

Model-based invariant filters for quadcopter wind estimation

Hao Chen

Oklahoma State University, OK, USA

He Bai

Oklahoma State University, OK, USA

Abstract— Wind estimation for small unmanned aerial vehicles (sUAVs) can not only improve navigation and flight performance, but also be used in environmental studies, agriculture, and meteorology. We consider a nonlinear estimation problem where a quadcopter hovers in a wind field. Considering a general form of system dynamics including drag, thrust models, and IMU biases, we establish the invariance of system dynamics and equivariance of measurements by designing transformations to states, inputs, and measurements. We then design an invariant extended Kalman filter (IEKF) to estimate a sUAV's states and wind velocities. We conduct five hover tests to validate the effectiveness of the proposed algorithms. The experimental results show that EKF and IEKF perform better compared to benchmark methods, such as static mapping method and translational EKF (TEKF). IEKF produces better transient performance when compared to EKF.

Index Terms— wind estimation, IEKF, invariance, quadcopter

I. Introduction

It has become popular to employ small unmanned aerial vehicles (sUAV) for wind estimation in recent years. Accurate wind velocity information can be used to improve flight performance [11] as well as in atmospheric, meteorology research [12], and environmental studies [12], [13]. Wind estimation is a challenging but important task, particularly in the lower atmosphere where wind is highly variable.

There are a number of traditional ways to measure wind including ground based systems such as SoDAR, LiDAR, and wind towers. However, those methods are expensive, immovable or hard to move, and only provide wind velocity estimates in a fixed point or a small area [10]. Another way is to use tethered balloons, which

is expensive, labor intensive, and hard to operate [14]. Since sUAV is affordable and flexible, it has become a popular platform for wind measurement and estimation.

In the last decade, both fixed-wing UAVs and rotary-wing UAVs have been increasingly used for wind sensing and estimation [15]. These two types of UAV platforms are complementary to each other. A fixed-wing UAV does not hover in space and requires to fly horizontally to measure wind. Therefore, they are more used for long-range missions. In comparison, a rotary-wing UAV has the capability of estimating wind profiles or temporal variability of the wind at one location.

There are four major approaches to rotary UAV wind estimation. (1) Mounting wind sensors on sUAVs [16], [17], [14]: This method requires rotary UAV equipped with different types of anemometers to estimate wind velocities. It is easy to implement but there are also some disadvantages. First, flight endurance will be reduced due to the weight and power requirements of sensors and the platforms. Second, wind sensors may be expensive compared to a sUAV platform. Third, sUAV propellers heavily impact the airflow around the sUAV. (2) Static mapping methods: Those methods explore the relation between wind velocity and states of the sUAV, such as tilt angle, using collected data [18], [19], [17], [14]. These methods are also easy to implement but may not be accurate in strong wind fields. Recently, [3] develops a “wind vane mode” algorithm which commands the self-made CopterSonde to always face the wind, and then the horizontal wind can be estimated based on tilt angle through a linear regression model. (3) Machine Learning (ML) methods [5], [20], [21], [22]: They are data-driven to find the relation between quadcopters states/measurements and wind velocity. Such methods do not need to consider complex dynamic models but require high quality training data. ML results may be specific to the training data of a quadcopter and may not extend to others. In [5], a long short-term memory neural network (LSTM NN) has been verified in simulations. In [20], LSTM NN and K-nearest neighbor (KNN) have been verified experimentally. Reference [21] uses not only LSTM NN and NN, but also Gaussian-process-regression based on collected test flights data. Deep NN (DNN) has been used in [22]. (4) Model-based methods [23], [24], [8], [9], [10], [1], [2]: such methods model the quadcopter system dynamics, available measurements, and wind velocity, so that the wind velocity information is extracted from a quadcopter's motion. The authors in [8] propose a Kalman filter based on standard navigation sensors and equations of motion. A small quadrotor with spherical shape has been developed and indoor and outdoor flight tests have been done to validate the proposed algorithm. Reference [9] proposes a new dynamic model-based approach, which relies on the assumption that the thrust can be measured or estimated while drag can be related to airspeed. The effectiveness of this method is validated by the hover and flight experiments. Reference [10] proposes three models to estimate the wind: the first one is the

Manuscript received XXXXX 00, 0000; revised XXXXX 00, 0000; accepted XXXXX 00, 0000. This work is supported by the National Science Foundation (NSF) under Grant No.1925147. (Corresponding author: He Bai).

Hao Chen and He Bai are with the school of mechanical and aerospace engineering, Oklahoma State University, Stillwater, OK 74078 USA (e-mail: hao.chen@okstate.edu, he.bai@okstate.edu).

TABLE I
Comparison of various wind estimation methods using rotary UAVs since 2019

Year	Author	Method	simulation/experiments	Thrust is from	Drag Model	Vert. Wind
2019	Gonzalez-Rochaz et al. [1]	Model-based	Outdoor hovering	Wind tunnel/flight tests	Quadratic	NO
2020	Gonzalez-Rochaz et al. [2]	Model-Based	Outdoor hovering	NA	NA	NO
2020	Segales et al. [3]	Static Mapping	Outdoor hovering	N/A	Quadratic	NO
2020	Simmma et al. [4]	Model-based	Outdoor hovering	Thrust stand test	Linear	NO
2020	Allision et al. [5]	Machine learning	Simulation	NA	NA	NO
2022	Chen et al. [6]	Model-based	Simulation	Nominal thrust	Quadratic	Yes
2022	Chen, Bai [7]	Model-based	Simulation	Various thrust models	Quadratic	Yes
2022	Gautier et al. [8]	Model-based	Indoor/outdoor flying	Gravity projection	Linear	NO
2022	Meier et al. [9]	Model-based	Outdoor hovering	Nominal thrust	Linear/quadratic	Yes
2022	McConville et al. [10]	Model-based	Outdoor flying	Wind tunnel test	Quadratic	NO

mapping method between wind speed and body angle, the second one is a novel approach to modeling through maneuvers, and the last method relies on the models which are obtained via wind tunnel tests. Gonzalez-Rochaz et al. [1] compare various model based algorithms which include kinematic particle model, dynamic particle model and rigid-body model. In [2], Gonzalez-Rochaz et.al also propose a model based approach to estimate vertical profile of horizontal wind velocity components. In Table I, we summarize and compare the publications about the wind estimation using rotary UAVs since 2019.

Given the sUAV's models and measurements, a common way to fuse their information and estimate states and wind velocity is to use Kalman filter or extended Kalman filter (EKF). EKF is designed for nonlinear systems but the stability of EKF is not guaranteed. Fortunately, symmetry properties of a dynamical system can be leveraged to improve EKF performance and stability. Reference [25] proposes an invariant observer design procedure to take advantage of symmetry properties of a system. Compared to observers, EKF is more popular due to its adaptive gain tuning through a Riccati equation. Therefore invariant EKFs (IEKFs) are designed and they are more stable in transient stage than EKFs by taking advantage of symmetry. Reference [26] extends IEKFs for state and additive disturbance estimation. Reference [27] derives an IEKF on matrix Lie groups and shows that IEKF is computationally less intensive and produces better performance than the EKF. IEKF has found various applications, such as legged robot state estimation [28] and SLAM [29]. Reference [30] proposes a novel equivariant filter which is a general filter for system on homogeneous spaces. In [31], equivariant filter is used for aided inertial navigation with input measurement biases.

In this paper, we consider the model-based approach to wind estimation using a quadcopter. We take advantage of the constructed symmetry property of system dynamics to design an IEKF. The IEKF shows less transient error and fast convergence speed for wind and states estimation in simulations and experiments. The contributions of this paper are mainly on three aspects: (1) We design general transformations for states, inputs and measurements to identify the symmetry in system dynamics and measurement equations; (2) We derive the wind estimation IEKF

with IMU biases; (3) We conduct outdoor experiments to validate the effectiveness of the IEKF for estimation of horizontal wind velocity. Compared to our previous work [6], [7], this paper establishes invariance and IEKF designs for biased IMUs and presents outdoor experimental results to verify the effectiveness of the proposed methods.

