


Article

Low-Cost Sensors State Estimation Algorithm for a Small Hand-Launched Solar-Powered UAV

An Guo ¹, Zhou Zhou ^{1,*}, Xiaoping Zhu ² and Fan Bai ¹

¹ School of Aeronautics, Northwestern Polytechnical University, Xi'an 710072, China; guoanuav@mail.nwpu.edu.cn (A.G.); bfaner@mail.nwpu.edu.cn (F.B.)

² Science and Technology on UAV Laboratory, Northwestern Polytechnical University, Xi'an 710072, China; zhuxp@nwpu.edu.cn

* Correspondence: zhouzhou@nwpu.edu.cn; Tel.: +86-139-9131-2003

Received: 4 September 2019; Accepted: 17 October 2019; Published: 24 October 2019



Abstract: In order to reduce the cost of the flight controller and improve the control accuracy of solar-powered unmanned aerial vehicle (UAV), three state estimation algorithms based on the extended Kalman filter (EKF) with different structures are proposed: Three-stage series, full-state direct and indirect state estimation algorithms. A small hand-launched solar-powered UAV without ailerons is used as the object with which to compare the algorithm structure, estimation accuracy, and platform requirements and application. The three-stage estimation algorithm has a position accuracy of 6 m and is suitable for low-cost small, low control precision UAVs. The precision of full-state direct algorithm is 3.4 m, which is suitable for platforms with low-cost and high-trajectory tracking accuracy. The precision of the full-state indirect method is similar to the direct, but it is more stable for state switching, overall parameters estimation, and can be applied to large platforms. A full-scaled electric hand-launched UAV loaded with the three-stage series algorithm was used for the field test. Results verified the feasibility of the estimation algorithm and it obtained a position estimation accuracy of 23 m.

Keywords: low-cost sensor; state estimation; extended Kalman filter (EKF); three-stage series; full-state direct; full-state indirect; model calibration

1. Introduction

Solar-electrically powered fixed-wing unmanned aerial vehicles (UAVs) promise significantly increased flight endurance over pure electrically or even gas-powered aerial vehicles. Large-scale disaster relief, meteorological surveys in remote areas, and continuous border or wildlife protection benefit in particular from the multi-hour continuous flight capability provided by these robotic systems [1]. Traditional solar-powered UAVs require runways or complicated catapults for take-off and landing which are not commonly available in most field areas. The large size makes it inconvenient to apply and carry. The size of the UAV can be reduced to a wingspan of 2 to 5 m. With its excellent aerodynamic shape and lightweight structure, the hand-launched solar-powered UAV has great advantages in convenient application, long-endurance, and various mission execution [2–4].

Smaller-scale solar-powered UAVs are mostly designed for low-altitude long-endurance (LALE) applications [5]. Though faced with the more challenging meteorological phenomena of the lower atmosphere, low-altitude UAVs provide great potential of higher resolution imaging, reduced complexity and cost and simplified handling, for example, through hand-launched take-off. As a verification scheme, SoLong [3] has achieved a continuous 48-h flight using solar power and actively seeking out thermal updrafts, and Skysailor [4] has achieved a continuous 27-h solar-powered flight without the use of thermals, but both of these require a pilot to control them. AtlantikSolar [5] has

completed an 81-h autonomous flight, except for launch and landing, which created the current unofficial flight endurance world record for all aircraft below 50 kg total mass. The current trend of the LALE solar-powered UAV is from the research stage to the real-life mission application and civil field [6].

The manufacturing cost is a crucial factor in the marketization of a UAV. It consists of body frame, energy system, engine system, flight controller, and payload. Except for the controller, the cost of other systems is relatively fixed. The industrial flight controller consists of high-precision sensors that ensure the UAV has sufficient control precision and trajectory tracking capability, but its high price accounts for a large proportion of the entire cost. In contrast, the flight controller consisting of a low-cost micro-electro-mechanical system (MEMS), inertial measurement unit (IMU), magnetometer, GPS and barometer has a great price advantage, but it lacks measurement accuracy and long-time reliability, and cannot be directly applied to a solar-powered UAV platform [7]. Therefore, it is a key technology to reduce the cost of UAV manufacturing by using a suitable state estimation algorithm for long endurance stable estimation of low-cost sensors [8]. The solar-powered UAV has the characteristics of low flight speed and long-endurance, but the measurement error of low-cost sensors will be further amplified with time accumulation and temperature changes. By using the appropriate state estimation algorithm, the measurement accuracy of the low-cost sensor is effectively improved to meet the trajectory tracking accuracy requirements [9], for example, a four-sided route with an area of 1 km² has an acceptable tracking error of approximately 30 m (30 m/km²), and a heading error of nearly 13 degrees, thereby reducing the cost of the flight controller.

As a bridge between sensor measurement and controller calculation, the estimated state is fed back to the flight controller through the fusion of measurement of each sensor, and its accuracy will directly affect the positioning accuracy and control effect of the UAV. The nonlinearity and cross-axis sensitivity of industrial-grade sensors is significantly better than the low-cost sensors, as shown in Table 1. When the solar-powered UAV is equipped with a low-cost sensor, how to balance the error of the sensors for a long time to ensure the stable estimation of the state during the mission is the key to state estimation. With a large number of applications of low-cost flight controller platforms on miniature air vehicles, relevant research has also made some progress [10].

