Received: 30 January 2022 | Revised: 27 June 2023 | Accepted: 30 July 2023

DOI: 10.1002/rnc.6935

RESEARCH ARTICLE

WILEY

# System identification for high-performance UAV control in wind

**Animesh K. Shastry**[1] | **Derek A. Paley**[1,2]

[1]Department of Aerospace Engineering, University of Maryland, College Park, Maryland, USA

[2]Institute for Systems Research, University of Maryland, College Park, Maryland, USA

**Correspondence**
Animesh K. Shastry and Derek A. Paley, Department of Aerospace Engineering, University of Maryland, College Park, MD 20742, USA.
Email: animeshs@umd.edu and dpaley@umd.edu

**Funding information**
US Army, Grant/Award Number: W911W61720004

**Summary**

This article describes and experimentally evaluates a comprehensive system identification framework for high-performance UAV control in wind. The framework incorporates both linear offline and nonlinear online methods to estimate model parameters in support of a nonlinear model-based control implementation. Inertial parameters of the UAV are estimated using a frequency-domain linear system identification program by incorporating control data obtained from motor-speed sensing along with state estimates from an automated frequency sweep maneuver. The drag-force coefficients and external wind are estimated recursively in flight with a square-root unscented Kalman filter. A custom flight controller is developed to handle the computational demand of the online estimation and control. Flight experiments illustrate the nonlinear controller's tracking performance and enhanced gust rejection capability.

**KEYWORDS**

adaptation models, estimation, Kalman filter, parameter estimation, system identification, uncertain systems

## 1 | INTRODUCTION

An unmanned aerial vehicle (UAV) relies on the performance of its flight controller's estimation and control algorithms to reconstruct its state and track reference trajectories. The conventional approach for implementing stable and reliable controllers for a UAV involves a trial-and-error procedure, where control gains are manually adjusted based on flight performance and the pilot's experience. Linear controllers such as the proportional–integral–derivative controller (PID) are almost universally used in robotics and aerospace control systems.[1,2] Commercially available quadrotors and flight control firmware[3] such as PX4, Ardupilot, and Betaflight implement cascaded PID control for stabilization. However, there are two major drawbacks to linear control: (1) performance degrades in some situations where tracking precision is required due to the linearity assumption; and (2) gain tuning can be time-consuming and requires the presence of an experienced pilot. Furthermore, it is impractical to tune PID controllers for UAVs that pick up and carry payloads or packages of unknown size and weight.

A modern approach for UAV control utilizes model-based nonlinear control laws to ensure global stability of the vehicle. Feedback linearization controllers, also known as dynamic inversion controllers, achieve high control performance and have been extensively researched.[4,5] The state-of-the-art for nonlinear quadrotor control are adaptive control[6] and geometric control,[7] which have been widely used to demonstrate aggressive and agile flights. However, a major

drawback to a model-based nonlinear control is that its performance depends on the accuracy of the model parameters.[5] The same problem affects UAV state estimation. For example, the Kalman filter is the state-of-the-art, but as a model-based framework, its performance is also limited by the accuracy of the dynamic model.[8] Widely accepted solutions to parameter uncertainty are to either directly measure system parameters before flight or use a system identification process in a controlled setting. These existing solutions can often be costly, risky, and/or require specialized equipment and experiments,[9,10] making them undesirable for rapid development of high-performance UAVs.

The system identification process involves utilizing flight testing data to develop a dynamic model of the aircraft. Linear methods involve measuring sensor outputs of the aircraft in response to inputs and computing the state-space representation of the aircraft. For a typical UAV, the inputs are the commands generated by the controller and the outputs are obtained from the state estimator. Using commanded control inputs is beneficial in settings where inertial characteristics of the system do not need to be precisely estimated. If the end goal is the implementation of a linear controller, then knowledge of the state-space model is more useful than inertial parameters. On the contrary, if a nonlinear framework is desired, then inertial parameters need to be estimated, which is difficult to extract from the identified model since it contains the inertial characteristics mixed in with the speed controller's response and structural characteristics of the rotor.[11] The state-of-the-art industry-standard in rotorcraft frequency-domain linear system identification is the Comprehensive Identification from FrEquency Response (CIFER®)[12] program developed by the Ames Research Center. The CIFER® tool is widely used for aircraft and rotorcraft system identification,[13] including quadrotors; however, it is almost exclusively used for linear controller implementations.[14,15] One novelty of this article is that it improves the inertial parameter estimation accuracy of the linear system identification technique by directly sensing the individual motor revolutions per minute (RPM) to get better values for the control inputs than a traditional approach. The nonlinear dynamic model parameters are extracted from the resulting linearized model and used for the nonlinear control implementation. We augment offline linear system identification of parameters with online recursive estimation of them.

Nonlinear approaches towards system identification rely on parameter estimation models that indirectly select frequencies based on parametrized trajectories, use recursive estimation to identify all parameters simultaneously, and achieve weak observability almost globally. Notably, a multi-rotor often loses observability in hover, a flight condition typically used in system identification. Recursive estimation of mass and inertia parameters have been demonstrated using only onboard sensor data with an unscented Kalman filter (UKF),[16] but the estimates in our testing were observed to be less accurate than frequency-domain system identification methods. Parameters related to aerodynamic effects have higher observability when the aircraft operates away from hover conditions, such as high-speed maneuvers and aerobatic flight, especially using onboard sensing of inertial velocity via visual odometry. Alternate approaches for parameter estimation include adaptive control[17] and probabilistic optimization.[18] Notable developments include recent developments of adaptive nonlinear dynamic inversion control[19-21] and adaptive recursive orthogonal least-squares[22] frameworks. This article adopts a square-root unscented Kalman filter (sq-UKF) framework due to its ease of development and lower computational requirements than existing alternatives.

The state and nonlinear parameter estimation algorithm utilized here is based on the authors' prior work[23] in which a framework for online state, parameter, and wind estimation was developed for a UAV equipped with an IMU and a ground-velocity sensor, such as a visual-inertial odometry camera. The framework was previously evaluated in simulation using a sq-UKF. The work done in Reference 23 was primarily theoretical and included simulation results only, whereas the work done in this article builds upon it and demonstrates its feasibility through real-world experiments. Control inputs steered the process model and the ground-velocity and IMU measurements updated the state and parameter estimates. This article demonstrates experimentally the estimation performance of a subset of the estimator's state vector. Additionally, the aerodynamic drag torque expression is simplified, assuming zero sensor bias drift. A model-based feedback linearization controller operates on the Special Euclidean Group SE(3)[7] with necessary modifications to include aerodynamic interactions.[23] Nominal values for mass and inertia are obtained from a weight scale and a CAD model, respectively. Instead of manually tuning the position and orientation control gains, they were obtained by setting the damping ratio to 0.75 and natural frequency to 2 for translation control and 13 for orientation control. The gain values in simulations of prior work[23] also follow the same method, reinforcing the idea that manual tuning is not required if inertial parameters are known. Inertial parameters are estimated offline with a linear system identification process, whereas the aerodynamic parameters including drag force coefficients and wind are estimated online in a recursive manner with sq-UKF. The resulting parameter estimates are utilized by the nonlinear controller to achieve high control performance. The controller assumes zero wind and drag force coefficients to generate flight data for the linear system identification process. For nonlinear estimation and control experiments, the controller uses the drag force coefficients and wind estimates generated recursively by the sq-UKF estimator in flight.

The contributions of this article are (1) synthesizing linear and nonlinear estimation methods from robotics, rotorcraft, and flight dynamics and control to accurately estimate model parameters including mass, inertia, drag force coefficients, and external wind; (2) improving the inertial parameter estimation accuracy of linear system identification through direct sensing of the control inputs; and (3) development of a custom, low-cost flight controller capable of handling the high computational demands of nonlinear control and high-dimensional, sampling-based estimation. Through the use of a vision-based localization camera, the system's dependency on external positioning is eliminated, which reduces the identification processes' time and cost without sacrificing accuracy. The significance of these contributions lies in enabling rapid deployment of nonlinear frameworks for high-performance unmanned flight vehicles while eliminating the need for a manual tuning process and the presence of an experienced pilot.

The remainder of this article is structured as follows. Section 2 describes the aircraft and corresponding hardware and software components that enable real-time implementation of nonlinear estimation and control. Section 3 presents a nonlinear aircraft dynamics model along with a control allocation and propulsion model. A summary of parameters and their estimation methods is also provided. Section 4 describes the linearization of the aircraft dynamics and presents the estimates of the model parameters extracted from frequency-domain linear system identification. Section 5 illustrates the recursive, in-flight estimation of drag force coefficients and wind gusts through the use of a nonlinear state and parameter estimation and control framework. Finally, Section 6 summarizes the article and ongoing work.

## 2 | SYSTEM DESCRIPTION

A quadrotor with customized hardware and software was used in this research. This section presents an overview of the aircraft's hardware components, avionics, and software development.