The rest of the paper is organized as follows. In Section II, we introduce the modeling of wind estimation which includes quadcopter system dynamics, measurements, thrust and drag models. In Section III, we construct the invariance of the system dynamics and the equivariance of the measurement equations, and derive our IEKF algorithm. Section IV provides outdoor experimental results for hover tests and compare wind estimation of various estimators. Conclusions and future work are presented in Section V.

II. Problem Formulation

A. system dynamics and measurement equations

The translational dynamics and attitude kinematics of a quadcopter subject to a wind disturbance in the north-east-down (NED) frame are given by [1]:

$$\begin{aligned}\dot{x} &= R_b v \\ m\dot{v} &= mv \times \omega + mR_b^T g - f_c b_3 + f_d + mR_b^T \dot{v}_w \\ \dot{R}_b &= R_b S(\omega)\end{aligned}\quad (1)$$

where $x \in \mathbb{R}^3$ is the inertial position, $v \in \mathbb{R}^3$ is the ground velocity in the body frame, $v_w \in \mathbb{R}^3$ is the wind vector in the inertial frame, $R_b \in SO(3)$ is the orientation of the quadcopter with respect to the inertial frame and $SO(3)$ is the 3D special orthogonal group, $m \in \mathbb{R}^+$ is mass of the quadcopter, $g = [0, 0, g]^T \in \mathbb{R}^3$ denotes the gravity acceleration vector in the body frame, $f_c \in \mathbb{R}^+$ is the amplitude of the thrust control input and $f_c b_3 \in \mathbb{R}^3$ in which $b_3 = [0, 0, 1]^T$ denotes the thrust vector in body frame, $\omega \in \mathbb{R}^3$ is the angular velocity of the quadcopter in the body frame, the function $S(\cdot) : \mathbb{R}^3 \rightarrow so(3)$ satisfies $S(a)b = a \times b$ for $a, b \in \mathbb{R}^3$ where $so(3)$ denotes lie algebra of the $SO(3)$, and $f_d \in \mathbb{R}^3$ denotes the drag force.

We model the wind vector v_w as the output of a linear dynamical system

$$\dot{d} = Ad, \quad v_w = Cd, \quad (2)$$

where $d \in \mathbb{R}^{m \times 1}$, $A \in \mathbb{R}^{m \times m}$, $C \in \mathbb{R}^{3 \times m}$ and the matrices A and C are assumed known. Such wind dynamics can be used to model various types of mean wind profiles, such as sinusoidal and constant wind. In the filter implementation, we add process noise to the d dynamics to mitigate the effect of the turbulent component in the wind on the estimation.

Denote the biases in the gyroscope and accelerometer measurements by $b_\omega \in \mathbb{R}^3$ and $b_a \in \mathbb{R}^3$, respectively. We rewrite the system dynamics in (1) in terms of x , v_r , R_b , d , b_ω and b_a as

$$\begin{aligned} \dot{x} &= R_b v_r + Cd \\ \dot{v}_r &= v_r \times (\bar{\omega} - b_\omega) + R_b^T g - \frac{1}{m} f_c b_3 + \frac{1}{m} f_d \\ \dot{R}_b &= R_b S(\bar{\omega} - b_\omega) \\ \dot{d} &= Ad \\ \dot{b}_\omega &= 0 \\ \dot{b}_a &= 0 \end{aligned} \quad (3)$$

where $\bar{\omega} = \omega + b_\omega$ and $\bar{a} = a + b_a$. Note that $\bar{\omega} \in \mathbb{R}^3$ is the measured angular velocity which contains the bias b_ω . Similarly let $\bar{a} \in \mathbb{R}^3$ be the measured acceleration containing the bias b_a .

We assume that the quadcopter is equipped with a GPS, and a 3-axis accelerometer and gyroscope, and a magnetometer. The measured angular velocity from the gyroscope is used in the system dynamics as one of the inputs. The measured outputs $y = (y_x^T, y_a^T, y_b^T)^T$ are

$$y = \begin{pmatrix} y_x \\ y_a \\ y_b \end{pmatrix} = \begin{pmatrix} x \\ a + b_a \\ R_b^T B \end{pmatrix} \quad (4)$$

where $B \in \mathbb{R}^3$ is the earth's magnetic field expressed in the inertial frame and $a \in \mathbb{R}^3$ is the specific acceleration vector in the body frame given by

$$a = \frac{1}{m} (-f_c b_3 + f_d). \quad (5)$$

B. thrust model

The thrust model is important for the accuracy of the wind estimation, especially for the vertical wind estimation. Obtaining an accurate thrust model in the presence of wind is a challenging task. In our previous work [7], we survey existing thrust models that consider aerodynamic effects. Such models require either wind tunnel tests or thrust stand tests. However, our simulation results in [7] indicate that such complex thrust models mainly improve vertical wind estimation. In our experiments, we find that a simple thrust model is sufficient for horizontal wind estimation.

C. drag model

Denote by $v_r \in \mathbb{R}^3$ the relative air velocity in the body frame. That is, $v_r = v - R_b^T v_w$. The drag model describes a relation between the drag force and relative air velocity. In [18], a quadratic drag model is used

$$f_d = \frac{1}{2} \rho A_{\text{proj}} D v_r^2 \quad (6)$$

where ρ as the air density, A_{proj} is the aircraft surface area normal to the wind and D is the drag coefficient matrix. Such a quadratic drag model is also used in [1], [10], [9]. In [4], [8], [9], linear drag models are used.

Our invariance properties apply to both linear and quadratic drag models. Thus, we consider a general drag model including a constant offset and linear drag and quadratic drag terms

$$f_d = d_0 + D_1 v_r + D_2 |v_r| v_r, \quad (7)$$

where $d_0 \in \mathbb{R}^3$ is a constant offset term, and $D_1 \in \mathbb{R}^{3 \times 3}$ and $D_2 \in \mathbb{R}^{3 \times 3}$ are coefficient matrices of linear and quadratic drag terms, respectively.

III. IEKF Design

A. Invariance of the dynamics and equivariance of the measurements

The symmetries of the system are associated with the special Euclidean group $G = SE(3)$ which consists of rotations and translations in dimension 3. Let $(x_g, R_g) \in G$, where $x_g \in \mathbb{R}^3$ and $R_g \in SO(3)$.

We define the state as $X = (x, v_r, R_b, d, b_\omega, b_a)$ and the input as $U = (\bar{\omega}, f_c b_3)$. We construct two transformations on the state X and the input U such that the system dynamics is invariant and the measurement equation is equivariant.

We consider simple thrust models which are not a function of states, e.g., the polynomial thrust model (34) as shown in Section IV or $f_c = k_\Omega \sum_{i=1}^4 \Omega_i^2$ where k_Ω is the thrust model coefficient and Ω_i is the angular velocity of the rotor. Then the thrust is treated as one of the inputs to the system. For some complicated thrust models which are functions of the states, the transformations may need to be specially designed and we refer to [7] for such transformations.