Dynamic observation theory and nonlinear filtering have proven to be effective in UAV state estimation. The complementary filter is a kind of data fusion algorithm based on frequency domain [11]. It uses gyroscope data in a short time, and regularly uses accelerometer and magnetometer data to correct the gyroscope. The Kalman filter (KF) is an optimal filter based on the state space method in the time domain, which resolves and eliminates noise according to the statistical characteristics of the noise measurement. For example, Extended Kalman filter (EKF), Unscent Kalman filter (UKF). The EKF method is the most widely used, and the core is to linearize the measurement model [12]. Different structures can be derived according to the type of sensor and the expression of the attitude angle. The UKF uses a deterministic approach to calculate the mean value and covariance, which results in approximations accurate to the third order (Taylor series expansions) [13]. However, the nonlinear method is computationally intensive and is not suitable for application on a low-cost flight controller.

Heikki [14] proposed a novel altitude estimation algorithm for fusing triaxial gyroscope and accelerometer measurements based on the EKF method. By introducing a variable covariance matrix and a low-cost, temperature-based calibration method, the robustness of the estimation results is improved. Michael [15] proposed an altitude determination system for small quadrotor UAV using consumer-grade Components-Off-The-Shelf (COTS). The exchange of estimated information is achieved by two extended Kalman filters without magnetometer. Nak [16] used an invariant extended Kalman filter (IEKF) for UAV navigation, based on EKF. Flight tests show that the method can be applied to rotary- and fixed-wing UAV platforms. Hu [17] proposed the external acceleration roll angle Kalman filter (EARAKF), which achieved a stable estimation of the roll angle during continuous overload with low-cost sensors.

Table 1. Various sensors with measurement performance.

Accel	MPU6050	Ellipse	Gyro	MPU6050	Ellipse	Magneto	LSM303D	Barometer	MS5611	MPXV7002	GPS	NEO-M8N
Dynamic range g	±16	±8	Dynamic range °/sec	±2000	±450	Dynamic range gauss	±12	Dynamic range mbar kPa	10–1200	±2	Horizontal accuracy m	±2.5
Nonlinearity %	0.5	0.2	Nonlinearity%	0.2	0.01	Nonlinearity %	0.5	Accuracy mbar %V _{fs}	±1.5	±2.5	Update rate Hz	10
Cross-Axis Sensitivity %	±2	0.05	Cross-Axis Sensitivity%	±2	0.05	Cross-Axis Sensitivity%/gauss	1	Error mbar V/kPa	±2.5	±1	Start Time s	26
Bias°/sec-rms	±50/±50/±80	-	Mean variance°/sec-rms	0.05	0.135	Sensitivity %/°	±0.05	Delay Time ms	0.5	1	Sensitivity dbm	−148

In this paper, three novel state estimation algorithms are proposed to improve the measurement accuracy of low-cost sensors, reduce the cost of flight controller, and realize the application of three-stage series method in hand-launched solar-powered UAV. Additionally, the UAV is special, as it is designed without ailerons and a landing gear to ensure that the full wing is covered with solar cells and is lightweight (Figure 1). Since only the rudder is used for lateral control, the trajectory tracking accuracy of such a UAV is poor, and the state estimation algorithms can combine the characteristics of the UAV to achieve reliable heading estimation, ensuring that it meets the mission requirements and realizes the reduction of platform cost.

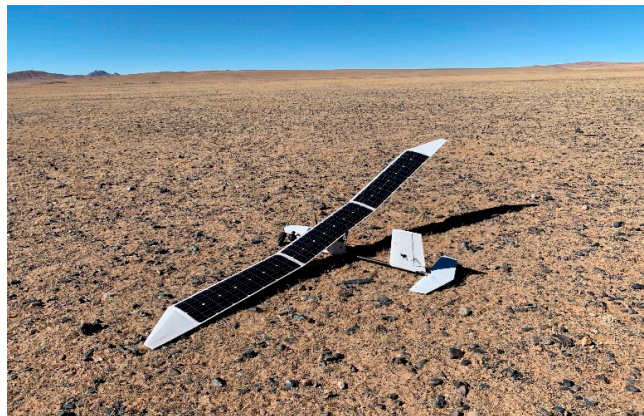


Figure 1. Small hand-launched solar-powered UAV (unmanned aerial vehicle).

The paper is organized as follows. Low-cost sensors measurement error model and error statistics are described in Section 2. Section 3 presents three state estimation algorithms based on EKF. Section 4 verifies the above algorithms according to the whole mission path simulation. The field experiment results and model calibration methods are presented in Section 5, and Section 6 concludes the paper.

2. Sensors Measurement Error Model

The sensors combination in the low-cost flight controller is usually composed of MEMS IMU, magnetometer, pressure sensor and GPS [18], as shown in Figure 2. The SBG-Ellipse IMU module is chosen as an industrial-grade accuracy comparison.