### 2.1 | Hardware overview

The aircraft shown in Figure 1 is a squashed-X quadrotor with a 230 mm wheelbase. A dummy payload in the shape of a cuboid is attached to the bottom to introduce unmodeled dynamic effects and additional disturbance so that the proposed framework's performance can be evaluated in a worst-case scenario. Propulsion and avionics components are assembled onto a freestyle carbon fiber airframe. A pair of clockwise and counter-clockwise rotating Gemfan 5043 propellers are driven by Lumenier ZIP 2207 2450 kV brushless direct-drive electric motors, each with 12 electromagnets on the stator and 14 N52SH curved neodymium permanent magnets on the rotor. The motors are controlled by electronic speed controllers (ESC) running KISS firmware, and communication with the ESC is established using DShot600 digital signals

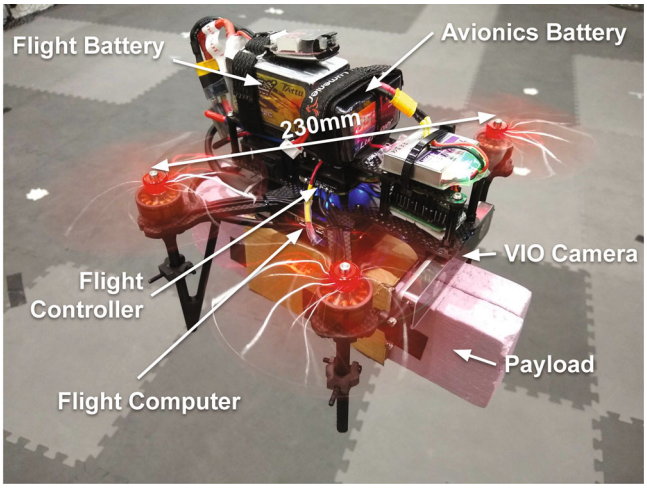


FIGURE 1 The quadrotor platform.

generated using direct memory access (DMA). The ESC is powered by a 4 cell (14.8 V) 2000 mAh lithium polymer battery. Avionics include a custom-built flight controller board, Intel Realsense T265 Tracking Camera, and a Raspberry Pi 4B flight computer. The T265 Tracking Camera enables vision-based localization for the quadrotor; eliminates the requirement for a motion-capture system or a Global Positioning System (GPS) for localization, thus reducing the system's cost, complexity, and setup time.

Commercially available, off-the-shelf, open-source flight controllers[3] such as Pixhawk, PixRacer, Sparky2, Chimera, Atom, APM 2.8, FlyMaple, PXFMini, and so forth, have insufficient computing power for implementation of modern nonlinear control and estimation algorithms that are increasingly using sampling and optimization-based techniques. The recursive online and onboard estimation scheme described in Reference 23 is high-dimensional and takes more computational load as compared to the controller. A backup estimator also needs to run in parallel so that in case the framework diverges, the software will choose the estimates provided by the backup estimator. The framework also needs to support a high receiving rate of visual-inertial odometry data; hence, the hardware will need higher cache memory. Additionally, the framework developed in this article required testing of many different motor speed measurement techniques and utilizing off-the-shelf flight controllers will restrict the freedom of choice as they haven't been designed for research or experimentation. Moreover, modifying the estimator framework of open-source firmware such as PX4 and Ardupilot comes with a very steep learning curve. Hence, seeking a custom approach, we designed a flight controller board with a focus on cost-effectiveness, high performance, crash resistance, and modularity, making it suitable for a wide class of UAVs.

Figure 2 shows the custom flight controller module, which carries the Teensy 4.0 Micro-Controller Unit (MCU) on the top and the ICM-20948 Inertial Measurement Unit (IMU) on the bottom. The Teensy 4.0 is presently the fastest available MCU; it contains an NXP IMXRT1062 processor with a 32-bit ARM Cortex-M7 CPU core, making it capable of executing instructions at 600 MHz. The high computing power of the MCU allows for the implementation of computationally demanding nonlinear estimation and control laws such as the high-dimensional sq-UKF and SE(3) control. The flight control software, when configured with the control and estimation laws mentioned in Section 1 runs seamlessly at 120 Hz on the MCU. Higher computational speeds are also possible by using faster communication protocols and by overclocking the CPU to a maximum speed of 1 GHz. The design files of the flight controller[24] have been made open-source by the authors.

## 2.2 | Software overview

The Teensy 4.0 MCU is integrated with the Arduino Integrated Development Environment (IDE), which facilitated the development of a user-friendly flight control software geared towards students and researchers. Through the use of C++ templates and matrix software utilities, standard operators are overridden to be used naturally in algebraic expressions, which results in simplistic Matlab-type syntaxes. Figure 3 shows a simplified block diagram of the overall software architecture. The system is designed to work fully onboard with negligible intervention from the operator. The flight control software[25] has been made open-source by the authors.

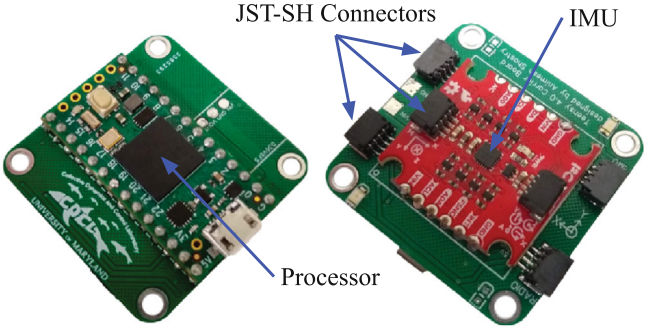


**FIGURE 2** The custom-made flight controller module. (Left) the top side; (right) the bottom. The module measures $37 \times 36 \times 6$ mm$^3$ and has the standardized $30.5 \times 30.5$ mm$^2$ mounting with JST-SH connectors.

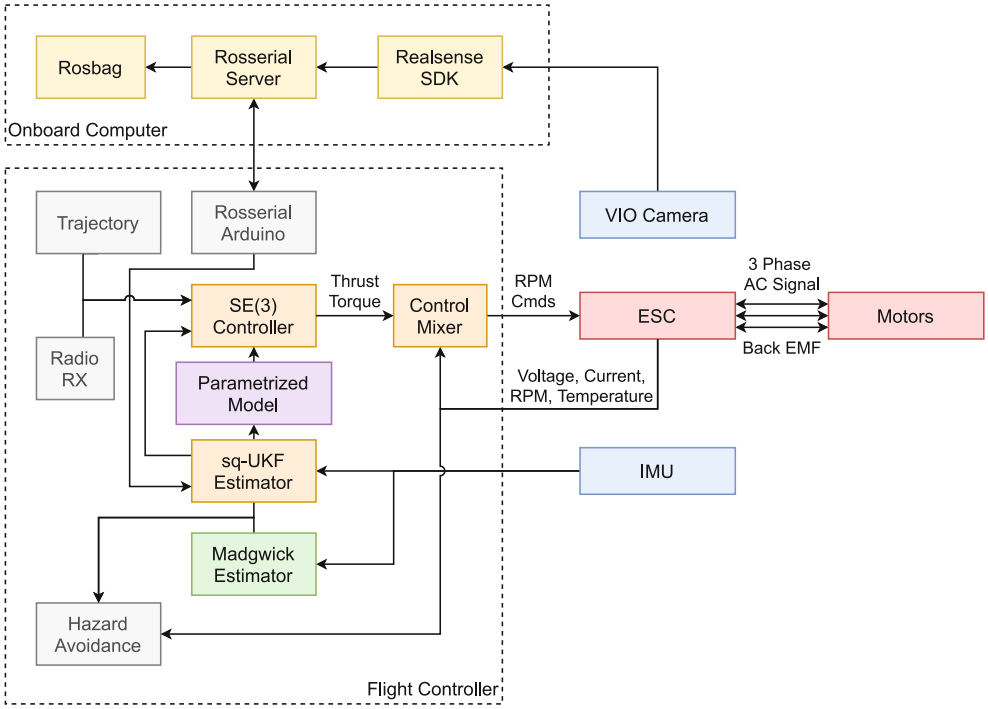


**FIGURE 3** Software architecture overview.

## 3 | SYSTEM MODELING

This section describes the mathematical models utilized by the UAV estimation, control, and system identification frameworks. Although the models described here are primarily applicable to quadrotors, they can be modified to suit other UAVs, rotorcrafts, and fixed-wing vehicles.

### 3.1 | Notation and conventions

A right-handed orthonormal inertial frame represented by $\mathcal{W} = \{\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3\}$ is defined such that $\mathbf{w}_3$ points opposite to the direction of gravity $g$. A body-fixed frame represented by $\mathcal{B} = \{\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3\}$ is attached to the vehicle, such that $\mathbf{b}_1$ points forward, $\mathbf{b}_3$ is aligned with the collective thrust, and $\mathbf{b}_2 = \mathbf{b}_3 \times \mathbf{b}_1$ completes the orthonormal triad. Figure 4 shows the inertial frame and body-fixed frames of reference. The orientation of the vehicle is represented by a rotation matrix $R \in \mathrm{SO}(3)$, where $[R]_{ij} = \mathbf{w}_i \cdot \mathbf{b}_j$. Additionally, the superscript notation $\mathbf{v}^{\mathcal{F}}$ represents the components of $\mathbf{v}$ in frame $\mathcal{F}$. Also, the convention for representing the set of standard basis vectors is $\{\mathbf{e}_i\}$, where $\mathbf{e}_i$ is a three-dimensional unit vector.