Define three transformations $\varphi_g(X)$, $\psi_g(U)$, and $\varrho_g(h(X, U))$ on the state X , the input U , and the measurement equation h as

$$\varphi_g(X) = \begin{pmatrix} x + x_g \\ R_g v_r \\ R_b R_g^T \\ d \\ R_g b_\omega \\ R_g b_a \end{pmatrix}, \quad \psi_g(U) = \begin{pmatrix} R_g \bar{\omega} \\ R_g b_3 \\ \Omega \\ \Theta_{g1}(\theta_c) \\ \Theta_{g2}(\theta_d) \end{pmatrix}, \quad (8)$$

and

$$\varrho_g(h(X, U)) = \begin{pmatrix} x + x_g \\ R_g(a + b_a) \\ R_g R_b^T B \end{pmatrix}, \quad (9)$$

respectively. In (8), θ_c includes thrust model coefficients and $\Theta_{g1}(\theta_c)$ represents a group operation on θ_c which will be identified in each of the thrust models. θ_d includes drag model coefficients and the $\Theta_{g2}(\theta_d)$ represents a group operation on θ_d which will be identified in each of the drag models. For the theoretical framework of IEKF, we consider the general system dynamics with the general thrust and drag models, and therefore we expand the input from $U = (\bar{\omega}, f_c b_3)$ to $U = (\bar{\omega}, b_3, \Omega, \theta_c, \theta_d)$ to construct symmetry.

PROPOSITION 1. *Suppose that there exist two transformations $\Theta_{g1}(\cdot)$ and $\Theta_{g2}(\cdot)$ such that for any R_g ,*

$$f_c(\Omega, v_r, \theta_c) = f_c(\Omega, R_g v_r, \Theta_{g1}(\theta_c)) \quad (10)$$

$$R_g f_d(v_r, \theta_d) = f_d(R_g v_r, \Theta_{g2}(\theta_d)). \quad (11)$$

Then the system (3) is invariant with respect to the transformations $\varphi_g(X)$ and $\psi_g(U)$, and the measurement equation (4) is G -equivariant with respect to transformations $\varrho_g(h)$. \square

Proof:

According to definition 2 in [25], the system is said to be invariant for all g, X, U if

$$\frac{d}{dt}(\varphi_g(X)) = f(\varphi_g(X), \psi_g(U)). \quad (12)$$

By setting $\varphi_g(X) = [\tilde{x}, \tilde{v}_r, \tilde{R}_b, \tilde{d}, \tilde{b}_\omega, \tilde{b}_a]$ and $\psi_g(U) = [\tilde{\omega}, \tilde{b}_3, \tilde{\Omega}, \tilde{\theta}_c, \tilde{\theta}_d]$, we verify

$$\begin{aligned} \dot{\tilde{x}} &= \dot{x} + \dot{x}_g = R_b v_r + C d = R_b R_g^T R_g v_r + C \tilde{d} \\ &= \tilde{R}_b \tilde{v}_r + C \tilde{d}, \end{aligned}$$

$$\begin{aligned} \dot{\tilde{v}}_r &= R_g \dot{v}_r = R_g(v_r \times (\bar{\omega} - b_\omega) + R_b^T g - \frac{1}{m} f_c b_3 + \frac{1}{m} f_d) \\ &= R_g v_r \times R_g(\bar{\omega} - b_\omega) + R_g R_b^T g - \frac{1}{m} f_c R_g b_3 + \frac{1}{m} R_g f_d \\ &= \tilde{v}_r \times (\tilde{\omega} - \tilde{b}_\omega) + \tilde{R}_b^T g - \frac{1}{m} f_c \tilde{b}_3 + \frac{1}{m} R_g f_d, \end{aligned}$$

$$\begin{aligned} \dot{\tilde{R}}_b &= \dot{R}_b R_g^T = R_b S(\bar{\omega} - b_\omega) R_g^T = R_b R_g^T R_g(\bar{\omega} - b_\omega) R_g^T \\ &= \tilde{R}_b S(\tilde{\omega} - \tilde{b}_\omega), \end{aligned}$$

$$\dot{\tilde{d}} = \dot{d} = A \tilde{d},$$

$$\dot{\tilde{b}}_\omega = R_g \dot{b}_\omega = 0,$$

$$\dot{\tilde{b}}_a = R_g \dot{b}_a = 0.$$

For the thrust term and drag term in $\dot{\tilde{v}}_r$, if there exist a $\Theta_{g1}(\cdot)$ such that $f_c(\Omega, v_r, \theta_c) = f_c(\Omega, R_g v_r, \Theta_{g1}(\theta_c)) = f_c(\Omega, \tilde{v}_r, \tilde{\theta}_c) = \tilde{f}_c$, and a $\Theta_{g2}(\cdot)$ such that $R_g f_d(v_r, \theta_d) = f_d(R_g v_r, \Theta_{g2}(\theta_d)) = f_d(\tilde{v}_r, \tilde{\theta}_d) = \tilde{f}_d$, $\dot{\tilde{v}}_r$ can then be written as

$$\dot{\tilde{v}}_r = \tilde{v}_r \times (\tilde{\omega} - \tilde{b}_\omega) + \tilde{R}_b^T g - \frac{1}{m} \tilde{f}_c \tilde{b}_3 + \frac{1}{m} \tilde{f}_d. \quad (13)$$

Thus, the dynamics of the transformed variables shown above satisfies (12), which means that (3) is invariant under the transformation (8).

Similarly, the measurement equation is G -equivariant if there exists a $\varrho_g(\cdot)$ such that

$$\varrho_g(h(X, U)) = h(\varphi_g(X), \psi_g(U)). \quad (14)$$

We verify

$$\begin{aligned} \varrho_g(h(X, U)) &= \begin{pmatrix} x + x_g \\ R_g(a + b_a) \\ R_g R_b^T B \end{pmatrix} = \begin{pmatrix} x + x_g \\ R_g(-\frac{1}{m} f_c b_3 + \frac{1}{m} f_d + b_a) \\ R_g R_b^T B \end{pmatrix} \\ &= \begin{pmatrix} \tilde{x} \\ -\frac{1}{m} f_c R_g b_3 + \frac{1}{m} R_g f_d + \tilde{b}_a \\ \tilde{R}_b^T B \end{pmatrix}. \end{aligned} \quad (15)$$

Similarly, according to the assumption $R_g f_d = \tilde{f}_d$ and $f_c = \tilde{f}_c$, we get

$$\varrho_g(h(X, U)) = \begin{pmatrix} \tilde{x} \\ -\frac{1}{m} \tilde{f}_c \tilde{b}_3 + \frac{1}{m} \tilde{f}_d + \tilde{b}_a \\ \tilde{R}_b^T B \end{pmatrix} = h(\varphi_g(X), \psi_g(U)). \quad (16)$$

Since $\varrho_g(h(X, U)) = h(\varphi_g(X), \psi_g(U))$, the measurement equation (4) is G -equivariant. \blacksquare

In our previous work [7], we identify $\Theta_{g1}(\cdot)$ and θ_c for each of the thrust models and establish their invariance properties. We now identify $\Theta_{g2}(\cdot)$ and θ_d for the general drag model (7) and establish its invariance property. We let $\theta_d = \{d_0, D_1, D_2\}$ and $\Theta_{g2}(\theta_d) = \{R_g d_0, R_g D_1 R_g^T, R_g D_2 R_g^T\}$. Then we verify

$$\begin{aligned} R_g f_d &= R_g(d_0 + D_1 v_r + D_2 |v_r| v_r) \\ &= R_g d_0 + R_g D_1 v_r + R_g D_2 |v_r| v_r \\ &= R_g d_0 + R_g D_1 R_g^T R_g v_r + |v_r| R_g D_2 R_g^T R_g v_r \\ &= f_d(R_g v_r, \Theta_{g2}(\theta_d)), \end{aligned} \quad (17)$$

which indicates that Proposition 1 holds.

B. IEKF Design

The invariance property established in the previous section allows us to design an invariant observer or filter for the wind estimation problem. We consider the system dynamics

$$\begin{aligned} \dot{x} &= q * v_r * q^{-1} + C d \\ \dot{v}_r &= v_r \times (\bar{\omega} - b_\omega) + q^{-1} * g * q - \frac{1}{m} f_c b_3 + \frac{1}{m} f_d \\ \dot{q} &= \frac{1}{2} q * (\bar{\omega} - b_\omega) \\ \dot{d} &= A d \\ \dot{b}_\omega &= 0 \\ \dot{b}_a &= 0 \end{aligned} \quad (18)$$

where we use the quaternion q as a representation of $R_b \in SO(3)$ and $*$ denotes the quaternion multiplication. In a quaternion multiplication, any vector in \mathbb{R}^3 is augmented to a quaternion with 0 being the scalar part.

