore, for every EKF we track its reference attitude $R_{\mathrm{ref},k}^i$.

We assume that the initial state is normally distributed around mean 0, with an initial covariance matrix P_0^i as

$$\boldsymbol{x}_0^i \sim \mathcal{N}(0, P_0^i). \tag{20}$$

Note that it is assumed that the gyroscope measurement noise is negligible in relation to the uncertainty of the state estimate such that we may choose

$${}_B\hat{\boldsymbol{\omega}}_k^i = \boldsymbol{z}_{\mathrm{gyro},k}. \tag{21}$$

1) Prior Update: Assuming a constant acceleration over the time interval ΔT, (5) can be discretized, and expressed in the body-fixed coordinate frame B yielding the prior update equation of the velocity

$$\begin{aligned} {}_B\hat{\bar{\boldsymbol{v}}}_k^i &= {}_B\hat{\boldsymbol{v}}_{k-1}^i + \\ &\Delta T \left(-{}_B\hat{\boldsymbol{\omega}}_{k-1}^i \times {}_B\hat{\boldsymbol{v}}_{k-1}^i + \frac{1}{m} {}_B\boldsymbol{f}_c + \hat{R}_{k-1}^i {}_I\boldsymbol{g}^i \right), \end{aligned} \tag{22}$$

where the bar on top of a symbol indicates that the quantity is obtained after the prior update step. Assuming a constant angular rate over the time interval ΔT, the kinematic relation (4) can be discretized and the reference attitude can directly be updated according to

$$\hat{\bar{R}}_{\mathrm{ref},k}^i = \exp\left([\![-{}_B\hat{\boldsymbol{\omega}}_{k-1}^i]\!]_\times \right) \hat{R}_{\mathrm{ref},k-1}^i, \tag{23}$$

resulting in a prior mean state estimate

$$\hat{\bar{\boldsymbol{x}}}_k^i = ({}_B\hat{\bar{\boldsymbol{v}}}_k^i, 0, 0). \tag{24}$$

The covariance matrix P_{k-1}^i is propagated according to

$$\bar{P}_k^i = F_{k-1}^i P_{k-1}^i \left(F_{k-1}^i\right)^\mathsf{T} + Q, \tag{25}$$

where F_{k-1}^i is the state-transition matrix of the linearized system model evaluated at $\left({}_B\hat{\boldsymbol{\omega}}_{k-1}^i, {}_B\hat{\boldsymbol{v}}_{k-1}^i\right)$ (see the Appendix), and Q is the process noise covariance matrix.

2) Posterior Update: The posterior update is computed as

$${}_B\boldsymbol{r}_k^i = {}_B\boldsymbol{z}_{\mathrm{acc},k} - \frac{1}{m} {}_B\boldsymbol{f}_c({}_B\hat{\bar{\boldsymbol{v}}}_k^i, {}_B\hat{\boldsymbol{\omega}}_{k-1}^i) \tag{26}$$

$$K_k^i = \bar{P}_k^i \left(H_k^i\right)^\mathsf{T} \left(H_k^i \bar{P}_k^i \left(H_k^i\right)^\mathsf{T} + \Sigma_{\mathrm{acc}} \right)^{-1} \tag{27}$$

$$\hat{\boldsymbol{x}}_k^i = \hat{\bar{\boldsymbol{x}}}_k^i + K_k^i {}_B\boldsymbol{r}_k^i, \tag{28}$$

where H_k^i is the linearization of the measurement equation (7) evaluated at ${}_B\hat{\bar{\boldsymbol{v}}}_k^i$ (see the Appendix), and Σ_{acc} is the measurement noise covariance matrix of the accelerometer. The state covariance matrix is updated according to

$$P_k^i = \left(\mathbb{I} - K_k^i H_k^i\right) \bar{P}_k^i. \tag{29}$$

Finally, the reference attitude is updated as

$$\hat{R}_{\mathrm{ref},k}^i = \exp([\![-\hat{\boldsymbol{\delta}}_k^i]\!]_\times) \hat{\bar{R}}_{\mathrm{ref},k}^i, \tag{30}$$

and the error rotation and the state covariance matrix are reset according to [18].