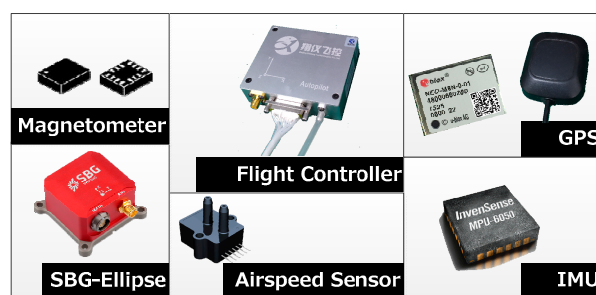


Figure 2. Low-cost flight controller, GPS and (inertial measurement unit) IMU..

The IMU consists of a triaxial gyroscope, a triaxial accelerometer, the hardware is MPU6050. The accelerometer with a capacitive transducer to convert the mass displacement into a voltage output. The measurements are subject to signal bias and random uncertainty, and the simulation model is constructed as follows:

$$\begin{cases} y_{accel,x} = \dot{u} + qw - rv + g\sin\theta + \beta_{accel,x} + \eta_{accel,x} \\ y_{accel,y} = \dot{v} + ru - pw + g\cos\theta\sin\varphi + \beta_{accel,y} + \eta_{accel,y} \\ y_{accel,z} = \dot{w} + pv - qv + g\cos\theta\cos\varphi + \beta_{accel,z} + \eta_{accel,z} \end{cases} \quad (1)$$

where β_{accel} is a bias term, which is dependent on temperature and should be calibrated prior to each flight. The η_{accel} is zero-mean Gaussian noise with the variance σ_{accel}^2 . The output of a gyroscope can be modeled as

$$\gamma_{gyro} = k_{gyro}\Omega + \beta_{gyro} + \eta_{gyro} \quad (2)$$

where γ_{gyro} corresponds to the measured angular rate in volts; k_{gyro} is a gain converting the rate in radians per second to volts and Ω is the angular rate (these two items can be replaced by the real angular rate in the simulation); β_{gyro} is a bias term, mainly dependent on temperature, and the drift is obviously greater than β_{accel} ; η_{gyro} is zero-mean Gaussian noise with covariance σ_{gyro}^2 . The measurement model of the triaxial magnetometer is as follows:

$$y_{mag} = \psi + \beta_{mag} + \eta_{mag} \\ B_0 = R^{-1}(\phi, \theta, \psi) \begin{bmatrix} 0 & 0 & y_{mag} \end{bmatrix}^T \quad (3)$$

where β and η are similar to the above, and R is a direction cosine matrix. The pressure sensor is divided into absolute pressure sensor to measure altitude and differential pressure sensor to measure airspeed. The measurement models are

$$y_{abspres} = \rho g h_{AGL} + \beta_{abspres} + \eta_{abspres} \quad (4)$$

$$y_{diffpres} = \rho V_a^2 / 2 + \beta_{diffpres} + \eta_{diffpres} \quad (5)$$

In the GPS measurement process, not only does the size of the positioning error need to be considered, but also the dynamic characteristics of the error are required. To model the transient behavior of the error, the Gauss–Markov process proved to be effective [19], and the expression is as follows:

$$v[n+1] = e^{-k_{GPS}T_s}v[n] + \eta_{GPS}[n] \quad (6)$$

The GPS measurement information includes UAV space position, ground speed and heading angle can be modeled as

$$y_{GPS,n/e/h}[n] = p_{n/e/h}[n] + v_{n/e/h}[n] \quad (7)$$

$$V_{gGPS} = \sqrt{(V_a \cos \psi + \omega_n)^2 + (V_a \sin \psi + \omega_e)^2} + \eta_V \\ \chi_{GPS} = \tan^{-1}(V_a \sin \psi + \omega_e, V_a \cos \psi + \omega_n) + \eta_\chi \quad (8)$$

where η_V and η_χ are zero-mean Gaussian processes with variance $\sigma_{V_g}^2$ and σ_{χ}^2 , which can be obtained by using basic principles of uncertainty analysis:

$$\sigma_{V_g} = \sqrt{\frac{V_n^2 \sigma_{V_n}^2 + V_e^2 \sigma_{V_e}^2}{V_n^2 + V_e^2}} \quad \sigma_\chi = \sqrt{\frac{V_n^2 \sigma_{V_e}^2 + V_e^2 \sigma_{V_n}^2}{V_n^2 + V_e^2}} \quad (9)$$

The statistical results of various low-cost sensor measurement errors are shown in Table 1, and the SBG-Ellipse is used for comparison.

According to the flight conditions of the UAV at low altitude, with an altitude of 600 m and an airspeed of 12.5 m/s during cruising conditions, combined with what from Table 1, the sensors are simulated as follows: No wind, an altitude of 600 m, an airspeed of 12.5 m/s, the controller temperature is 25 °C and remains unchanged, combined with the orthogonal error and bias of Table 1, the simulation results of various low-cost sensors are as follows. The solid line represents the true state and the dashed line represents the simulated measurement.