For different thrust and drag models, the IEKF design will be different. We choose a polynomial thrust model and a quadratic drag model as the example since they are used in our experiments. For the polynomial thrust model, no parameters need to be transformed while for

the quadratic drag model, $\theta_d = D$ and $\Theta_{g2}(\theta) = R_g D R_g^T$. We follow the design procedure in [25] and [6].

The invariants I and the invariant output error E are given by

$$\begin{aligned} I = & (I_{v_r}, I_d, I_{b_\omega}, I_{b_a}, I_{\bar{\omega}}, I_{b_3}, I_\Omega, I_D) \\ = & (\hat{q} * \hat{v}_r * \hat{q}^{-1}, \hat{d}, \hat{q} * \hat{b}_\omega * \hat{q}^{-1}, \hat{q} * \hat{b}_a * \hat{q}^{-1}, \\ & \hat{q} * \bar{\omega} * \hat{q}^{-1}, \hat{q} * b_3 * \hat{q}^{-1}, \hat{q} * \Omega * \hat{q}^{-1}, R(\hat{q}) D R^T(\hat{q})) \end{aligned} \quad (19)$$

and

$$E = \varrho_{\gamma(\hat{X})}(y) - \varrho_{\gamma(\hat{X})}(\hat{y}) = T \begin{pmatrix} y_x - \hat{y}_x \\ y_a - \hat{y}_a \\ y_b - \hat{y}_b \end{pmatrix} = \begin{pmatrix} E_x \\ E_a \\ E_b \end{pmatrix}, \quad (20)$$

respectively, where T is given below with e_i , $i = 1, 2, 3$, forming the canonical frame of \mathbb{R}^3 :

$$T = \begin{pmatrix} e_i & 0 & 0 \\ 0 & \hat{q} * e_i * \hat{q}^{-1} & 0 \\ 0 & 0 & \hat{q} * e_i * \hat{q}^{-1} \end{pmatrix}. \quad (21)$$

Invariant frame is given by

$$\begin{aligned} W = & (D\varphi_{\gamma(X)}(X))^{-1} \frac{\partial}{\partial X} \\ = & \begin{pmatrix} e_i & 0 & 0 & 0 & 0 & 0 \\ 0 & \hat{q}^{-1} * e_i * \hat{q} & 0 & 0 & 0 & 0 \\ 0 & 0 & e_i * \hat{q} & 0 & 0 & 0 \\ 0 & 0 & 0 & e_i & 0 & 0 \\ 0 & 0 & 0 & 0 & \hat{q}^{-1} * e_i * \hat{q} & 0 \\ 0 & 0 & 0 & 0 & 0 & \hat{q}^{-1} * e_i * \hat{q} \end{pmatrix}. \end{aligned} \quad (22)$$

The invariant pre-observer is designed as

$$\begin{aligned} \dot{\hat{x}} &= \hat{q} * \hat{v}_r * \hat{q}^{-1} + C\hat{d} + \bar{\mathcal{L}}^x E \\ \dot{\hat{v}}_r &= \hat{v}_r \times (\bar{\omega} - b_\omega) + \hat{q}^{-1} * \mathbf{g} * \hat{q} - \frac{1}{m} f_c b_3 + \frac{1}{m} f_d \\ &+ \hat{q}^{-1} * (\bar{\mathcal{L}}^{v_r} E) * \hat{q} \\ \dot{\hat{q}} &= \frac{1}{2} \hat{q} * (\bar{\omega} - b_\omega) + (\bar{\mathcal{L}}^q E) * \hat{q} \\ \dot{\hat{d}} &= A\hat{d} + \bar{\mathcal{L}}^d E \\ \dot{\hat{b}}_\omega &= \bar{\mathcal{L}}^{b_\omega} E \\ \dot{\hat{b}}_a &= \bar{\mathcal{L}}^{b_a} E, \end{aligned}$$

where $\bar{\mathcal{L}}^*$ is the gain matrix

$$\bar{\mathcal{L}}^* = \begin{pmatrix} \bar{\mathcal{L}}_{x,1}^* & \bar{\mathcal{L}}_{a,1}^* & \bar{\mathcal{L}}_{b,1}^* \\ \bar{\mathcal{L}}_{x,2}^* & \bar{\mathcal{L}}_{a,2}^* & \bar{\mathcal{L}}_{b,2}^* \\ \bar{\mathcal{L}}_{x,3}^* & \bar{\mathcal{L}}_{a,3}^* & \bar{\mathcal{L}}_{b,3}^* \end{pmatrix} \in \mathbb{R}^{3 \times 9}, * \in \{x, v_r, q, d, b_\omega, b_a\} \quad (23)$$

The invariant state error η is defined as

$$\begin{aligned} \eta &= \varphi_{\gamma(\hat{X})}(X) - \varphi_{\gamma(\hat{X})}(\hat{X}) \\ &= \begin{pmatrix} x - \hat{x} \\ \hat{q} * (v_r - \hat{v}_r) * \hat{q}^{-1} \\ q * \hat{q}^{-1} \\ d - \hat{d} \\ \hat{q} * (b_\omega - \hat{b}_\omega) * \hat{q}^{-1} \\ \hat{q} * (b_a - \hat{b}_a) * \hat{q}^{-1} \end{pmatrix} = \begin{pmatrix} \eta_x \\ \eta_{v_r} \\ \eta_q \\ \eta_d \\ \eta_{b_\omega} \\ \eta_{b_a} \end{pmatrix}. \end{aligned} \quad (25)$$

We next differentiate (25) with respect to time and obtain the invariant error dynamics

$$\begin{aligned} \dot{\eta}_x &= \eta_q * \eta_{v_r} * \eta_q^{-1} + \eta_q * I_{v_r} * \eta_q^{-1} + C\eta_d - I_{v_r} - \bar{\mathcal{L}}^x E \\ \dot{\eta}_{v_r} &= \eta_q^{-1} * \mathbf{g} * \eta_q - \mathbf{g} - (\eta_{v_r} + I_{v_r}) \times \eta_{b_\omega} + I_{f_d} \\ &+ 2(\bar{\mathcal{L}}^q E) \times \eta_{v_r} - \bar{\mathcal{L}}^{v_r} E \\ \dot{\eta}_q &= -\frac{1}{2} \eta_q * \eta_{b_\omega} - \eta_q * (\bar{\mathcal{L}}^q E) \\ \dot{\eta}_d &= A\eta_d - (\bar{\mathcal{L}}^d E) \\ \dot{\eta}_{b_\omega} &= (I_{\bar{\omega}} - I_{b_\omega}) \times \eta_{b_\omega} + 2(\bar{\mathcal{L}}^q E) \times \eta_{b_\omega} - \bar{\mathcal{L}}^{b_\omega} E \\ \dot{\eta}_{b_a} &= (I_{\bar{\omega}} - I_{b_\omega}) \times \eta_{b_a} + 2(\bar{\mathcal{L}}^q E) \times \eta_{b_a} - \bar{\mathcal{L}}^{b_a} E, \end{aligned}$$

where

$$I_{f_d} = \frac{1}{m} \left(-\frac{1}{2} \rho I_D |I_{v_r} + \eta_{v_r}| (I_{v_r} + \eta_{v_r}) \right) - \frac{1}{m} \left(-\frac{1}{2} \rho I_D |I_{v_r}| I_{v_r} \right). \quad (26)$$