### 3.2 | Aircraft dynamics model

The quadrotor aircraft is modeled as a rigid body with gravity, motor thrust, and aerodynamic forces acting on the aircraft's center of mass. The translational and rotational kinematics and dynamics of the aircraft are

$$\begin{aligned}
\dot{\mathbf{r}}^{\mathcal{W}} &= \mathbf{v}^{\mathcal{W}} \\
\dot{\mathbf{v}}^{\mathcal{B}} &= \frac{1}{m}\left(T\mathbf{e}_3 + \mathbf{F}_a^{\mathcal{B}}\right) - gR^{\mathsf{T}}\mathbf{e}_3 - \boldsymbol{\omega}^{\mathcal{B}} \times \mathbf{v}^{\mathcal{B}} \\
\dot{R} &= R\left[\boldsymbol{\omega}^{\mathcal{B}}\right]_{\times} \\
\dot{\boldsymbol{\omega}}^{\mathcal{B}} &= J^{-1}\left(\boldsymbol{M} - \boldsymbol{\omega}^{\mathcal{B}} \times J\boldsymbol{\omega}^{\mathcal{B}} + \boldsymbol{\tau}_a^{\mathcal{B}}\right)
\end{aligned} \tag{1}$$

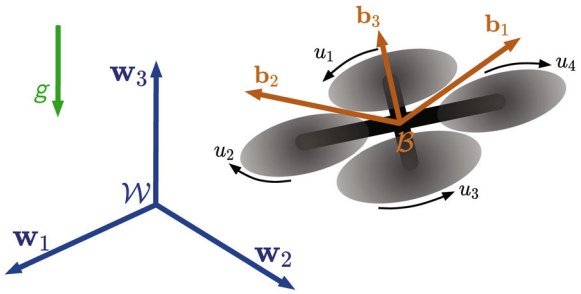


FIGURE 4 Reference frames.

where $\mathbf{r}^{\mathcal{W}}$ is the position of the vehicle in $\mathcal{W}$, $\mathbf{v}^{\mathcal{B}} := [v_x\ v_y\ v_z]^{\mathsf{T}} \in \mathbb{R}^3$ is the inertial velocity of the vehicle in $\mathcal{B}$, $\boldsymbol{\omega}^{\mathcal{B}} := [\omega_x\ \omega_y\ \omega_z]^{\mathsf{T}} \in \mathbb{R}^3$ is the inertial angular velocity of the vehicle in $\mathcal{B}$, $m$ is the vehicle's mass, $J \in \mathbb{R}^{3\times3}$ is the vehicle's inertia tensor, $T \in \mathbb{R}^1$ is the thrust generated by the motors, $\boldsymbol{M} \in \mathbb{R}^3$ is the moment generated by the motors, $\mathbf{F}_a^{\mathcal{B}}$ is the aerodynamic force experienced by the vehicle in $\mathcal{B}$, and $\boldsymbol{\tau}_a^{\mathcal{B}}$ is the aerodynamic torque experienced by the vehicle in $\mathcal{B}$.

The aerodynamic force components $\mathbf{F}_{a,i}^{\mathcal{B}}\ \forall\ i \in \{1,2,3\}$ are modeled as

$$\mathbf{F}_{a,i}^{\mathcal{B}} = -C_{D_i}\left|\left(\mathbf{v}^{\mathcal{B}} - \mathbf{v}_a^{\mathcal{B}}\right)\cdot\mathbf{e}_i\right|\left(\mathbf{v}^{\mathcal{B}} - \mathbf{v}_a^{\mathcal{B}}\right)\cdot\mathbf{e}_i, \tag{2}$$

where $\mathbf{C}_D^{\mathcal{B}} := [C_{D_1}\ C_{D_2}\ C_{D_3}]^{\mathsf{T}} \in \mathbb{R}^3$ is the vector of body drag force coefficients, and $\mathbf{v}_a^{\mathcal{B}}$ is the wind velocity in $\mathcal{B}$. The wind is assumed to be static in $\mathcal{W}$ and, hence, its evolution in $\mathcal{B}$ is described by

$$\dot{\mathbf{v}}_a^{\mathcal{B}} = -\boldsymbol{\omega}^{\mathcal{B}} \times \mathbf{v}_a^{\mathcal{B}} \tag{3}$$

The aerodynamic torque components $\boldsymbol{\tau}_{a,i}^{\mathcal{B}}\ \forall\ i \in \{1,2,3\}$ are modeled as

$$\boldsymbol{\tau}_{a,i}^{\mathcal{B}} = -C_{M_i}\left(\mathbf{b}_3 \times \left(\mathbf{v}^{\mathcal{B}} - \mathbf{v}_a^{\mathcal{B}}\right)\right)\cdot\mathbf{e}_i, \tag{4}$$

where $\mathbf{C}_M^{\mathcal{B}} := [C_{M_1}\ C_{M_2}\ 0]^{\mathsf{T}} \in \mathbb{R}^3$ is the vector of body drag moment coefficients.

## 3.3 | Control allocation model

The thrust and moment forces generated by the motors for a quadrotor in squashed-X configuration are modeled as Reference 26

$$\begin{bmatrix} T \\ \boldsymbol{M} \end{bmatrix} = \begin{bmatrix} k_f & k_f & k_f & k_f \\ k_f L_y & k_f L_y & -k_f L_y & -k_f L_y \\ -k_f L_x & k_f L_x & k_f L_x & -k_f L_x \\ -k_m & k_m & -k_m & k_m \end{bmatrix} \begin{bmatrix} u_1^2 \\ u_2^2 \\ u_3^2 \\ u_4^2 \end{bmatrix}, \tag{5}$$

where $k_f$ is the thrust coefficient for the propellers, $k_m$ is the moment coefficient for the propellers, $L_x$ and $L_y$ are the moment arm lengths, and $u_i$ is the speed of the $i$th motor. Assume that both the forces and moments generated by the motors are proportional to the square of the speed of the motors.

## 3.4 | Propulsion model

The quadrotor's electric brushless motors are controlled by varying the voltage amplitude of the active phases of the individual coils. The rotor torque $\tau_{z_i}$ and the applied voltage $V_i$ of the $i$th motor is modeled by Reference 27

$$V_i = K_e u_i + R_a I_i$$
$$\tau_{z_i} = K_q I_i, \tag{6}$$

where $I_i$ is the current through the $i$th motor, $R_a$ is its electrical resistance, and $K_e$ and $K_q$ are motor parameters. Since the torque is assumed to be proportional to the square of the rotor speed, the expression for the applied voltage as a function of motor speed is

$$V_i = \frac{R_a k_m}{K_q} u_i^2 + K_e u_i \tag{7}$$

To compensate for the motor friction, a constant voltage $V_0$ is added and the final expression for the applied voltage is

$$V_i = V_{calib}\left(K_2 u_i^2 + K_1 u_i\right) + V_0, \tag{8}$$

where $V_{calib}$ is the calibration voltage at which the constants $K_2 = \frac{R_a k_m}{K_q V_{calib}}$ and $K_1 = \frac{K_e}{V_{calib}}$ are calculated. The parameters $V_0$, $K_1$, and $K_2$ can be obtained either from manufacturer data for common motor-propeller setups or from calibration experiments such as a motor bench test for custom setups. Finally, as the electric motors are controlled by Pulse-Width Modulation (PWM) signals, their values can be obtained by

$$PWM_i = \frac{V_i}{V_{battery}} \tag{9}$$

where $PWM_i$ is the PWM setting for the $i$th motor and $V_{battery}$ is the total voltage of the battery pack. Typically, voltages of lithium-based batteries and other battery types decrease as the battery drains and (9) will maintain the motor speed at desired levels by increasing the PWM setting, thus ensuring consistent flight performance. Additionally, to minimize the influence of sharp voltage changes and sensing noise on the vehicle's control performance, a digital low-pass filter is added to the battery voltage sensor readings.

## 3.5 | Model parameter estimation

The parameters of the aircraft dynamic model (1) are estimated collectively from linear system identification and nonlinear parameter estimation techniques. The wind is treated as a time-varying parameter. For verification purposes, beliefs of true values for the parameters were obtained from independent experiments, CAD modeling, and specialized tools. Table 1 lists the parameters used in the aircraft dynamics model and their respective estimation method(s). The parameters used in the control allocation and propulsion model were obtained by mounting the motor-propeller setup on a thrust stand and measuring the thrust, torque, and RPM values at various PWM settings. Figure 5 shows the experimental data from the thrust stand and fitted propulsion and control allocation models. The calibration voltage is 15 V, and each data point is recorded by setting the PWM at a fixed value and averaging 500 measurements after the motor RPM reaches a steady state. Table 2 lists the parameters used in the control allocation and propulsion model along with their measured values.