In Figure 3, (a,b), which is the IMU simulation, refers to Equations (1) and (2), where the accelerometer measurement is a high frequency signal; (c) is a magnetometer (refer to Equation (3)), and the measurement is a medium–low frequency signal. In order to consider the interference of the solar cell, the model error is appropriately increased, making the measurement slightly unstable; (d,e) is GPS

(refer to Equations (6)–(8)); and (f) is pressure sensors simulation, as low frequency as GPS. The sensor modeling process mainly considers the influence of Gaussian white noise and bias, which is reflected in the range of variation range, frequency and steady-state deviation of the measured value. For the dynamic process, the measurement is consistent with the trend of the true signal. When entering the steady-state, the measurement noise is large and there is a significant deviation.

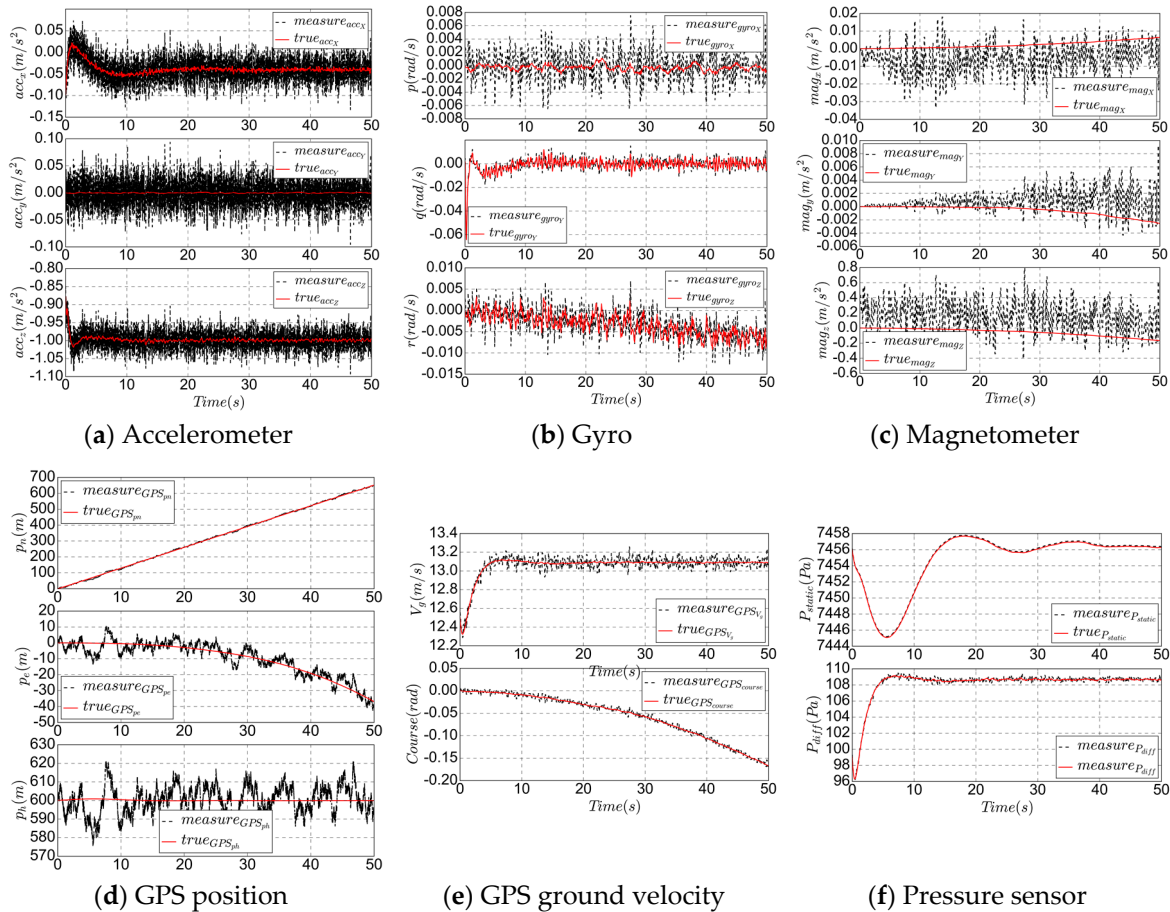


Figure 3. True states and measurement states of low-cost sensors.

3. EKF State Estimation Structure

The extended Kalman filter is similar to the linear Kalman filter, which uses an estimate of statistical variance to calculate the weight of sensor corrections to state estimates [20,21]. The state and observation equations of the nonlinear system are as follows:

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \mathbf{u}) + \mathbf{w}(t) \\ \mathbf{z}_k &= \mathbf{h}(\mathbf{x}(t_k), \mathbf{u}) + \mathbf{v}_k\end{aligned}\quad (10)$$

where $\mathbf{w}(t)$ is used to account for disturbance and unmodeled part, \mathbf{v}_k represents the measurement noise. It is assumed that the $\mathbf{w}(t)$ and \mathbf{v}_k are zero-mean Gaussian noise, and covariance Q and R , respectively. The EKF is first-order linearization near the state, and the system time update equations are

$$\dot{\hat{\mathbf{x}}} = \mathbf{f}(\hat{\mathbf{x}}, \mathbf{u}) \quad (11)$$

$$\mathbf{A}(\mathbf{x}, \mathbf{u}) = \left. \frac{\partial \mathbf{f}(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}} \right|_{\mathbf{x}=\hat{\mathbf{x}}} \quad (12)$$

$$\dot{\mathbf{P}} = \mathbf{A}(\hat{\mathbf{x}}, \mathbf{u})\mathbf{P} + \mathbf{P}\mathbf{A}(\hat{\mathbf{x}}, \mathbf{u})^T + \mathbf{Q} \quad (13)$$