3) Hypothesis Likelihood: The posterior probability mass function of hypothesis i at time step k, given the measurement sequence $\boldsymbol{z}_{1:k}$, is computed using Bayes' law

$$p(i|\boldsymbol{z}_{1:k}) = \frac{p(\boldsymbol{z}_{1:k}|i)p(i)}{p(\boldsymbol{z}_{1:k})}, \tag{31}$$

where the likelihood $p(\boldsymbol{z}_{1:k}|i)$ is updated according to the recursion

$$p(\boldsymbol{z}_{1:k}|i) = p(\boldsymbol{z}_k|\boldsymbol{z}_{1:k-1}, i) p(\boldsymbol{z}_{1:k-1}|i). \tag{32}$$

The term $p(\boldsymbol{z}_k|\boldsymbol{z}_{1:k-1}, i)$ is given [20] as

$$p(\boldsymbol{z}_k|\boldsymbol{z}_{1:k-1}, i) = \frac{1}{\sqrt{2\pi|W_k^i|}} \exp\left[-\frac{1}{2} \left(\boldsymbol{r}_k^i\right)^\mathsf{T} \left(W_k^i\right)^{-1} \boldsymbol{r}_k^i \right], \tag{33}$$

where $|\cdot|$ is the matrix determinant, and $\boldsymbol{r}_k^i$ and $W_k^i = H_k^i \bar{P}_k^i \left(H_k^i\right)^\mathsf{T} + \Sigma_{\mathrm{acc}}$ are computed in the posterior update of the i-th EKF.

Note that the term $p(\boldsymbol{z}_{1:k})$ in (31) can be calculated by the law of total probability

$$p(\boldsymbol{z}_{1:k}) = \sum_{i=1}^{N} p(\boldsymbol{z}_{1:k}|i) p(i) \tag{34}$$

The output of the bank of EKF is chosen as the posterior mean $\boldsymbol{x}_k^{i^*}$ and posterior covariance $P_k^{i^*}$ of the most likely filter

$$i^* = \arg\max p(i|\boldsymbol{z}_{1:k}). \tag{35}$$

In a failure recovery scenario, this output may be passed to the estimator used for nominal operation after processing a sufficient amount of data.

4) Approximate Conditional Distribution of Gravity: In addition, an approximation to the conditional probability distribution $p({}_B\boldsymbol{g}|\boldsymbol{z}_{1:k})$ can be found as

$$p({}_B\boldsymbol{g}|\boldsymbol{z}_{1:k}) \approx \sum_{i=1}^{N} p({}_B\boldsymbol{g}|i) p(i|\boldsymbol{z}_{1:k}). \tag{36}$$

V. Experimental Setup

This section summarizes the experimental setup used to verify the performance of the proposed method for gravity estimation. The method is embedded into a complete state estimate recovery framework with the goal being to reinitialize an estimate of the vehicle state after being thrown by a human in an aggressive manner, allowing a controller to subsequently recover the vehicle into controlled flight. The experiments are conducted in the Flying Machine Arena,

where ground truth position and attitude measurements provided by the motion capture system are used for analysis and monitoring purposes only. All computation is performed onboard and the vehicle has no access to the motion capture data. The bank of EKFs is implemented with $N = 40$ hypotheses.

A. Recovery Logic

As in related work [11], full estimation and control is established sequentially. In particular, the vehicle is switched on while held by a human, and its propellers are commanded to spin at idle speed. Then, the vehicle is thrown, which is detected by the recovery logic if the following condition is true

$$\|\boldsymbol{z}_{\text{acc},k}\| < \beta_1, \tag{37}$$

where β_1 is the detection threshold, chosen as $\beta_1 = 7\,\text{m/s}^2$.

After a throw is detected, the gravity estimator is run until it has converged according to the condition

$$S_{\boldsymbol{g}} < \beta_2, \tag{38}$$

where $S_{\boldsymbol{g}}$ is a variable representing the variance of the gravity estimate, which we define as

$$S_{\boldsymbol{g}} := 1 - \frac{\left\| \sum_{i=1}^{N} p(i|\boldsymbol{z}_{1:k})_B \hat{\boldsymbol{g}}_k^i \right\|}{g}, \tag{39}$$

and β_2 is a threshold, chosen as $\beta_2 = 0.06$. This formulation is inspired by the spherical variance presented in [21]. Note that $0 \leq S_{\boldsymbol{g}} \leq 1$, where $S_{\boldsymbol{g}} = 0$ means that the estimator has converged to a single direction, and $S_{\boldsymbol{g}} = 1$ represents maximum uncertainty.