**TABLE 1** Estimation techniques for the parameters of the aircraft dynamics model.

| Parameter name | Symbol | Units | Linear system ID | Nonlinear estimation | True value source |
|---|---|---|---|---|---|
| Mass | $m$ | kg | ✓ | | Weight scale, CAD |
| Inertia | $J$ | kg·m$^2$ | ✓ | | CAD |
| Drag moment coefficient | $\mathbf{C}_M^{\mathcal{B}}$ | N·s | ✓ | | — |
| Drag force coefficient | $\mathbf{C}_D^{\mathcal{B}}$ | N·s$^2$/m$^2$ | ✓ | ✓ | Gust generation system |
| Wind | $\mathbf{v}_a^{\mathcal{B}}$ | m/s | | ✓ | Anemometer |

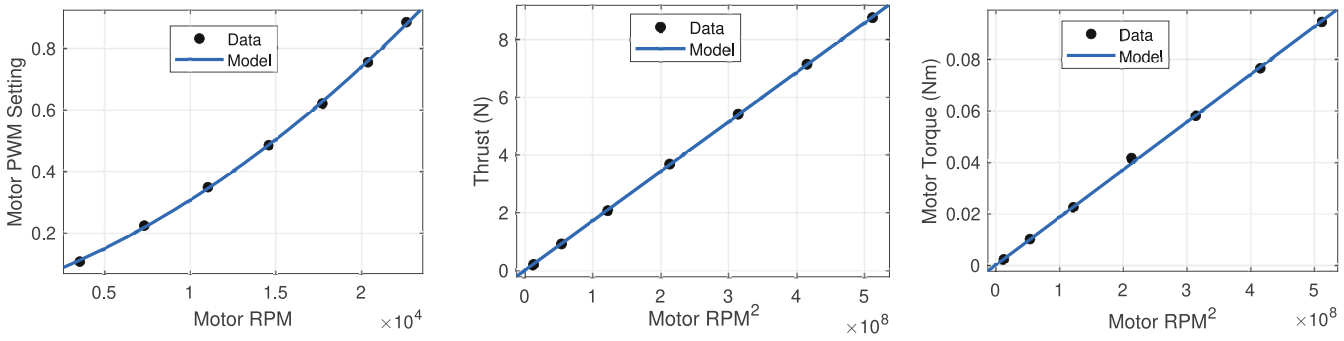


**FIGURE 5** Propulsion and control allocation models extracted from thrust-stand data.

**TABLE 2** Control allocation and propulsion model parameters.

| Parameter name | Symbol | Value | Units |
|---|---|---|---|
| Propeller thrust coefficient | $k_f$ | $1.717\times10^{-8}$ | N/RPM$^2$ |
| Propeller moment coefficient | $k_m$ | $1.845\times10^{-10}$ | N·m/RPM$^2$ |
| Motor constant voltage | $V_0$ | 0.4575 | V |
| Motor model linear coefficient | $K_1$ | $1.975\times10^{-5}$ | RPM$^{-1}$ |
| Motor model quadratic coefficient | $K_2$ | $7.835\times10^{-10}$ | RPM$^{-2}$ |

## 4 | LINEAR SYSTEM IDENTIFICATION

This section linearizes the nonlinear dynamic model about hover and performs frequency-domain system identification. Two sets of control data using different methods are collected for each flight, and the resulting model-parameter accuracies are compared.

### 4.1 | Model description

Before deriving the linearized model of the aircraft, certain assumptions are made so that the final state-space model resembles a standard rotorcraft model. Assumptions include negligible cross diagonal inertia terms, no wind, that is, $\mathbf{v}_a^{\mathcal{W}} = 0$, and a simplified linear drag force model. As a consequence, the models presented in (2) and (4) become

$$\begin{aligned}\mathbf{F}_{a,i}^{\mathcal{B}} &= -\overline{C}_{Di}\mathbf{v}^{\mathcal{B}}\cdot\mathbf{e}_i\\ \boldsymbol{\tau}_{a,i}^{\mathcal{B}} &= -C_{M_i}\left(\mathbf{b}_3\times\mathbf{v}^{\mathcal{B}}\right)\cdot\mathbf{e}_i\end{aligned} \tag{10}$$

where $\overline{C}_{Di}$ are the body-drag force coefficients for the simplified linear aerodynamic force model.

The nonlinear aircraft model presented in Section 3.2 is linearized about hover and the resulting quadrotor dynamics decomposed into longitudinal, lateral, heave, and yaw degrees of freedom. Additionally, the orientation representation of the vehicle is changed to a Z-Y-X Euler angles set. The resulting linearized dynamics model is

$$\begin{aligned}\dot{v}_x &= -\frac{\overline{C}_{D1}}{m}v_x + g\theta & \dot{\phi} &= \omega_x & \dot{\omega}_x &= -\frac{C_{M_1}}{J_{xx}}v_y + \frac{1}{J_{xx}}\tau_x\\ \dot{v}_y &= -\frac{\overline{C}_{D2}}{m}v_y - g\phi\ , & \dot{\theta} &= \omega_y\ , & \dot{\omega}_y &= -\frac{C_{M_2}}{J_{yy}}v_x + \frac{1}{J_{yy}}\tau_y\\ \dot{v}_z &= -\frac{\overline{C}_{D3}}{m}v_z + \frac{1}{m}T & \dot{\psi} &= \omega_z & \dot{\omega}_z &= \frac{1}{J_{zz}}\tau_z\end{aligned} \tag{11}$$

Converting (11) to state-space representation and adding time-delay parameters to the control inputs, the resulting state-space gray-box models are

$$\begin{bmatrix}\dot{v}_x\\ \dot{\omega}_y\\ \dot{\theta}\end{bmatrix}=\begin{bmatrix}X_{v_x} & 0 & g\\ M_{v_x} & 0 & 0\\ 0 & 1 & 0\end{bmatrix}\begin{bmatrix}v_x\\ \omega_y\\ \theta\end{bmatrix}+\begin{bmatrix}0\\ M_{\tau_y}\\ 0\end{bmatrix}\tau_y(t-\Delta t_1) \tag{12}$$

$$\begin{bmatrix}\dot{v}_y\\ \dot{\omega}_x\\ \dot{\phi}\end{bmatrix}=\begin{bmatrix}X_{v_y} & 0 & -g\\ M_{v_y} & 0 & 0\\ 0 & 1 & 0\end{bmatrix}\begin{bmatrix}v_y\\ \omega_x\\ \phi\end{bmatrix}+\begin{bmatrix}0\\ M_{\tau_x}\\ 0\end{bmatrix}\tau_x(t-\Delta t_2) \tag{13}$$

$$\begin{bmatrix}\dot{\omega}_z\\ \dot{\psi}\end{bmatrix}=\begin{bmatrix}0 & 0\\ 1 & 0\end{bmatrix}\begin{bmatrix}\omega_z\\ \psi\end{bmatrix}+\begin{bmatrix}M_{\tau_z}\\ 0\end{bmatrix}\tau_z(t-\Delta t_3) \tag{14}$$

$$\dot{v}_z=X_{v_z}v_z+X_T T(t-\Delta t_4), \tag{15}$$

where $X_{v_x}=-m^{-1}\overline{C}_{D1}, X_{v_y}=-m^{-1}\overline{C}_{D2}, X_{v_z}=-m^{-1}\overline{C}_{D3}, M_{v_x}=-J_{yy}^{-1}C_{M_2}, M_{v_y}=-J_{xx}^{-1}C_{M_1}, M_{\tau_x}=J_{xx}^{-1}, M_{\tau_y}=J_{yy}^{-1}, M_{\tau_z}=J_{zz}^{-1}$, $X_T=m^{-1}$, and $\Delta t_i \forall\ i\in\{1,2,3,4\}$ are the unknown stability derivatives and model parameters.

## 4.2 | Model parameter identification

The Comprehensive Identification from FrEquency Response (CIFER®)[12] program was utilized to identify the parameters of the state-space model representing the quadrotor dynamics linearized about hover. The frequency-domain system identification procedure is more suitable than time-domain techniques for an unstable vehicle like the quadrotor. It avoids divergence due to time-domain integration of the equations of motion while simultaneously minimizing errors associated with bias effects and processing noise.

To perform frequency-domain system identification, individual degrees of freedom of the quadrotor are excited using an automated frequency sweep maneuver. The frequency sweep involves a sinusoidal trajectory with frequency