For the update of state estimates when a measurement is received, the measurement update equations are

$$C(x, u) = \frac{\partial h(x^-, u)}{\partial x} \tag{14}$$

$$L = P^- C^T (R + C(\hat{x}, u) P C(\hat{x}, u)^T)^{-1} \tag{15}$$

$$P = (I - LC(\hat{x}, u)) P^- \tag{16}$$

$$\hat{x} = \hat{x}^- + L(z - h(\hat{x}^-, u)) \tag{17}$$

where **A** is the linearized state update matrix, **C** is the linearized model output matrix, **P** is the state covariance matrix, **L** is the Kalman gain matrix, **Q** is the process noise covariance matrix, and **R** is the sensor covariance matrix.

Based on the EKF, this paper divides the state estimation algorithm into full-state estimation and three-state series estimation method according to whether the estimation structure is hierarchical. The full-state estimation consists of the direct and indirect method. The same hand-launched UAV simulation model was used, with an altitude of 600 m and an airspeed of 12.5 m/s, and the hovering condition of the windless conditions.

3.1. Three-Stage Series State Estimation

This algorithm divides the estimation process into three stages. Firstly, the measurement data are low-pass filtered, and gyroscope data are used for prediction and combined with the accelerometer for measurement update, which is the altitude estimation. The second stage is the heading estimation, especially when the UAV is flying in the wind field. The accurate heading angle is crucial for stable control. The final stage is the navigation estimation, based on the previous results and fused with GPS for measurement updates. The algorithm structure is shown in Figure 4.

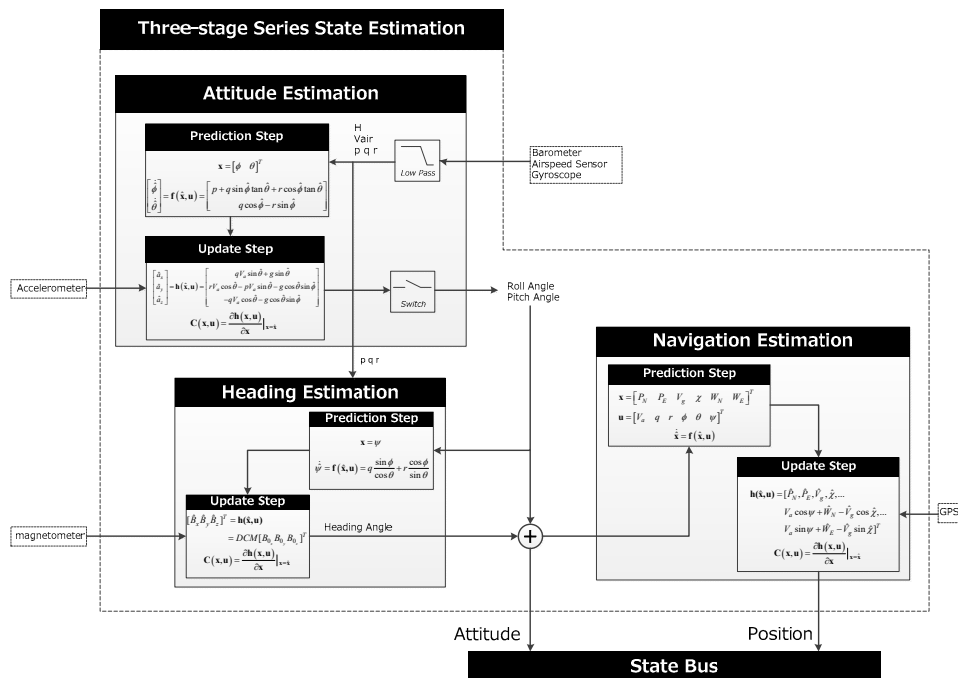


Figure 4. Three-stage series state estimation algorithm structure.

3.1.1. Attitude Estimation

For this stage, the states, inputs and outputs are

$$\mathbf{x} = \begin{bmatrix} \phi & \theta \end{bmatrix}^T \quad \mathbf{u} = \begin{bmatrix} p & q & r & V_a & H \end{bmatrix}^T \quad \mathbf{z} = \begin{bmatrix} a_x & a_y & a_z \end{bmatrix}^T$$

The state variables to be estimated in this stage are pitch and roll. Angular rates, airspeed and altitude make up the input vector, \mathbf{u} , and body frame accelerations composed the system outputs, \mathbf{z} , for this stage. According to the relationship between body frame rotations and changes in roll and pitch angle [22], the time update of the states is

$$\begin{bmatrix} \dot{\hat{\phi}} \\ \dot{\hat{\theta}} \end{bmatrix} = \mathbf{f}(\hat{\mathbf{x}}, \mathbf{u}) = \begin{bmatrix} p + q \sin \hat{\phi} \tan \hat{\theta} + r \cos \hat{\phi} \tan \hat{\theta} \\ q \cos \hat{\phi} - r \sin \hat{\phi} \end{bmatrix} \quad (18)$$