Linearizing around $\eta_x = 0, \eta_{v_r} = 0, \eta_q = [1, 0, 0, 0]^T, \eta_d = 0, \eta_{b_\omega} = 0, \eta_{b_a} = 0$ yields the state matrix needed for implementing the IEKF at time step k :

$$A_{lin} = \begin{pmatrix} 0 & I_3 & 0 & -2S(I_{v_r}) & C & 0 & 0 \\ 0 & \alpha & 0 & 2S(\mathbf{g}) & 0 & -S(I_{v_r}) & 0 \\ 0 & 0 & 0 & 0 & 0 & -\frac{1}{2} I_3 & 0 \\ 0 & 0 & 0 & 0 & A & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & S(I_{\bar{\omega}}) - S(I_{b_\omega}) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & S(I_{\bar{\omega}}) - S(I_{b_\omega}) \end{pmatrix} \quad (27)$$

where I_3 is the 3 by 3 identity matrix and

$$\alpha = \frac{1}{m} \left(-\frac{1}{2} \rho I_D \right) \frac{\partial (|I_{v_r} + \eta_{v_r}| (I_{v_r} + \eta_{v_r}) - (\eta_{v_r} + I_{v_r}) \times \eta_{b_\omega})}{\partial \eta_{v_r}}.$$

Linearizing the invariant output error E around $\eta_x = 0, \eta_{v_r} = 0, \eta_q = [1, 0, 0, 0]^T, \eta_d = 0, \eta_{b_\omega} = 0, \eta_{b_a} = 0$ yields the output matrix H_{lin} needed for the IEKF:

$$H_{lin} = \begin{pmatrix} I_3 & 0 & 0 & 0 & 0 & 0 \\ 0 & \beta & 0 & 0 & 0 & I_3 \\ 0 & 0 & 2S(B) & 0 & 0 & 0 \end{pmatrix} \quad (28)$$

where

$$\beta = \frac{1}{m} \left(-\frac{1}{2} \rho I_D \right) \frac{\partial (|I_{v_r} + \eta_{v_r}| (I_{v_r} + \eta_{v_r}))}{\partial \eta_{v_r}}.$$

The A_{lin} and H_{lin} matrices are developed for the IEKF algorithm given in Algorithm 1. To derive the IEKF algorithm, the process noise $w = [w_x^T, w_\omega^T, w_f^T, w_d^T, w_{b_\omega}^T, w_{b_a}^T]^T$ is added to the invariant observer (23). The IEKF algorithm employs (23) with $L = 0$ and the linearized matrix A_{lin} in (27) to propagate the estimates and the covariance between the prediction step. For the correction step, we select L as the Kalman filter gain based on A_{lin} and H_{lin} and update \hat{X} using L and the measurements $y_m = (y_{x_m}^T, y_{a_m}^T, y_{b_m}^T)^T$. We note that the invariant state error rotates the conventional estimation error to another frame.

To represent the uncertainty more accurately, the initial state covariance and process noise and measurement noise covariance matrices, P_0 , Q , and R , need to be transformed to the invariant frame from the original frame [32], [26]. The transformed matrices are given by: $P_{0_{rot}} = W^T P_0 W$, $Q_{rot} = N Q N^T$, and $R_{rot} = T R T^T$

where $N = \frac{\partial \dot{\eta}}{\partial w}$ and $\dot{\eta}$ is the invariant error dynamics (25) with process noise, T and W are given in (21) and (22), respectively.

Algorithm 1: The proposed IEKF

```

1 Initialize  $X_0, P_0$  in the original coordinates.
2  $P = W^T P_0 W$ 
3 for  $k = 1$  to  $n$  do
4   Prediction: in between measurements ( $t \in [t_{k-1}, t_k]$ )
5   Propagate  $\hat{X} = f(\hat{X}, U)$  to get  $\hat{X}_k^-$ 
6   Compute  $A_{lin}$  from (27)
7   Compute  $Q_{rot} = N Q N^T$ 
8   Propagate  $\dot{P} = A_{lin} P + P A_{lin}^T + Q_{rot}$  to get
      $P_k^-$ 
9   Correction: at the  $k^{th}$  sensor measurement ( $t = t_k$ )
10   $R_{rot} = T R T^T$ 
11  Compute  $H_{lin}$  from (28)
12   $L_k = P_k^- H_{lin}^T (H_{lin} P_k^- H_{lin}^T + R_{rot})^{-1}$ 
13   $\hat{X}_k^+ = \hat{X}_k^- + W L_k T (y_{m,k} - h(\hat{X}_k^-, U))$ 
14   $P_k^+ = (I - L_k H_{lin}) P_k^-$ 
15 end

```

C. Observability Analysis

We consider the system dynamics (18) with a polynomial thrust model and a quadratic drag model. Since we use the unit quaternion as representation of the rotation, there is an extra constraint $q^T q = 1$. We extend this constraint to the measurement equation

$$y = \begin{pmatrix} -\frac{1}{m} f_c b_3 - \frac{1}{m} \frac{1}{2} \rho D |v_r| v_r + b_a \\ q^{-1} * B * q \\ q^T * q \end{pmatrix}. \quad (29)$$

The first order and second order time derivative of the measured equations are represented as \dot{y} and \ddot{y} respectively. Then we take derivative of y , \dot{y} , and \ddot{y} with respect to states X and construct an observability matrix as [33]

$$\mathcal{O} = [H_0^T \ H_1^T \ H_2^T]^T \quad (30)$$

where $H_0 = \frac{\partial y}{\partial X}$, $H_1 = \frac{\partial \dot{y}}{\partial X}$, $H_2 = \frac{\partial \ddot{y}}{\partial X}$. Using MATLAB's symbolic toolbox, the column rank of the 30×19 -dimensional observability matrix is 19, except at certain points in the state space, such as hover.

In outdoor experiments, the quadcopter is typically hovering at a fixed location. To ensure observability, we will pre-calibrate the IMU biases and verify the wind estimation performance of EKF and IEKF without IMU biases.

IV. Outdoor experiments

A. experimental setup

We design and develop a larger size quadcopter SK8 as shown in Fig. 1, which is used to perform the hover

tests and collect the data in the outdoor environment. SK8 is a self-built quadcopter from Oklahoma State University Unmanned Systems Research Institute (USRI). The SK8 quadcopter characteristics is shown in Table II. There is a TriSonica Mini anemometer mounted on the top of a gimbal to collect measured wind data for comparison. The Pixhawk Orange Cube is the flight controller hardware, but we have Ardupilot software installed which allows us to use Mission Planner as the ground control station to conduct custom auto missions.



Fig. 1. SK8 quadcopter.

We have completed two separate outdoor experiments. For each experiment, the experimental locations and date are carefully selected based on the safety and suitable wind field concern. The first one occurred at Oklahoma State University (OSU) Unmanned Aircraft Flight Station (UAFS) on Oct.17 2022 as shown in Fig. 2, and it includes three hover tests at 11:46:35, 12:05:49 and 12:25:11 on Oct.17.2022. For the second experiment, it occurred at Oklahoma State University campus on Dec.01 2022 as shown in Fig. 3. There were two hover tests at 10:55:04 and 11:10:25 on Dec.01.2022. In Table III, we choose the dataset at 12-25-11 Oct.17.2022 as the calibrate dataset. We denote the rest hover tests as test1, test2, test3 and test4 in chronological order to validate the various wind estimators. The statistics of the wind measurements from anemometer is also shown in Table III. For each hover test, we use a logging tool provided by Pixhawk4 to log the GPS data at $5Hz$, IMU (including gyroscope, accelerometer and magnetometer) data at $200Hz$, in-built EKF estimated states at $10Hz$ and PWM data at $10Hz$ separately. Also, we use the Arduino board to log the Anemometer data at $5Hz$.

B. calibration methodology

To estimate the wind, a thrust model is required. For the large scale quadcopter SK8 equipped with large-sized blades, building a thrust stand and conducting the wind tunnel test is challenging and expensive. Therefore, we propose a projection thrust model and a polynomial thrust model to obtain the thrust from the calibration dataset.