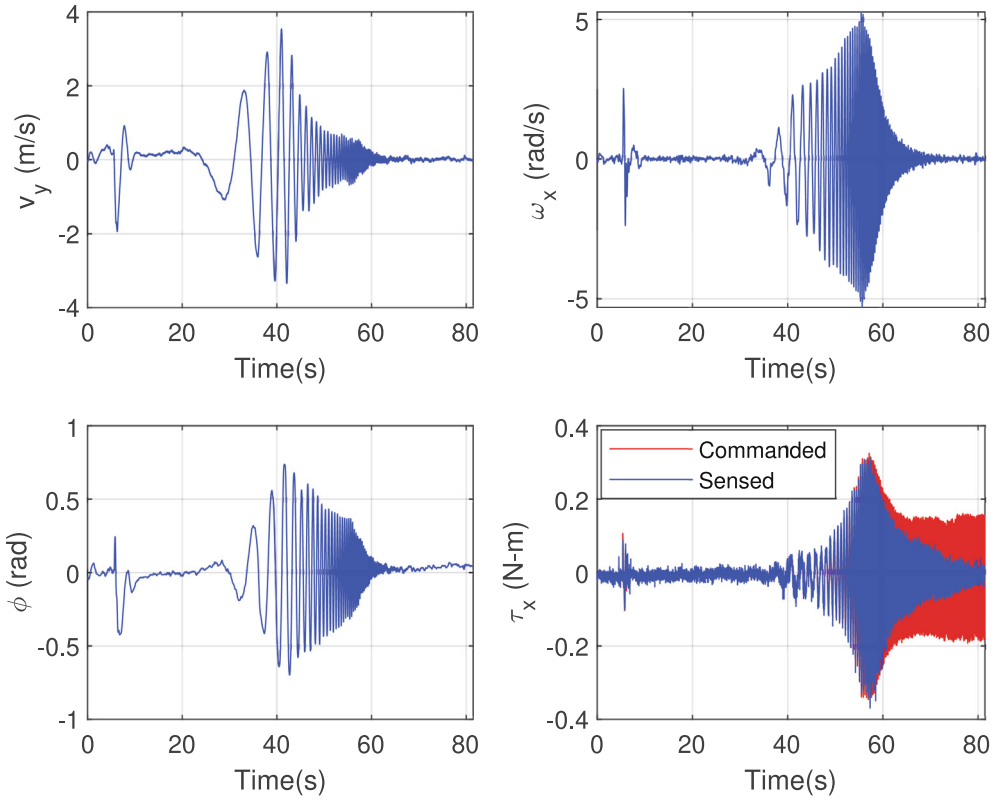


**FIGURE 6** Automated frequency sweep data for lateral degree of freedom used as input to CIFER®.

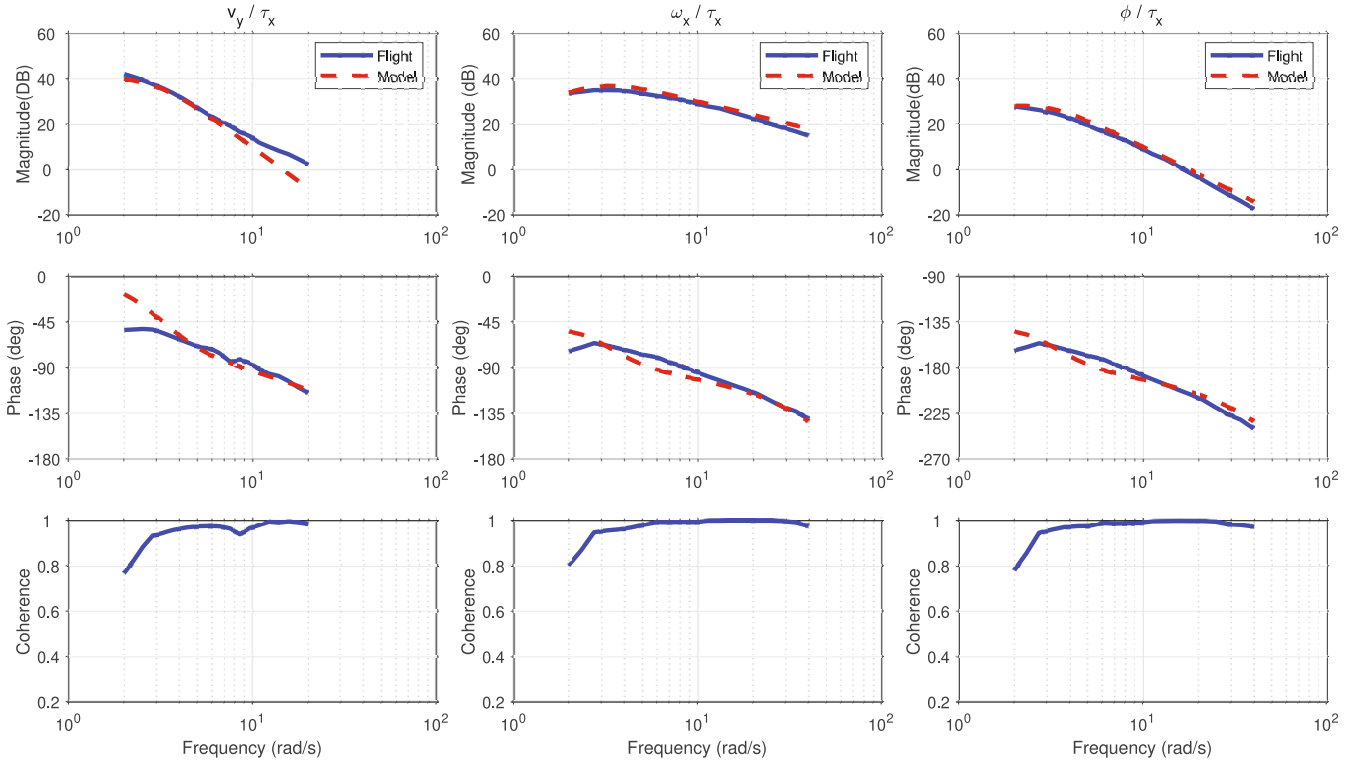


**FIGURE 7** Model-data agreement using stability derivatives identified in CIFER®. Coherence represents the fractional part of the output signal power that is produced by the input at that frequency.

increasing exponentially in time from 0.1 to 50 rad/s. The amplitude is automatically adjusted within manually specified bounds to ensure the control commands do not saturate and the current consumption is below a set threshold. In addition to the controller-generated commands, individual motor RPMs were also estimated using each motor's measured back Electromotive force (EMF) and supplied to the control allocation model to obtain more accurate albeit noisy control input data. Traditionally, system identification is performed with controller-generated commands, and this procedure is labeled as Method 1. An alternate procedure involving control input measurements obtained from motor RPM data is also investigated and labeled Method 2.

Figure 6 shows the state estimates and the lateral control inputs of the automated frequency sweep maneuver of the lateral degree of freedom. The difference between the commanded torque and the torque obtained from sensing the motor RPM increases with frequency due to the influence of the ESCs' response and structural characteristics of the rotor. Figure 7 shows the resulting model fit to the flight data for the lateral degree of freedom. Similar results were also obtained for the other degrees of freedom. Table 3 presents the identified stability derivatives, as well as their Cramer-Rao (C-R) bounds and insensitivities showing the level of confidence of the identification. Some parameters have low confidence scores as it is difficult to improve coherence at lower frequencies. Table 4 shows a comparison of the model parameters extracted from the identified stability derivatives with true value estimates obtained from sources mentioned in Table 1. Method 2 provides more accurate estimates of the inertial parameters. As for the drag parameters, since their confidence scores are low, their estimates have large errors.

## 5 | NONLINEAR PARAMETER ESTIMATION AND CONTROL

This section presents a methodology for recursive, in-flight estimation of drag force coefficients and wind gusts. Although the nonlinear model described in Section 3 is used for estimation, other models can also replace it without changing the estimation methodology. Experimental results are summarized with relevant observations and discussions detailing the framework's benefits and limitations.

**TABLE 3** Identified stability derivatives.

| | Method 1 | | | Method 2 | | | |
|---|---|---|---|---|---|---|---|
| **Parameter** | **Value** | **C-R (%)** | **Insensitivity** | **Value** | **C-R (%)** | **Insensitivity** | **Confidence rating** |
| $X_{v_x}$ | −0.11 | 158.2 | 72.59 | −0.055 | 245.7 | 119.3 | LOW |
| $M_{v_x}$ | −4.27 | 5.91 | 2.55 | −5.235 | 5.34 | 2.28 | MED |
| $M_{\tau_y}$ | 234.2 | 2.48 | 1.08 | 273.7 | 2.6 | 1.1 | HIGH |
| $\Delta_{t_1}$ | 0.051 | 3.2 | 1.59 | 0.022 | 7.11 | 3.54 | MED |
| $X_{v_y}$ | −1.32 | 16.09 | 7.74 | −1.513 | 14.24 | 6.92 | LOW |
| $M_{v_y}$ | −1.89 | 9.12 | 4.08 | −2.5 | 8.37 | 3.68 | MED |
| $M_{\tau_x}$ | 298.6 | 2.43 | 1.07 | 319.6 | 2.49 | 1.07 | HIGH |
| $\Delta_{t_2}$ | 0.048 | 3.38 | 1.67 | 0.023 | 7.08 | 3.51 | MED |
| $M_{\tau_z}$ | 241.8 | 2.63 | 1.32 | 241.6 | 2.64 | 1.32 | HIGH |
| $\Delta_{t_3}$ | 0.064 | 11.84 | 5.92 | 0.096 | 5.58 | 2.79 | MED |
| $X_{v_z}$ | −0.16 | 36.03 | 17.6 | −0.15 | 39.98 | 19.54 | LOW |
| $X_T$ | 0.99 | 3.81 | 1.88 | 1.045 | 3.81 | 1.89 | HIGH |
| $\Delta_{t_4}$ | 0.016 | 34.18 | 16.84 | 0.013 | 41.59 | 20.49 | LOW |

*Note*: Method 1 uses controller generated command data and Method 2 uses control data obtained from RPM sensing.

**TABLE 4** Estimates of model parameters extracted from the identified stability derivatives.

| **Parameter name** | **Symbol** | **Method 1** | **Method 2** | **True value belief** | **Units** |
|---|---|---|---|---|---|
| Mass | $m$ | 1.004 | 0.957 | 0.966 | kg |
| $x$ moment of inertia | $J_{xx}$ | 0.00335 | 0.00312 | 0.00309 | kg·m$^2$ |
| $y$ moment of inertia | $J_{yy}$ | 0.00427 | 0.00365 | 0.00345 | kg·m$^2$ |
| $z$ moment of inertia | $J_{zz}$ | 0.00413 | 0.00414 | 0.00423 | kg·m$^2$ |
| $x$ body drag moment coefficient | $C_{M_1}$ | 0.0063 | 0.0078 | — | N·s |
| $y$ body drag moment coefficient | $C_{M_2}$ | 0.018 | 0.019 | — | N·s |
| $x$ body linear drag force coefficient | $\overline{C}_{D1}$ | 0.11 | 0.0526 | 0.21 | N·s/m |
| $y$ body linear drag force coefficient | $\overline{C}_{D2}$ | 1.325 | 1.45 | 0.42 | N·s/m |
| $z$ body linear drag force coefficient | $\overline{C}_{D3}$ | 0.16 | 0.143 | — | N·s/m |

*Note*: Method 1 uses controller generated command data and Method 2 uses control data obtained from RPM sensing.