After the gravity estimator has converged, the estimator corresponding to the most likely hypothesis is determined using (35). Its posterior mean state estimate and covariance are then used to initialize an estimator and a controller intended for normal operation.

B. Estimation

After an estimate of the direction of gravity is obtained, an estimator similar to one single filter of the bank of EKFs presented in the previous section is initialized which differs only from the latter in the aerodynamic force model and in additionally using pressure measurements to update the estimate. We will refer to this estimator as the *nominal estimator* hereafter. The aerodynamic force model optimized for near hover conditions is described in detail in [14]. Since neither this aerodynamic force model nor the one described by (9) is designed to approximate the aerodynamic effects sufficiently accurately during an aggressive recovery maneuver, we disable accelerometer measurement updates as long as the attitude control error is greater than a threshold β_3, chosen as $\beta_3 = 15\,\text{deg}$.

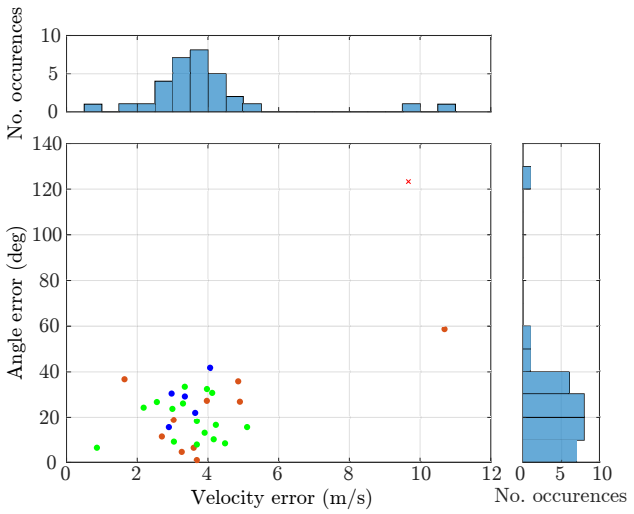

Fig. 1. Angle error between estimated and true direction of gravity, error magnitude between estimated and true velocity immediately after the gravity estimator has converged, and corresponding error histograms for 32 throw experiments: successful recovery (green), recovery to brief hover (blue), no successful recovery within the flight space (orange), and crash (red cross).

C. Control

The controller utilized in this work is a cascaded controller. The outermost loop is a velocity and altitude control loop which computes a desired acceleration in the inertial frame given the desired velocity and altitude and the estimates thereof as outlined in the appendix of [16]. The desired acceleration is then passed to the reduced attitude control loop which computes the desired angular velocity in the body-fixed coordinate frame as described in [19]. Finally, the innermost control loop tracks the desired angular velocity by adjusting the rotational speeds of the propellers as documented in the appendix of [16].

D. Quadcopter

The quadcopter utilized for all experiments has a mass $m = 0.5\,\text{kg}$, inertia matrix $J = \text{diag}(0.0023, 0.0023, 0.0046)\text{kgm}^2$, and arm length $L = 0.17\,\text{m}$. Computation required for flight is performed on a Snapdragon Flight board. Specifically, estimation and control described in sections V-B and V-C respectively, are performed on its digital signal processor, while gravity estimation is performed on its application processor. The bank of EKFs is updated at a frequency of $1000\,\text{Hz}$.

VI. Results

In this section, experimental results of the proposed method are presented. First, the results of an experimental robustness analysis are shown, and then a single experiment is analyzed in greater depth. A selection of successful experiments is shown in the accompanying video.

A. Robustness Analysis

In order to investigate the robustness of the proposed approach, we conducted 32 throw experiments where the

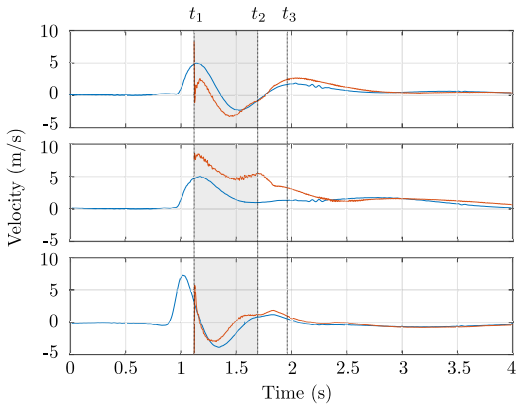

Fig. 2. Ground truth (red) and estimated (blue) velocity of the vehicle expressed in the body-fixed frame B during a single throw experiment. At t_1 a throw is detected, at t_2 the bank of EKFs has converged, and at t_3 nominal operation is entered.