Fig. 2. SK8 experiment1



Fig. 3. SK8 experiment2

TABLE II
SK8 quadcopter characteristics

System characteristics	Descriptions	Frequency
mass	7.910kg	N/A
materials	aluminum & wood	N/A
motor to motor length	1.0414 m	N/A
motors	T-Motor MN505-S 320 kV	N/A
propellers	T-motor MF2009	N/A
autopilot	Pixhawk Orange Cube	N/A
GPS	Here2 GPS unit	5Hz
IMU	Pixhawk in-built IMU	200Hz
LiDAR	Lightware SF-11C	20 Hz
Anemometer	Trisonica Mini	5 Hz

For the projection thrust model, we first define z as the altitude of the quadcopter and $R_{b,3}$ as the third row of the rotation matrix R_b . Consider the translational dynamics of the quadcopter in the z direction in the inertial frame

$$\ddot{z} = \frac{1}{m} R_{b,3} f_c b_3 - g - \frac{1}{m} R_{b,3} f_d. \quad (31)$$

We make the assumption that the drag force in z direction of the body frame is small compared to the thrust. Thus we ignore the drag force term in (31) to obtain

$$\ddot{z} \approx \frac{f_c}{m} \cos \theta \cos \phi - g \quad (32)$$

where ϕ and θ are roll and pitch angles of the quadcopter respectively. Suppose that there is little vertical motion when the quadcopter is hovering. We can obtain a simple projection thrust model as

$$f_c = \frac{mg}{\cos \theta \cos \phi}. \quad (33)$$

For the polynomial thrust model, by using thrust data obtained from (33) based on the calibration dataset, we fit a polynomial thrust model [34]

$$\begin{aligned} f_c &= f(volt, p_i) \\ &= a_1 - a_2 * \sum_{i=1}^4 p_i - a_3 * volt + a_4 * \sum_{i=1}^4 p_i^2 + a_5 * \sum_{i=1}^4 p_i * volt \end{aligned} \quad (34)$$

where $volt$ is the normalized battery voltage, p_i is the normalized PWM command for motor i ($i = 1, 2, 3, 4$), and $[a_1 \ a_2 \ a_3 \ a_4 \ a_5] = [78.2270 \ 1.0525 \ 0.6386 \ 0.8445 \ 1.1095]$.

For the drag coefficient matrix calibration, we have considered both linear and quadratic drag models. The drag force vector in the body frame is obtained as

$$f_d = R_b^T (R_b \dot{v}_r - m\mathbf{g} - R_b(f_c b_3)). \quad (35)$$

A linear regression with the basis function below is used to find the drag coefficient matrix

$$f_d = D\phi(v_r) \quad (36)$$

where $\phi(v_r) = v_r$ for the linear drag model and $\phi(v_r) = |v_r|v_r$ for the quadratic drag model. Through the linear regression, the drag coefficient matrix is identified as $D = \text{diag}([2.74 \ 1.44 \ 2.67])$ for the linear drag model and $D = \text{diag}([0.36 \ 0.28 \ 0.35])$ for the quadratic drag model. The IMU biases are calibrated by Mission Planner before every tests.

C. TEKF

Motivated by [35], we also design and implement a translational EKF (TEKF) for wind estimation. Such EKF only considers the translational dynamics of the system as follows

$$\begin{aligned} \dot{x} &= R_b v_r + C d \\ \dot{v}_r &= v_r \times \omega + R_b^T \mathbf{g} - \frac{1}{m} f_c b_3 + \frac{1}{m} f_d \\ \dot{d} &= A d. \end{aligned} \quad (37)$$

It only uses GPS measurements, i.e., $y = x$. TEKF is based on the ease of design, tuning, and the better computational efficiency compared to the full EKF or full IEKF. The attitude information R_b can be obtained from the Ardupilot in-built filter or self-designed attitude and heading reference system (AHRS) filter. The TEKF is cascaded to the attitude filter.

D. experimental results

We assume that the wind velocity consists of a constant velocity and a turbulent velocity. Therefore we set A and C matrices used in the TEKF/EKF/IEKF to $A = 0_{3 \times 3}$ and $C = \text{diag}(1, 1, 1)$.

Table IV shows mean error and RMSE of wind horizontal speed estimation for each tests. We trim the data during the take-off and landing stages of all tests before we use our estimators to make sure that we only use hovering data. The TEKF, EKF and IEKF consider the system dynamics with a polynomial thrust and a quadratic drag model. Note that the dataset used for finding model parameters is collected successively with test1 and test2 on Oct.17.2022. The test3 and test4 are conducted on Dec.01.2022. We can see that for static mapping (stMapping) and TEKF, the estimation performance has a large difference between test1/test2 and test3/test4, which shows the poor generalization ability of static mapping and TEKF. The performance of EKF and IEKF is more consistent for all tests, which means that their generalization ability is better. Overall, the mean error and RMSE of EKF/IEKF is significantly less than that of stMapping/TEKF in test3 and test4. From Fig. 4, we can see that all four methods performs good and similarly for test1 while the wind estimation performance of EKF/IEKF is significantly better than stMapping/TEKF

TABLE III
SK8 outdoor experiments

Test	Time	Location	Duration (s)	Mean Wind (m/s)	Max wind (m/s)	Min wind (m/s)
Calibration test	12:25:11 Oct.17.2022	OSU UAFS	600	7.0039	10.6957	3.7444
test1	11:46:35 Oct.17.2022	OSU UAFS	1000	6.8932	10.9659	1.6240
test2	12:05:49 Oct.17.2022	OSU UAFS	1000	6.1288	10.2693	1.3787
test3	10:55:04 Dec.01.2022	OSU campus	780	8.4859	16.4302	1.8024
test4	11:10:25 Dec.01.2022	OSU campus	345	7.5732	13.3935	0.4252

for test3. As for the difference between EKF and IEKF, we verify that IEKF has less error and converges faster during the transient stage (first 10 seconds) compared to EKF.

TABLE IV
Horizontal wind speed estimation.

estimator	error metric	test1	test2	test3	test4
stMapping	Mean	0.0332	0.3485	2.2901	1.9475
	RMSE	0.9763	0.9431	2.5922	2.3115
EKF	Mean	0.5102	0.2070	0.6675	0.2783
	RMSE	1.1605	1.0526	1.3128	1.2339
IEKF	Mean	0.4963	0.2056	0.6339	0.3261
	RMSE	1.1201	0.9908	1.2953	1.2296
TEKF	Mean	0.0024	0.6777	2.0775	1.8726
	RMSE	1.1757	1.2669	2.3990	2.2619

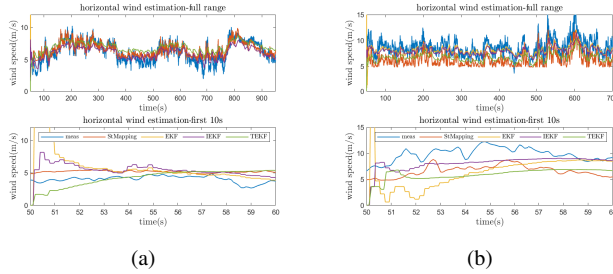


Fig. 4. Horizontal wind speed estimation. (a): Test1. (b): Test3.

We list the main references which contain model-based methods and outdoor experiments in Table V. This table compares the experimental wind estimation results under their own wind speed condition. For our experiments, IEKF shows the best wind estimation performance on test2 with the value of RMSE 0.99 m/s under the wind speed between 1.62 – 10.97 m/s. It is inconclusive whose work provides the most accurate estimates of the wind since there are many uncontrollable factors, such as various wind conditions, wind measurement setting, and quadcopter types. However, in terms of overall accuracy, we conclude that the model-based wind estimation results presented in [1], [4] and this paper compare favorably with non-model-based results.