After evaluating the linearization about the current estimate states, the linearized state update matrix \mathbf{A} is obtained:

$$\mathbf{A}(\mathbf{x}, \mathbf{u}) = \left. \frac{\partial \mathbf{f}(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}} \right|_{\mathbf{x}=\hat{\mathbf{x}}} = \begin{bmatrix} q \cos \hat{\phi} \tan \hat{\theta} - r \sin \hat{\phi} \tan \hat{\theta} & \frac{q \sin \hat{\phi} - r \cos \hat{\phi}}{\cos^2 \hat{\theta}} \\ -q \sin \hat{\phi} - r \cos \hat{\phi} & 0 \end{bmatrix} \quad (19)$$

According to Equation (1), assuming that linear acceleration is zero, that is, $\dot{u} = \dot{v} = \dot{w} \approx 0$, the measurement update and linearized model output matrix \mathbf{C} are shown as follows.

$$\begin{bmatrix} \hat{a}_x \\ \hat{a}_y \\ \hat{a}_z \end{bmatrix} = \mathbf{h}(\hat{\mathbf{x}}, \mathbf{u}) = \begin{bmatrix} qV_a \sin \hat{\theta} + g \sin \hat{\theta} \\ rV_a \cos \hat{\theta} - pV_a \sin \hat{\theta} - g \cos \hat{\theta} \sin \hat{\phi} \\ -qV_a \cos \hat{\theta} - g \cos \hat{\theta} \sin \hat{\phi} \end{bmatrix} \quad (20)$$

$$\mathbf{C}(\mathbf{x}, \mathbf{u}) = \left. \frac{\partial \mathbf{h}(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}} \right|_{\mathbf{x}=\hat{\mathbf{x}}} = \begin{bmatrix} 0 & qV_a \cos \hat{\theta} + g \cos \hat{\theta} \\ -g \cos \hat{\phi} \cos \hat{\theta} & -rV_a \sin \hat{\theta} - pV_a \cos \hat{\theta} + g \sin \hat{\phi} \sin \hat{\theta} \\ g \sin \hat{\phi} \cos \hat{\theta} & (qV_a + g \cos \hat{\phi}) \sin \hat{\theta} \end{bmatrix} \quad (21)$$

Using Equations (11) to (17), the state covariance and Kalman gain matrix are updated to achieve the altitude angle estimate. The simulation results of this stage are as follows.

As shown in Figure 5, the subscribe hat and true are the estimated and real state, respectively. Using the variable gain observer (VGO) as a comparison, the three-stage series algorithm has higher accuracy for altitude estimation. The noise of the gyroscope is reflected in the time update, while the accelerometer noise is applied to the measurement update, and the estimated altitude is used as the input of the heading estimation.

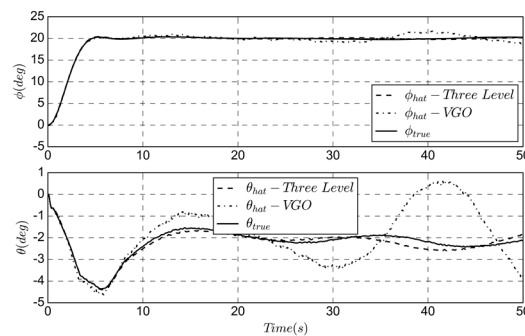


Figure 5. Comparison of true and estimated roll and pitch angle.

3.1.2. Heading Estimation

An accurate heading is important to the lateral control of a UAV without ailerons. The rudder maneuvering produces a yaw moment, then the side slip occurs, and the UAV rolls depending on its own stability. Since the roll control is not direct, it will affect the trajectory tracking accuracy. Magnetometer update heading estimation is useful for cases where a wind induced crab needs to be accounted for when interpreting data from cameras or other sensors [23]. Due to the interference of the solar cell, the magnetometer can be externally mounted and installed at the wing tip to obtain accurate measurement, which is suitable for large aspect ratio solar-powered UAV. The groundspeed will reduce under windy conditions and the heading approximated from GPS will become inaccurate, leading to the vehicle flying uncontrollably downwind [24]. In this stage, the heading estimation is a single-state EKF. The state, inputs and outputs are

$$\mathbf{x} = \psi \quad \mathbf{u} = \begin{bmatrix} \phi & \theta & q & r \end{bmatrix}^T \quad \mathbf{z} = \begin{bmatrix} B_x & B_y & B_z \end{bmatrix}^T$$

The time update of the estimation using gyroscope measurements by the expression

$$\dot{\hat{\psi}} = \mathbf{f}(\hat{\mathbf{x}}, \mathbf{u}) = q \frac{\sin \phi}{\cos \theta} + r \frac{\cos \phi}{\sin \theta} \tag{22}$$

$$\mathbf{A}(\mathbf{x}, \mathbf{u}) = \frac{\partial \mathbf{f}(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}} \Big|_{\mathbf{x}=\hat{\mathbf{x}}} = 0 \tag{23}$$