quadcopter was programmed to recover and subsequently hover. Figure 1 shows the angle error between the true and the estimated direction of gravity, as well as the error magnitude between the true and the estimated velocity at the moment when the bank of EKFs converges, i.e. when (38) is true. In 16 cases, the vehicle was able to successfully recover its attitude and hover for at least 10 seconds, after which the experiment was manually terminated and the vehicle commanded to land. This corresponds to a success rate of 50%. In another 5 cases, the vehicle was able to hover for at least 3 seconds, but less than 10 seconds. We consider these cases to be successful, since the vehicle was able to recover its attitude, despite translating too much to remain within a safe flight zone within the Flying Machine Arena. In one case the vehicle crashed due to a poor attitude estimate. In the remaining 10 cases we were not able to conclude whether the vehicle might have converged if given more space. The IMU was calibrated on average after every seventh flight.

B. Detailed Analysis

In this section, a single successful experiment is analyzed in greater depth. Again, the vehicle is programmed to recover and subsequently hover after being thrown. Figure 4 shows an image sequence of the experiment and Figure 2 illustrates the ground truth and estimated velocity of the vehicle expressed in the body-fixed coordinate frame. At $t_1 = 1.12\,\mathrm{s}$ a throw is detected according to criterion (37). Overall, the estimated velocity has a similar waveform when compared to ground truth, but exhibits a rather large bias in the x and y directions of frame B. After $570\,\mathrm{ms}$, i.e. at $t_2 = 1.69\,\mathrm{s}$, the bank of EKFs has converged according to criterion (38), and the nominal estimator is initialized. At $t_3 = 1.96\,\mathrm{s}$ the attitude control error has dropped below the threshold β_3, and from this moment on the nominal estimator is updated with accelerometer measurements. The estimated velocities then converge towards the true velocities.

Figure 3 shows the angle error between the true and the estimated direction of gravity, and the hypothesis likelihood $p(i|\boldsymbol{z}_{1:k})$ for all 40 hypotheses between t_1 and t_2, with interesting graphs highlighted. The angle error of the most likely estimator, i.e. the one used to subsequently initialize the nominal estimator, quickly decreases to 30 to 50 degrees, while the bank of EKFs often switches between different hypotheses. Towards, t_2, the most likely filter settles at an angle error of about 10 degrees. Note that in this experiment the estimator achieving the smallest angle error is not the most likely estimator, while after about $150\,\mathrm{ms}$, the worst performing estimator in terms of angle error is also the least likely estimator.

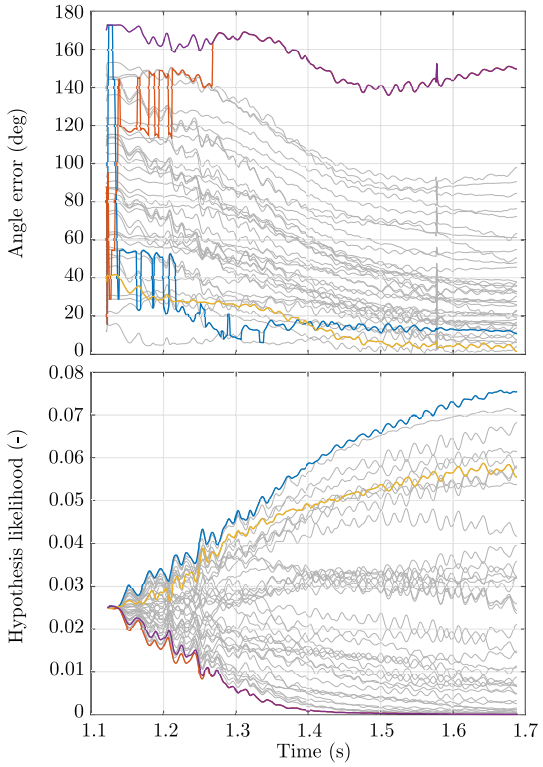

Fig. 3. Angle error between the estimated and true directions of gravity (top), and estimator likelihoods (bottom) for all 40 hypotheses during a single throw experiment. The most likely (blue) and least likely (red) estimator, and the estimators achieving the lowest (yellow) and largest (purple) angle error at t_2 are highlighted.