## 5.1 | Methodology

A coupled estimation and UAV control framework based on the author's prior work[23] is used to recursively estimate drag force coefficients and wind in separate experiments. The online parameter and state estimation is carried out by a sq-UKF framework, which determines an 11-dimensional state vector from 9-dimensional sensor data and 4-dimensional control inputs. In the wind estimation experiment, the observability is sufficient in the hover state. However, observability is lost in the drag force coefficient estimation experiment when speed is zero. Hence, to maintain a non-zero speed, the quadrotor is commanded to track an ellipsoid trajectory, providing sufficient activation for drag force coefficient estimation.

The state vector of sq-UKF framework includes the states and either the drag force coefficients or the wind vector, depending on the experiment. For the drag force coefficient estimation experiment, the estimator's state vector is

$$\mathbf{x} = \left[\mathbf{s}^{\mathsf{T}}\ \mathbf{v}^{\mathcal{B}\mathsf{T}}\ \boldsymbol{\omega}^{\mathcal{B}\mathsf{T}}\ \mathbf{C}_D^{\mathcal{B}\mathsf{T}}\right]^{\mathsf{T}} \tag{16}$$

and for the wind estimation experiment the state vector is

$$\mathbf{x} = \left[\mathbf{s}^\mathsf{T}\ \mathbf{v}^{\mathcal{B}\mathsf{T}}\ \boldsymbol{\omega}^{\mathcal{B}\mathsf{T}}\ \mathbf{v}_a^{\mathcal{B}\mathsf{T}}\right]^\mathsf{T}, \tag{17}$$

where $\mathbf{s} := [s_1\ s_2]^\mathsf{T} \in \mathbf{S}^2$ is the tilt of the quadrotor expressed in stereographic coordinates. The process model of each component of the state vector and the measurement model is the same as in the original formulation.[23] The orientation estimate mean values in the stereographic coordinates are combined with the yaw value from the VIO camera and transformed to SO(3) orientation before supplying it to the SE(3) controller.

## 5.2 | Drag force coefficient estimation

The drag force coefficients estimation experiments are performed in an indoor flying area of approximately $4 \times 3\ \mathrm{m}^2$ and a height of 4.5 m. For safety, the flight area was fenced in the flight controller's software to $2 \times 2\ \mathrm{m}^2$ and height to 2.5 m. The quadrotor is commanded to track an ellipse-shaped trajectory as shown in Figure 8 whose time-parameterized expression is

$$\boldsymbol{r}_d^{\mathcal{W}}(t) = \begin{bmatrix} A_x \sin(2\pi n_x t) \\ -A_y \cos(2\pi n_y t) \\ A_z \sin(2\pi n_z t) + z_0 \end{bmatrix}, \tag{18}$$

where $z_0 = 1.75$ m, $n_x = n_y = n_z = \frac{1}{\pi} s^{-1}$, and the amplitudes $A_x, A_y, A_z$ are linearly ramped up in 7 s from 0.25 to 0.5 m for $A_z$ and 0.5 to 1 m for both $A_x$ and $A_y$. The ramping up of the trajectory amplitudes is implemented to reduce the large oscillations caused by low control performance due to the inaccurate nominal values of the drag coefficients. The nominal values for the drag coefficients were selected as $\mathbf{C}_D^{\mathcal{B}} = [0.5\ 1\ 0]^\mathsf{T}$ to simulate a worst-case scenario where the controller's drag compensation almost eliminates the damping effects of velocity feedback, essentially starting the vehicle with an almost unstable control behavior. The $z$-drag coefficient $C_{D_3}$ is set to zero; we observed that the estimator is incapable of estimating it and, when set to a non-zero value, it can lead to divergence. Figure 9 shows the commanded position and the quadrotor's position-tracking performance improving with time. Figure 10 shows that the position and velocity tracking errors decrease rapidly in 5 s and oscillate around 0.2 m and 0.5 m/s, respectively. Figure 11 shows the drag force coefficients' estimation performance and Table 5 compares the final drag coefficient estimates with the true value belief. The estimates of the coefficients converge quickly to a value that is slightly higher than the value of the true coefficient belief. A possible cause is that there are other uncertainties present in the system dynamics. Also, the aerodynamic force model of (2) is a simplified model and does not fully capture the aerodynamic effects, thus creating residual dynamics affecting the estimation and tracking performance.