E. Monte Carlo simulations

To further verify the effectiveness of TEKF/EKF/IEKF, we conduct Monte-Carlo (MC)

TABLE V

Wind estimation performance of various studies (the experimental result is chosen from the best result of each reference)

Authors	Experiment result	Wind condition
Gonzalez-Rochaz et al. [1]	RMSE of 0.4 m/s	1 – 5 m/s
Meier et al. [9]	Bias of 0.29 m/s	0 – 3 m/s
Simmma et al. [4]	RMSE of 0.26 m/s	0 – 5 m/s
McConville et al. [10]	RMSE of 0.53 m/s	0 – 6.5 m/s

simulations of 50 runs for each test. The following parameters are used in EKF/IEKF MC tests:

$$\begin{aligned}\mu_0 &= [0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]^T \\ P_0 &= \text{diag}(1^2, 1^2, 1^2, 1^2, 1^2, 1^2, 1^2, 1^2, 1^2, 1^2, 1^2, 0.1^2) \\ \hat{X}_0 &\sim \mathcal{N}(\mu_0, P_0)\end{aligned}$$

where \hat{X}_0 represents the initial states for the EKF and the IEKF and $\hat{X}_0 \sim \mathcal{N}(\mu_0, P_0)$ means that the initial state is sampled from a Gaussian distribution with the mean μ_0 and the covariance matrix P_0 . For TEKF, we do not consider the quaternion. The initial states are obtained as

$$\begin{aligned}\mu_0 &= [0, 0, 0, 0, 0, 0, 0, 0, 0]^T \\ P_0 &= \text{diag}(1^2, 1^2, 1^2, 1^2, 1^2, 1^2, 1^2, 1^2, 0.1^2) \\ \hat{X}_0 &\sim \mathcal{N}(\mu_0, P_0).\end{aligned}$$

Table VI shows wind estimation RMSE of estimators which include the polynomial thrust model and the quadratic drag model. Note that the RMSE defined in Monte Carlo experiments is as $RMSE_i(t) = \sqrt{\frac{\sum_{j=1}^n (X_{i,j}(t) - \hat{X}_{i,j}(t))^2}{n}}$ where $X_{i,j}(t)$ represent the i th element of states at time t in the j th run while $\hat{X}_{i,j}(t)$ represents the estimates of $X_{i,j}(t)$. We see that the RMSE of the TEKF is significantly greater than that of EKF/IEKF for all four tests, which proves that TEKF is not an optimal estimator compared to EKF/IEKF. For IEKF, the RMSE is smaller than that of EKF for all tests, which can be explained by the superiority of IEKF's transient stability. From Fig. 5, we observe the above properties intuitively: (1) The RMSE of the TEKF is greater than EKF/IEKF; (2) EKF has a larger transient RMSE compared to IEKF but behaves almost the same as IEKF in the steady state.

Table VI, Table VII, and Table VIII show wind estimation RMSE of estimators with the polynomial thrust/quadratic drag models, the projection thrust/quadratic drag models, and the polynomial

thrust/linear drag models, respectively. Through the comparison between Table VI and Table VIII, we see that the estimators with the linear drag model shows higher RMSE for all 4 tests, which implies that the quadratic drag model may represent the drag force better. To check the impact between the two proposed thrust models on wind estimation performance, we compare Table VI and Table VII. The estimators with projection and polynomial thrust models have similar RMSE of wind estimation. Therefore, we conclude that estimators with projection and polynomial thrust models perform similarly for horizontal wind estimation.

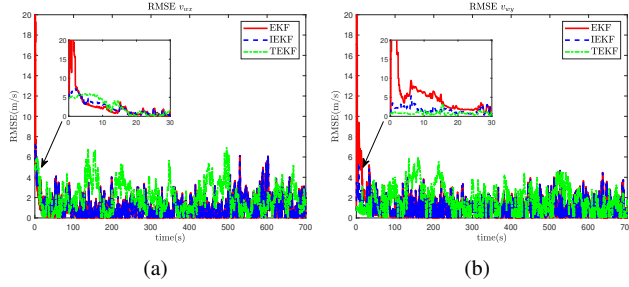


Fig. 5. RMSE comparison of EKF/IEKF/TEKF for test3. (a): RMSE v_{wx} . (b): RMSE v_{wy} .

TABLE VI

Wind estimation RMSE of estimators with polynomial thrust model and quadratic drag model

estimator type	wind vector	test1	test2	test3	test4
EKF	v_{wx}	1.2239	0.9998	1.2096	1.0521
EKF	v_{wy}	1.7251	1.4280	1.4983	1.3643
IEKF	v_{wx}	1.2216	0.9919	1.0868	1.0437
IEKF	v_{wy}	1.5467	1.3386	1.1416	1.3441
TEKF	v_{wx}	1.8033	1.7858	2.1932	2.3575
TEKF	v_{wy}	1.9260	2.1774	1.8051	1.4656

TABLE VII

Wind estimation RMSE of estimators with projection thrust model and quadratic drag model

estimator type	wind vector	test1	test2	test3	test4
EKF	v_{wx}	1.2696	0.9873	1.1582	1.1626
EKF	v_{wy}	1.5658	1.4352	1.3549	1.5008
IEKF	v_{wx}	1.1905	0.9676	1.0952	1.1023
IEKF	v_{wy}	1.2875	1.3354	1.1266	1.3413
TEKF	v_{wx}	1.8055	1.7914	2.2021	2.3602
TEKF	v_{wy}	1.9159	1.1722	1.8012	1.4646

V. Conclusions

In this paper, we focus on states and horizontal wind estimation for a quadcopter. We consider a general form of system dynamics including drag, thrust models and IMU biases. We establish the invariance of system dynamics and the equivariance of measurements by designing transformations to states, inputs, and measurements. Then

TABLE VIII

Wind estimation RMSE of estimators with polynomial thrust model and linear drag model

estimator type	wind vector	test1	test2	test3	test4
EKF	v_{wx}	1.7492	1.6921	1.1528	1.5973
EKF	v_{wy}	2.0007	1.5714	1.5794	2.5795
IEKF	v_{wx}	1.6652	1.6300	1.1076	1.1907
IEKF	v_{wy}	1.8250	1.3953	1.4798	1.7455
TEKF	v_{wx}	2.0321	2.0472	2.9129	2.5410
TEKF	v_{wy}	2.1183	2.6078	2.0532	1.6733

we design an IEKF and validate it through outdoor hover experiments. Our experimental results indicate that the EKF and the IEKF perform better when compared to the benchmark methods, such as static mapping method and TEKF. In addition, IEKF shows less transient error and converges faster when compared to EKF.

Future work includes estimating a spatial-temporal wind field and fusing wind information from multiple quadcopters.