VII. Conclusion

We have proposed and experimentally evaluated a method that allows an aerial vehicle to autonomously reinitialize its state estimate without prior information utilizing only IMU measurements. The method was shown to work for more than 50% of the tested recovery actions, but also led to a crash. A promising approach to increase the robustness of the method might be to evaluate different aerodynamic force models, as this is the critical component of the method. Such aerodynamic models might for example include aerodynamic effects due to varying propeller speeds or downwash. By using measurements from more sensors, e.g. by employing an optical flow sensor, the robustness could be further increased

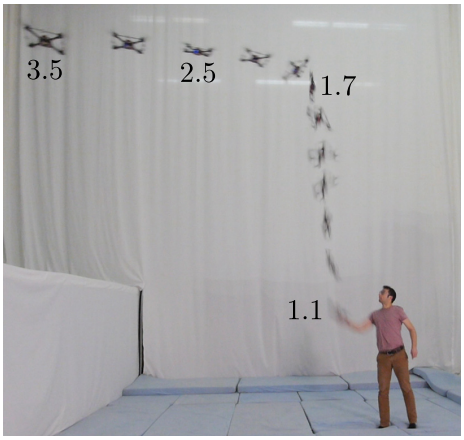

Fig. 4. An image sequence of the experiment discussed in VI-B and Figure 2, 3. The numbers indicate the time the image was taken in seconds.

and the dependence on unbiased IMU measurements could also be removed. Finally, the robustness and sensitivity to biases of the method could be studied in greater depth.

APPENDIX
LINEARIZATION

The state-transition matrix of the linearized system is

$$F_{k-1}^i = \mathbb{I} + \Delta T \begin{bmatrix} \Gamma_1 & \Gamma_2 \\ \Gamma_3 & \Gamma_4 \end{bmatrix}, \tag{40}$$

where

$$\Gamma_1 = -[\![{}_B\hat{\boldsymbol{\omega}}_{k-1}^i]\!]_\times + \frac{1}{m}\Upsilon_{k-1}^i \tag{41}$$

$$\Gamma_2 = \begin{bmatrix} 0 & -{}_B\hat{g}_{z,k-1}^i \\ {}_B\hat{g}_{z,k-1}^i & 0 \\ -{}_B\hat{g}_{y,k-1}^i & {}_B\hat{g}_{x,k-1}^i \end{bmatrix} \tag{42}$$

$$\Gamma_3 = \mathbb{O} \tag{43}$$

$$\Gamma_4 = \frac{1}{2}\begin{bmatrix} 0 & {}_B\hat{\omega}_{z,k-1}^i \\ -{}_B\hat{\omega}_{z,k-1}^i & 0 \end{bmatrix}, \tag{44}$$

with ${}_B\hat{\boldsymbol{g}}_{k-1}^i = \hat{R}_{k-1}^i {}_I\boldsymbol{g}$, $\mathbb{O}$ a matrix of zeros of appropriate size, and

$$\Upsilon_{k-1}^i = \left.\frac{\partial_B \boldsymbol{f}_c\left({}_B\boldsymbol{v}, {}_B\boldsymbol{\omega}\right)}{\partial_B \boldsymbol{v}}\right|_{{}_B\hat{\boldsymbol{v}}_{k-1}^i, {}_B\hat{\boldsymbol{\omega}}_{k-1}^i}$$
$$= \begin{cases} A + \frac{\left({}_B\hat{\boldsymbol{v}}_{k-1}^i {}_B\hat{\boldsymbol{v}}_{k-1}^{i\mathsf{T}} + \mathbb{I}\|{}_B\hat{\boldsymbol{v}}_{k-1}^i\|^2\right)}{\|{}_B\hat{\boldsymbol{v}}_{k-1}^i\|} & \text{if } {}_B\hat{\boldsymbol{v}}_{k-1}^i \neq \mathbb{O} \\ A & \text{otherwise.} \end{cases} \tag{45}$$

The linearization of the measurement model is

$$H_k^i = \begin{bmatrix} \frac{1}{m}\Upsilon_{k-1}^i & \mathbb{O} \end{bmatrix}. \tag{46}$$

ACKNOWLEDGMENTS

The authors would like to thank Marc-Andrè Corzillius for the help in designing the electrical hardware. Many people have contributed to the Flying Machine Arena in which this research was conducted. A list of these people can be found at: `http://flyingmachinearena.org/people`.