**FIGURE 8** Experimental setup showing the quadrotor tracking an ellipse trajectory.

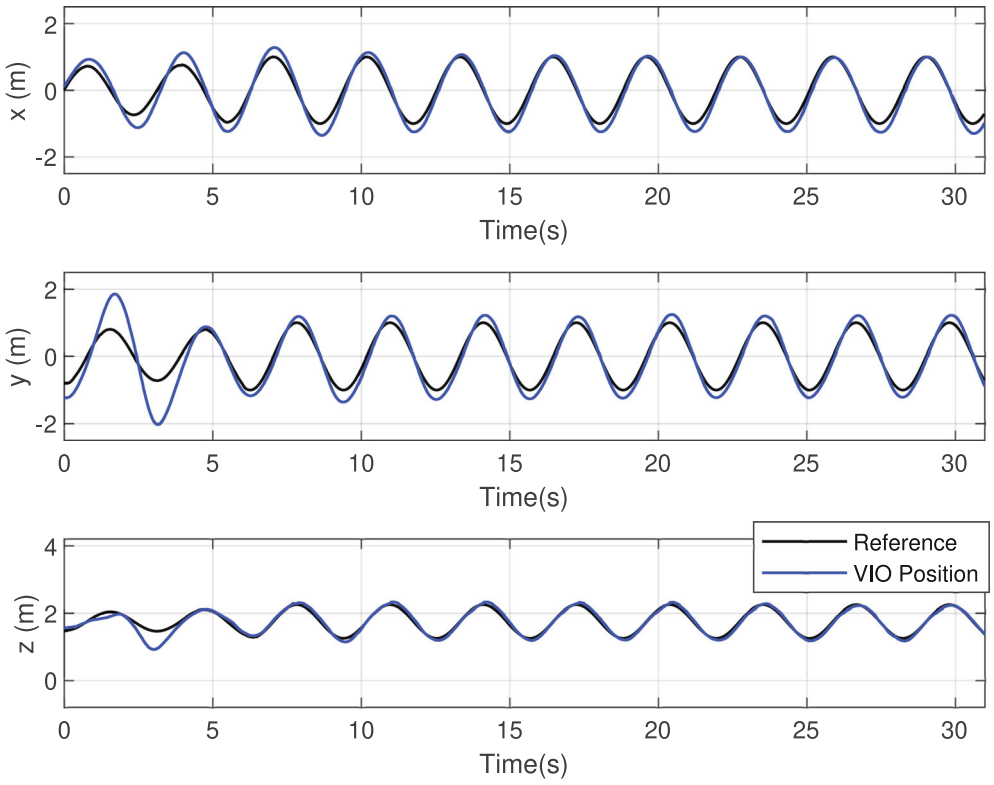


**FIGURE 9** Position tracking of the ellipse trajectory.

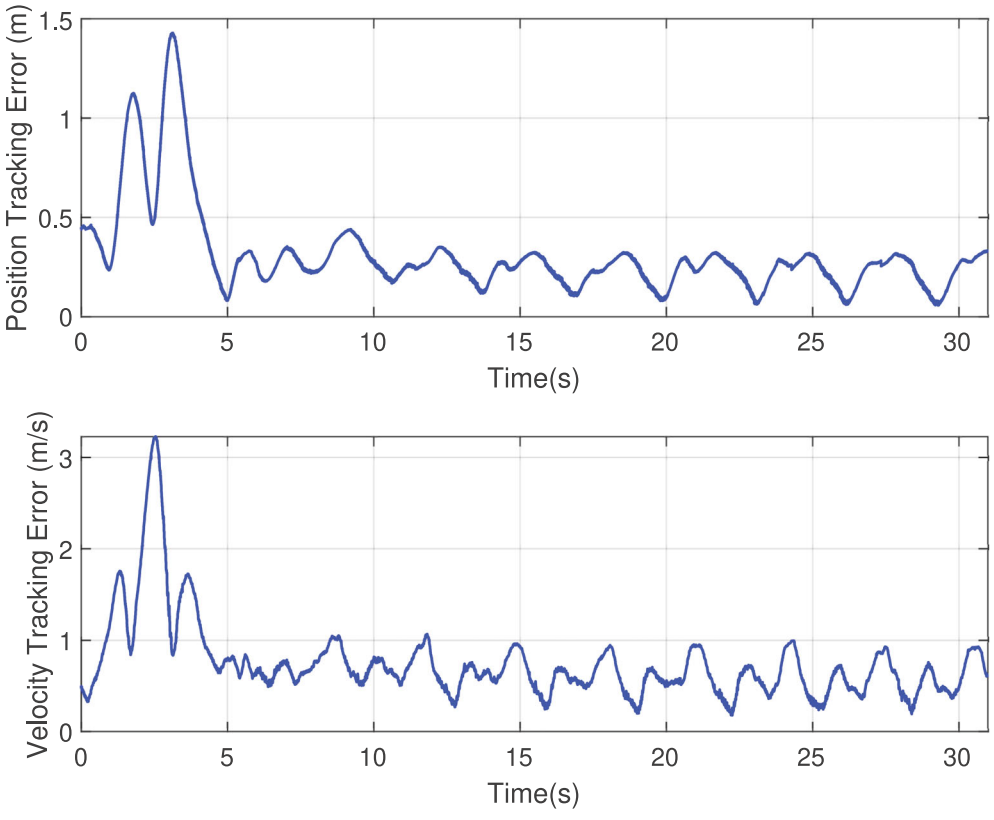


**FIGURE 10** Position and velocity tracking error magnitudes.

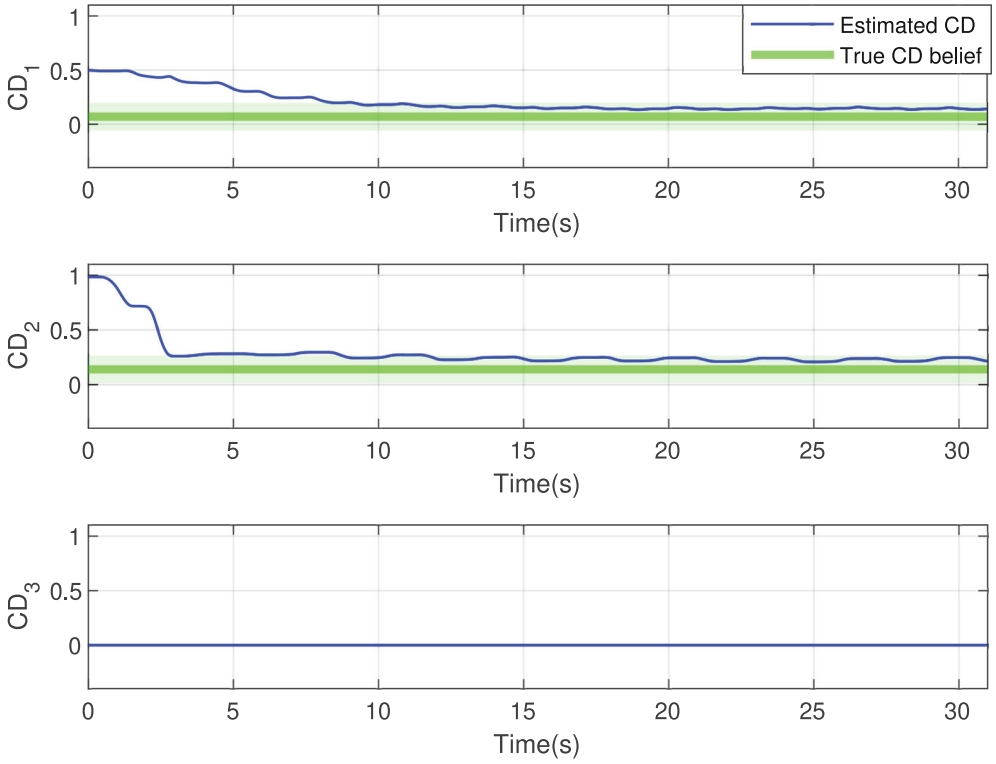


**FIGURE 11** Drag force coefficient estimation performance.

**TABLE 5** sq-UKF's final drag force coefficient estimates.

| Parameter name | Symbol | sq-UKF's estimate | True value belief | Units |
|---|---|---|---|---|
| $x$ body drag force coefficient | $C_{D_1}$ | 0.13 | 0.07 | $N{\cdot}s^2/m^2$ |
| $y$ body drag force coefficient | $C_{D_2}$ | 0.21 | 0.14 | $N{\cdot}s^2/m^2$ |
| $z$ body drag force coefficient | $C_{D_3}$ | 0 | — | $N{\cdot}s^2/m^2$ |

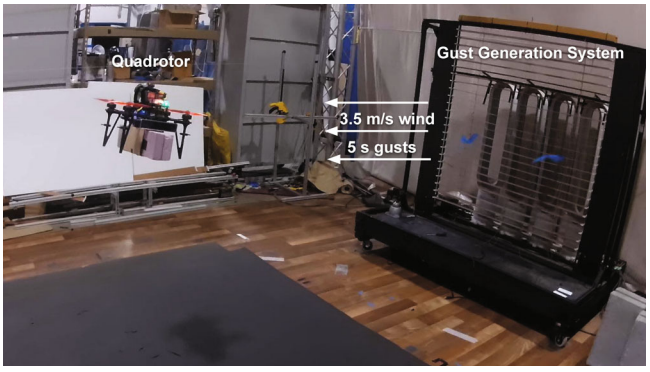


**FIGURE 12** Experimental setup showing the gust generation system on the right and the quadrotor opposing the wind gust forces on the left.

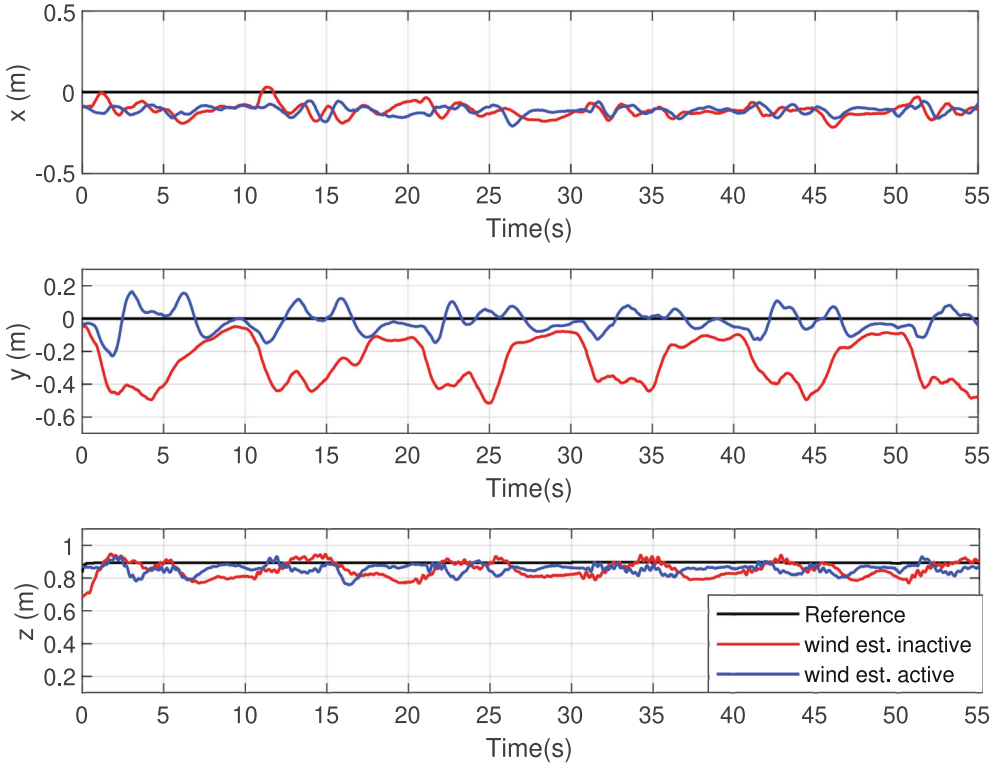


**FIGURE 13** Station holding performance comparison in 5 s duration 3.5 m/s wind gusts in $-w_2$ direction.

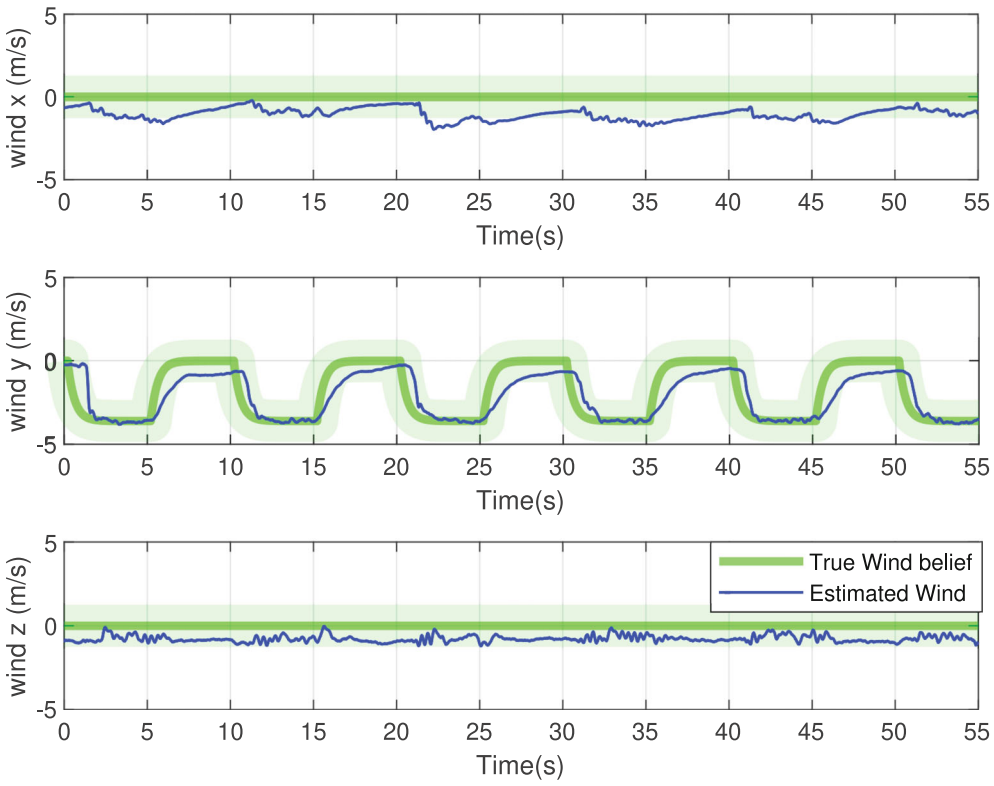


**FIGURE 14** Wind estimation performance in inertial frame. Body frame wind estimates are transformed to inertial frame $\mathcal{W}$.