As the single-state linearization yields a zero matrix, only model noise is considered in the covariance matrix update process. The sensor model is simply a rotational transformation of the earth-frame magnetic field to the body-frame sensor, given by

$$\begin{bmatrix} \hat{B}_x \\ \hat{B}_y \\ \hat{B}_z \end{bmatrix} = \mathbf{h}(\hat{\mathbf{x}}, \mathbf{u}) = \begin{bmatrix} \cos \theta \cos \hat{\psi} & \cos \theta \sin \hat{\psi} & -\sin \theta \\ \sin \phi \sin \theta \cos \hat{\psi} - \cos \phi \sin \hat{\psi} & \sin \phi \sin \theta \sin \hat{\psi} + \cos \phi \cos \hat{\psi} & \sin \phi \cos \theta \\ \cos \phi \sin \theta \cos \hat{\psi} + \sin \phi \sin \hat{\psi} & \cos \phi \sin \theta \sin \hat{\psi} - \sin \phi \cos \hat{\psi} & \cos \phi \cos \theta \end{bmatrix} \begin{bmatrix} B_{0x} \\ B_{0y} \\ B_{0z} \end{bmatrix} \tag{24}$$

Linearizing the current estimates, the linearized model output matrix is

$$\mathbf{C}(\mathbf{x}, \mathbf{u}) = \frac{\partial \mathbf{h}(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}} \Big|_{\mathbf{x}=\hat{\mathbf{x}}} = \begin{bmatrix} -\cos \theta \sin \hat{\psi} B_{0x} + \cos \theta \cos \hat{\psi} B_{0y} \\ (-\sin \phi \sin \theta \sin \hat{\psi} - \cos \phi \cos \hat{\psi}) B_{0x} + (\sin \phi \sin \theta \sin \hat{\psi} + \cos \phi \sin \hat{\psi}) B_{0y} \\ (-\sin \phi \sin \theta \sin \hat{\psi} - \cos \phi \cos \hat{\psi}) B_{0x} + (\sin \phi \sin \theta \sin \hat{\psi} - \cos \phi \sin \hat{\psi}) B_{0y} \end{bmatrix} \tag{25}$$

The heading estimation simulation is performed using the UAV hovering state, as shown in Figure 6. As the VGO does not have a separate heading estimate, the simulation results are compared with the real and estimated yaw angle.

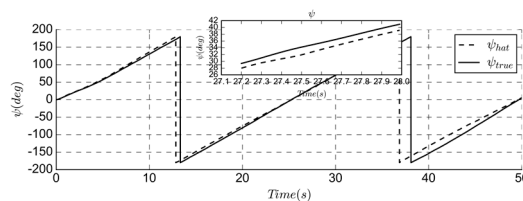


Figure 6. Comparison of true and estimated yaw angle.

Since the solar cell will affect the measurement of the magnetometer, the bias and noise are more obvious, as shown in Figure 3c. Accurate attitude estimation can ensure the accuracy of heading prediction. The measurement update of the magnetometer makes the trend of estimation consistent with the real state. The simulation result has a delay of 1 s and an error of 2 degrees.

3.1.3. Navigation Estimation

The final stage of this state estimation scheme keeps an estimate of vehicle location and path angle. With the altitude and heading obtained in first two stages and measured airspeed, the location, groundspeed and course angle of the airplane are predicted using a simple kinematic model and check against GPS measurements [25]. If the flight path angle $\gamma = 0$, then the evolution of the position is given by

$$\begin{aligned}\dot{P}_N &= V_a \cos \psi \cos \theta - W_N \\ \dot{P}_E &= V_a \sin \psi \cos \theta - W_E\end{aligned}\quad (26)$$

By differentiating Equation (8), the evolution of the ground speed is given by

$$\dot{V}_g = \frac{(V_a \cos \psi + W_N)(-V_a \dot{\psi} \sin \psi) + (V_a \sin \psi + W_E)(V_a \dot{\psi} \cos \psi)}{V_g}\quad (27)$$

According to the kinematic equation of the aircraft during the coordinated turning process, the evolution of χ is given by

$$\dot{\chi} = \frac{g}{V_g} \tan \varphi \cos(\chi - \psi)\quad (28)$$

The state, inputs and output of this stage is

$$\begin{aligned}\mathbf{x} &= [P_N \ P_E \ V_g \ \chi \ W_N \ W_E]^T \\ \mathbf{u} &= [V_a \ q \ r \ \phi \ \theta \ \psi]^T \\ \mathbf{z} &= [y_{GPS,N} \ y_{GPS,E} \ y_{GPS,V_g} \ y_{GPS,\chi} \ y_{W_N} \ y_{W_E}]^T\end{aligned}$$

The nonlinear propagation and linearized state update matrix are given by

$$\begin{bmatrix} \hat{P}_N \\ \hat{P}_E \\ \hat{V}_g \\ \hat{\chi} \\ \hat{W}_N \\ \hat{W}_E \end{bmatrix} = \mathbf{f}(\hat{\mathbf{x}}, \mathbf{u}) = \begin{bmatrix} V_a \cos \psi \cos \theta - \hat{W}_N \\ V_a \sin \psi \cos \theta - \hat{W}_E \\ [(V_a \cos \psi + \hat{W}_N)(-V_a \dot{\psi} \sin \psi) + (V_a \sin \psi + \hat{W}_E)(V_a \dot{\psi} \cos \psi)] / \hat{V}_g \\ g \tan \varphi \cos(\hat{\chi} - \psi) / V_g \\ \mathbf{0}_{2 \times 1} \end{bmatrix}\quad (29)$$