REFERENCES

[1] 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, August 2016.

[2] A. Freddi, S. Longhi, A. Monteriù, and M. Prist, "Actuator fault detection and isolation system for an hexacopter," in *2014 IEEE/ASME 10th International Conference on Mechatronic and Embedded Systems and Applications*, Sept 2014, pp. 1–6.

[3] T. Schneider, G. Ducard, K. Rudin, and P. Strupler, "Fault-tolerant control allocation for multirotor helicopters using parametric programming," in *International Micro Air Vehicle Conference and Flight Competition 2012*, July 2012.

[4] M. Kamel, K. Alexis, M. Achtelik, and R. Siegwart, "Fast nonlinear model predictive control for multicopter attitude tracking on SO(3)," in *2015 IEEE Conference on Control Applications*, Sept. 2015, pp. 1160–1166.

[5] A. Nemati, R. Kumar, and M. Kumar, "Stability and control of tilting-rotor quadcopter in case of a propeller failure," in *Proceedings of the ASME Dynamic Systems and Control Division*, October 2016.

[6] C. Berbra, S. Lesecq, and J. J. Martinez, "A multi-observer switching strategy for fault-tolerant control of a quadrotor helicopter," in *2008 16th Mediterranean Conference on Control and Automation*, June 2008, pp. 1094–1099.

[7] Y. Zhang, A. Chamseddine, C. Rabbath, B. Gordon, C.-Y. Su, S. Rakheja, C. Fulford, J. Apkarian, and P. Gosselin, "Development of advanced fdd and ftc techniques with application to an unmanned quadrotor helicopter testbed," *Journal of the Franklin Institute*, vol. 350, no. 9, pp. 2396–2422, 2013.

[8] 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, July 2016.

[9] H. E. Rauch, "Intelligent fault diagnosis and control reconfiguration," *IEEE Control Systems*, vol. 14, no. 3, pp. 6–12, June 1994.

[10] A. Freddi, S. Longhi, and A. Monteriù, "A Diagnostic Thau Observer for a Class of Unmanned Vehicles," *Journal of Intelligent & Robotic Systems*, vol. 67, no. 1, pp. 61–73, July 2012.

[11] M. Faessler, F. Fontana, C. Forster, and D. Scaramuzza, "Automatic re-initialization and failure recovery for aggressive flight with a monocular vision-based quadrotor," in *2015 IEEE International Conference on Robotics and Automation*, May 2015, pp. 1722–1729.

[12] C. Stracquodaine, A. Dolgikh, M. Davis, and V. Skormin, "Unmanned Aerial System security using real-time autopilot software analysis," in *2016 International Conference on Unmanned Aircraft Systems*, June 2016, pp. 830–839.

[13] S. Weiss, R. Brockers, S. Albrektsen, and L. Matthies, "Inertial optical flow for throw-and-go micro air vehicles," in *IEEE Winter Conference on Applications of Computer Vision*, Jan 2015, pp. 262–269.

[14] M. W. Mueller, M. Hamer, and R. D'Andrea, "Fusing ultra-wideband range measurements with accelerometers and rate gyroscopes for quadrocopter state estimation," in *2015 IEEE International Conference on Robotics and Automation*, May 2015, pp. 1730–1736.

[15] R. Gill and R. D'Andrea, "Propeller thrust and drag in forward flight," in *2017 IEEE Conference on Control Technology and Applications*, Aug 2017, pp. 73–79.

[16] 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.

[17] N. J. A. Sloane, with the collaboration of R. H. Hardin, W. D. Smith and others, "Spherical codes," published electronically at http://NeilSloane.com/packings/.

[18] M. W. Mueller, M. Hehn, and R. D'Andrea, "Covariance correction step for kalman filtering with an attitude," *Journal of Guidance, Control, and Dynamics*, 2016.

[19] D. Brescianini, M. Hehn, and R. D'Andrea, "Nonlinear quadrocopter attitude control," Tech. Rep., 2013.

[20] W. S. Chaer, R. H. Bishop, and J. Ghosh, "A mixture-of-experts framework for adaptive kalman filtering," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 27, no. 3, pp. 452–464, Jun 1997.

[21] K. V. Mardia, *Statistics of Directional Data*. Academic Press, 1972.