REFERENCES

- [1] J. González-Rocha, C. A. Woolsey, C. Sultan, and S. F. De Wekker Sensing wind from quadrotor motion *Journal of Guidance, Control, and Dynamics*, vol. 42, no. 4, pp. 836–852, 2019.
- [2] J. González-Rocha, S. F. De Wekker, S. D. Ross, and C. A. Woolsey Wind profiling in the lower atmosphere from wind-induced perturbations to multirotor UAS *Sensors*, vol. 20, no. 5, p. 1341, 2020.
- [3] A. R. Segales *et al.* The copertsonde: an insight into the development of a smart unmanned aircraft system for atmospheric boundary layer research *Atmospheric Measurement Techniques*, vol. 13, no. 5, pp. 2833–2848, 2020.
- [4] M. Simma, H. Mjøen, and T. Boström Measuring wind speed using the internal stabilization system of a quadrotor drone *Drones*, vol. 4, no. 2, p. 23, 2020.
- [5] S. Allison, H. Bai, and B. Jayaraman Wind estimation using quadcopter motion: A machine learning approach *Aerospace Science and Technology*, vol. 98, p. 105699, 2020.
- [6] H. Chen, H. Bai, and C. N. Taylor Invariant-EKF design for quadcopter wind estimation In *2022 American Control Conference (ACC)*. IEEE, 2022, pp. 1236–1241.
- [7] H. Chen and H. Bai Incorporating thrust models for quadcopter wind estimation *IFAC-PapersOnLine*, vol. 55, no. 37, pp. 19–24, 2022.
- [8] G. Hattenberger, M. Bronz, and J.-P. Condomines Estimating wind using a quadrotor *International Journal of Micro Air Vehicles*, vol. 14, p. 17568293211070824, 2022.
- [9] K. Meier, R. Hann, J. Skaloud, and A. Garreau Wind estimation with multirotor UAVs *Atmosphere*, vol. 13, no. 4, p. 551, 2022.
- [10] A. McConville, T. S. Richardson, and P. Moradi Comparison of multirotor wind estimation techniques through conventional on-board sensors In *AIAA SCITECH 2022 Forum*, 2022, p. 0411.
- [11] A. Tabassum, R. K. Vuppala, H. Bai, and K. Kara

- Variance reduction of quadcopter trajectory tracking in turbulent wind
IFAC-PapersOnLine, vol. 54, no. 20, pp. 102–107, 2021.
- [12] J. González-Rocha *et al.*
Sensing atmospheric flows in aquatic environments using a multirotor small uncrewed aircraft system (sUAS)
Environmental Science: Atmospheres, 2023.
- [13] J.-Y. Wang, B. Luo, M. Zeng, and Q.-H. Meng
A wind estimation method with an unmanned rotorcraft for environmental monitoring tasks
Sensors, vol. 18, no. 12, p. 4504, 2018.
- [14] R. T. Palomaki, N. T. Rose, M. van den Bossche, T. J. Sherman, and S. F. De Wekker
Wind estimation in the lower atmosphere using multirotor aircraft
Journal of Atmospheric and Oceanic Technology, vol. 34, no. 5, pp. 1183–1191, 2017.
- [15] S. Prudden, A. Fisher, M. Marino, A. Mohamed, S. Watkins, and G. Wild
Measuring wind with small unmanned aircraft systems
Journal of Wind Engineering and Industrial Aerodynamics, vol. 176, pp. 197–210, 2018.
- [16] I. De Boisblanc *et al.*
Designing a hexacopter for the collection of atmospheric flow data
In *2014 Systems and Information Engineering Design Symposium (SIEDS)*. IEEE, 2014, pp. 147–152.
- [17] C. A. Wolf *et al.*
Wind data collection techniques on a multi-rotor platform
In *2017 Systems and Information Engineering Design Symposium (SIEDS)*. IEEE, 2017, pp. 32–37.
- [18] P. P. Neumann and M. Bartholmai
Real-time wind estimation on a micro unmanned aerial vehicle using its inertial measurement unit
Sensors and Actuators A: Physical, vol. 235, pp. 300–310, 2015.
- [19] M. Marino, A. Fisher, R. Clothier, S. Watkins, S. Prudden, and C. S. Leung
An evaluation of multi-rotor unmanned aircraft as flying wind sensors
International Journal of Micro Air Vehicles, vol. 7, no. 3, pp. 285–299, 2015.
- [20] D. Crowe, R. Pamula, H. Y. Cheung, and S. F. De Wekker
Two supervised machine learning approaches for wind velocity estimation using multi-rotor copter attitude measurements
Sensors, vol. 20, no. 19, p. 5638, 2020.
- [21] S. Zimmerman, M. Yermi, R. Nagamune, and S. Rogak
Wind estimation by multirotor dynamic state measurement and machine learning models
Measurement, vol. 198, p. 111331, 2022.
- [22] M. O’Connell *et al.*
Neural-fly enables rapid learning for agile flight in strong winds
Science Robotics, vol. 7, no. 66, p. eabm6597, 2022.
- [23] J. Moyano Cano
Quadrotor uav for wind profile characterization
Master’s thesis, 2013.
- [24] A. Shastry and D. A. Paley
Uav state and parameter estimation in wind using calibration trajectories optimized for observability
IEEE Control Systems Letters, vol. 5, no. 5, pp. 1801–1806, 2020.
- [25] S. Bonnabel, P. Martin, and P. Rouchon
Symmetry-preserving observers
IEEE Transactions on Automatic Control, vol. 53, no. 11, pp. 2514–2526, 2008.
- [26] K. Coleman, H. Bai, and C. N. Taylor
Extended invariant-EKF designs for state and additive disturbance estimation
Automatica, vol. 125, p. 109464, 2021.
- [27] K. S. Phogat and D. E. Chang
Invariant extended Kalman filter on matrix lie groups
Automatica, vol. 114, p. 108812, 2020.
- [28] R. Hartley, M. Ghaffari, R. M. Eustice, and J. W. Grizzle
Contact-aided invariant extended Kalman filtering for robot state estimation
The International Journal of Robotics Research, vol. 39, no. 4, pp. 402–430, 2020.
- [29] M. Xu, Y. Song, Y. Chen, S. Huang, and Q. Hao
Invariant EKF based 2D active SLAM with exploration task
In *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 5350–5356.
- [30] P. van Goor, T. Hamel, and R. Mahony
Equivariant filter (EqF): A general filter design for systems on homogeneous spaces
In *2020 59th IEEE Conference on Decision and Control (CDC)*. IEEE, 2020, pp. 5401–5408.
- [31] A. Fornasier, Y. Ng, R. Mahony, and S. Weiss
Equivariant filter design for inertial navigation systems with input measurement biases
In *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 4333–4339.
- [32] A. Barrau and S. Bonnabel
The invariant extended kalman filter as a stable observer
IEEE Transactions on Automatic Control, vol. 62, no. 4, pp. 1797–1812, 2016.
- [33] R. Hermann and A. Krener
Nonlinear controllability and observability
IEEE Transactions on automatic control, vol. 22, no. 5, pp. 728–740, 1977.
- [34] G. Shi, W. Hönig, X. Shi, Y. Yue, and S.-J. Chung
Neural-swarm2: Planning and control of heterogeneous multirotor swarms using learned interactions
IEEE Transactions on Robotics, vol. 38, no. 2, pp. 1063–1079, 2021.
- [35] D. Kingston and R. Beard
Real-time attitude and position estimation for small UAVs using low-cost sensors
In *AIAA 3rd “Unmanned Unlimited” Technical Conference, Workshop and Exhibit*, 2004, p. 6488.



Hao Chen received the B.E. degree in Automation from Southwest University of Science and Technology, Mianyang, Sichuan, China in 2019. He is currently working toward the Ph.D. degree in Mechanical and Aerospace Engineering with the School of Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, US. His research interests include estimation, control, and data fusion of unmanned aerial vehicles.



He Bai received his B.Eng. degree from the Department of Automation at the University of Science and Technology of China, Hefei, China, in 2005, and the M.S. and Ph.D. degrees in Electrical Engineering from Rensselaer Polytechnic Institute in 2007 and 2009, respectively. From 2009 to 2010, he was a Post-doctoral Researcher at the Northwestern University, Evanston, IL. From 2010 to 2015, he was a Senior Research and Development

Scientist at UtopiaCompression Corporation. He was the Principal Investigator for a number of research projects on sense-and-avoid, cooperative target tracking, and target handoff in GPS-denied environments. In 2015, he joined the Mechanical and Aerospace Engineering Department at Oklahoma State University, where he is now an associate professor. He has published over 100 peer-reviewed journal and conference papers related to control, estimation, learning, and robotics and a research monograph “Cooperative control design: a systematic passivity-based approach” in Springer. He holds one patent on monocular passive

ranging. His research interests include multi-agent systems, nonlinear estimation and sensor fusion, path planning, intelligent control, and GPS-denied navigation.