### 5.3 | Wind estimation

The wind estimation experiments were performed in an indoor gust generation facility shown in Figure 12, which consists of a set of eight Dyson fans behind servo-controlled blinds operated by an Arduino. The blinds open and close automatically to produce 3.5 m/s wind gusts of 5 s duration. The quadrotor heading is set perpendicular to the wind direction to maximize the area on which the wind exerts aerodynamic forces, thereby simulating a worst-case scenario. To evaluate the wind estimation benefits, the control performance was compared against the scenario in which wind estimation is deactivated by removing it from the estimator's state vector. Figure 13 shows the displacements in the quadrotor's position due to the wind. With wind estimation, the quadrotor is able to reject the wind forces and thus has improved station holding performance. Figure 14 shows the sq-UKF's wind estimates compared with the true wind belief obtained from anemometer measurements. The estimates show strong agreement with ground truth belief. Since the wind is estimated from the quadrotor's motion, a delay is present in the estimation. Additionally, the estimator's performance decreases with the wind as the observability is less in hover and no-wind situations.

## 6 | CONCLUSION

This work presents the development and experimental evaluation of a combined approach toward system identification of a UAV by synthesizing tools from rotorcraft, robotics, and nonlinear control and estimation. The frequency-domain linear system identification tool CIFER® was used to accurately estimate the inertial parameters from an automated frequency sweep data. By sensing the motor RPM, accurate control values were obtained. Comparison with the traditional approach of using controller-generated commands revealed improvements in identification accuracy. A reduced version of the square-root unscented Kalman filter and a nonlinear model-based controller were utilized to recursively estimate the drag coefficient and external wind in separate experiments. The computational requirements of the nonlinear estimation and control frameworks were satisfied by designing a custom, low-cost flight controller. System identification accuracy of both linear and nonlinear techniques is evaluated by comparing parameter estimates with true value beliefs obtained from various sources. Experimental evaluation of the drag coefficient estimation showed rapid improvement in the controller's tracking performance. Additionally, wind estimation experiments demonstrated the framework's ability to reject wind gusts without directly sensing the wind.

Ongoing and future works seek to experimentally demonstrate the recursive, in-flight estimation of additional model parameters. Additionally, implementation of the frequency-domain system identification procedure on the onboard computer will enable robust, accurate, and periodic updates of inertial parameters. Furthermore, wind estimation and gust rejection performance can be improved by upgrading the IMU, reducing sensor delay, and incorporating control delay in the estimation framework.


### ACKNOWLEDGMENTS

This research was supported by the US Army under grant W911W61720004.


### CONFLICT OF INTEREST STATEMENT

The authors declare that they have no conflicts of interest.

### DATA AVAILABILITY STATEMENT

The data that support the findings of this study are openly available in quadx_software and Teensy4.0-Flight-Controller repositories at https://github.com/animeshshastry/quadx_software and https://github.com/animeshshastry/Teensy4.0-Flight-Controller respectively.


### ORCID

*Animesh K. Shastry* https://orcid.org/0000-0003-1285-7004


### REFERENCES


1. Hua MD, Hamel T, Morin P, Samson C. Introduction to feedback control of underactuated VTOLvehicles: a review of basic control design ideas and principles. *IEEE Control Syst Mag*. 2013;33(1):61-75. doi:10.1109/MCS.2012.2225931

2. Koo T, Sastry S. Output tracking control design of a helicopter model based on approximate linearization. Proceedings of the 37th IEEE Conference on Decision and Control (Cat No98CH36171). 1998:4, 4:3635-3640. doi:10.1109/CDC.1998.761745
3. Ebeid E, Skriver M, Terkildsen KH, Jensen K, Schultz UP. A survey of open-source UAV flight controllers and flight simulators. *Microprocess Microsyst*. 2018;61:11-20. doi:10.1016/j.micpro.2018.05.002
4. Enns D, Bugajski D, Hendrick R, Stein G. Dynamic inversion: an evolving methodology for flight control design. *Int J Control*. 1994;59(1):71-91. doi:10.1080/00207179408923070
5. Lee D, Kim HJ, Sastry S. Feedback linearization vs. adaptive sliding mode control for a quadrotor helicopter. *Int J Control Autom Syst*. 2009;7(3):419-428. doi:10.1007/s12555-009-0311-8
6. Hovakimyan N, Cao C. L1 adaptive control theory: guaranteed robustness with fast adaptation. *Society for Industrial and Applied Mathematics*; 2010.
7. Lee T, Leok M, McClamroch NH. Geometric tracking control of a quadrotor UAV on SE(3). Paper presented at: 49th IEEE conference on decision and Control. 2010:5420-5425. doi:10.1109/CDC.2010.5717652
8. Meda-Campaña JA. On the estimation and control of nonlinear systems with parametric uncertainties and noisy outputs. *IEEE Access*. 2018;6:31968-31973. doi:10.1109/ACCESS.2018.2846483
9. Jardin RM, Mueller ER. Optimized measurements of unmanned-air-vehicle mass moment of inertia with a bifilar pendulum. *J Aircr*. 2009;46(3):763-775. doi:10.2514/1.34015
10. Anton N, Botez RM, Popescu D. Stability derivatives for a delta-wing X-31 aircraft validated using wind tunnel test data. *Proc Inst Mech Eng Part G: J Aerosp Eng*. 2011;225(4):403-416.
11. Wei W, Cohen K, Tischler MB. System identification and controller optimization of a quadrotor UAV. Proceedings of the AHS International's 71st Annual Forum and Technology Display. 2015. doi:10.4050/JAHS.62.042007
12. Horn JF. Aircraft and rotorcraft system identification: engineering methods with flight test examples. *IEEE Control Syst Mag*. 2016;36(2):117-119. doi:10.1109/MCS.2015.2512078
13. Harding JW, Moody SJ, Jeram GJ, Mansur H, Tischler MB. Development of modern control laws for the AH-64D in hover/low speed flight. Annual Forum Proceedings-American Helicopter Society. 2006 62(3).
14. Tischler M. System identification methods for aircraft flight control development and validation. *Advances in Aircraft Flight Control*. Taylor & Francis; 1995:35-69.
15. Craig W, Lewis J, Paley DA, Young WH Jr. Stabilization of a quadrotor in wind with flow sensing: linear modeling and control for attitude and position hold. VFS Autonomous VTOL Technical Meeting and eVTOL Symposium. 2019 1-10.
16. Svacha J, Paulos J, Loianno G, Kumar V. IMU-based inertia estimation for a quadrotor using Newton-Euler dynamics. *IEEE Robot Autom Lett*. 2020;5(3):3861-3867. doi:10.1109/LRA.2020.2976308
17. Guo J, Tao G, Liu Y. A multivariable MRAC scheme with application to a nonlinear aircraft model. *Automatica*. 2011;47(4):804-812. doi:10.1016/j.automatica.2011.01.069
18. Yang J, Cai Z, Lin Q, Zhang D, Wang Y. System identification of quadrotor UAV based on genetic algorithm. Paper presented at: IEEE Chinese Guidance, Navigation and Control Conference. 2014 2336-2340.
19. Shastry AK, Bhargavapuri MT, Kothari M, Sahoo SR. Quaternion based adaptive control for package delivery using variable-pitch quadrotors. Paper presented at: 2018 Indian Control Conference (ICC). IEEE. 2018 340–345.
20. Bhargavapuri M, Shastry AK, Sinha H, Sahoo SR, Kothari M. Vision-based autonomous tracking and landing of a fully-actuated rotorcraft. *Control Eng Pract*. 2019;89:113-129.
21. Heim EH, Viken E, Brandon JM, Croom M. NASA's learn-to-fly project overview. Atmospheric Flight Mechanics Conference. 2018.
22. Lombaerts T, Oort ER, Chu Q, Mulder JA, Joosten D. Online aerodynamic model structure selection and parameter estimation for fault tolerant control. *J Guid Control Dyn*. 2010;33:707-723.
23. Shastry A, Paley DA. UAV state and parameter estimation in wind using calibration trajectories optimized for observability. *IEEE Control Syst Lett*. 2021;5(5):1801-1806. doi:10.1109/LCSYS.2020.3044491
24. Shastry A, Paley DA. Teensy4.0-flight-controller. 2022 https://github.com/animeshshastry/Teensy4.0-Flight-Controller
25. Shastry A, Paley DA. quadx_software. 2022 https://github.com/animeshshastry/quadx_software
26. Johnson NA, Singhose W. Dynamics and modeling of a quadrotor with a suspended payload. Applied Aerodynamics Conference. 2018. doi:10.2514/6.2018-4213
27. Bangura M, Mahony RE. Nonlinear dynamic modeling for high performance control of a quadrotor. Australasian Conference on Robotics and Automation. 2012.


**How to cite this article:** Shastry AK, Paley DA. System identification for high-performance UAV control in wind. *Int J Robust Nonlinear Control*. 2023;33(17):10451-10467. doi: 10.1002/rnc.6935