$$\dot{\mathbf{f}} = \begin{bmatrix} \mathbf{0}_{4 \times 2} & \begin{matrix} \cos \hat{\chi} & -V_g \sin \hat{\chi} & -1 & 0 \\ \sin \hat{\chi} & V_g \cos \hat{\chi} & 0 & -1 \\ -\dot{\hat{V}}_g / \hat{V}_g & 0 & -V_a \dot{\psi} \sin \psi & V_a \dot{\psi} \cos \psi \\ -g \tan \varphi \cos(\hat{\chi} - \psi) / \hat{V}_g^2 & -g \tan \varphi \cos(\hat{\chi} - \psi) / \hat{V}_g & 0 & V_a \dot{\psi} \cos \psi \end{matrix} \\ \mathbf{0}_{2 \times 2} & \mathbf{0}_{2 \times 4} \end{bmatrix}\quad (30)$$

GPS measurement information includes position, ground speed and course angle. In order to estimate the wind field, a pseudo measurement of wind speed is constructed as follows:

$$\begin{aligned}y_{W_N} &= V_a \cos \psi + W_N - V_g \cos \chi \\ y_{W_E} &= V_a \sin \psi + W_E - V_g \sin \chi\end{aligned}\quad (31)$$

Therefore, the measurement update and the Jacobian matrix are given by

$$\mathbf{h}(\hat{\mathbf{x}}, \mathbf{u}) = [\hat{P}_N \ \hat{P}_E \ \hat{V}_g \ \hat{\chi} \ V_a \cos \psi + \hat{W}_N - \hat{V}_g \cos \hat{\chi} \ V_a \sin \psi + \hat{W}_E - \hat{V}_g \sin \hat{\chi}]^T\quad (32)$$

$$\mathbf{C}(\mathbf{x}, \mathbf{u}) = \frac{\partial \mathbf{h}(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}} \Big|_{\mathbf{x}=\hat{\mathbf{x}}} = \begin{bmatrix} \mathbf{I}_4 & \mathbf{0}_{4 \times 2} \\ \mathbf{0}_{2 \times 4} & -\cos\chi & \dot{V}_g \sin\chi & \mathbf{I}_2 \\ & -\sin\chi & -V_g \cos\chi & \mathbf{I}_2 \end{bmatrix} \quad (33)$$

The simulation results of the navigation estimation are shown in Figure 7. The estimation results of ground speed and wind field are better than the VGO algorithm; the ground speed static bias is 0.5 m/s, which is improved to less than 0.2 m/s. The estimated GPS course is based on the results of the first two stages, so the position accuracy is higher. Algorithm 1 summary of the three-stage series estimation method.

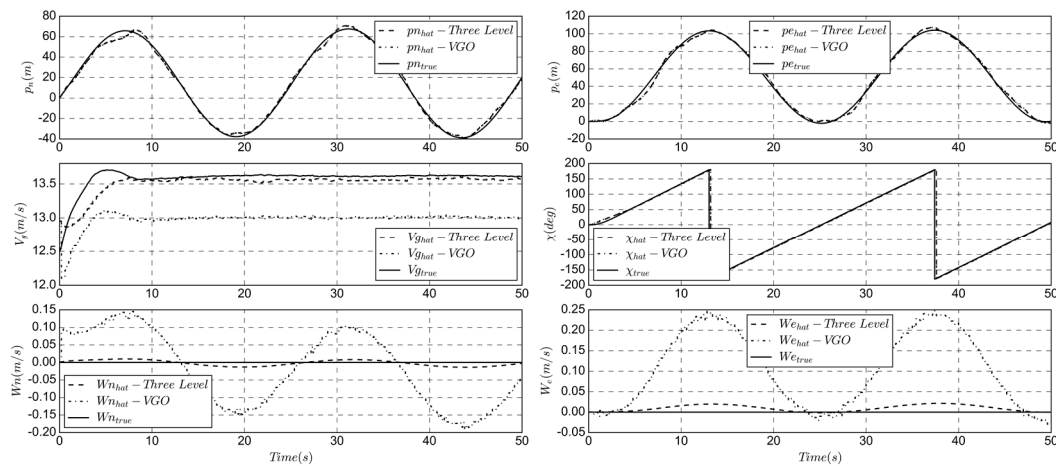


Figure 7. Comparison of true and estimated location, ground speed and course.

Algorithm 1. Three-stage series state estimation

	<ol style="list-style-type: none"> 1: Low-pass filtering of barometers, airspeed sensor and gyroscope measurements, set initial $\hat{\mathbf{x}}$. 2: Update the roll and pitch altitude, Equation (18).
Stage 1: Attitude estimation	<ol style="list-style-type: none"> 3: Compute the linearized state update, covariance matrix and update the prediction, Equations (13) and (19). 4: for each accelerometer axis then 5: Calculate appropriate sensor output, linearized output, and Kalman gain matrix, Equations (15), (20) and (21). 6: Update state estimate and covariance matrix, Equations (16) and (17). 7: end for
Stage 2: Heading estimation	<ol style="list-style-type: none"> 1: Obtain q and r from gyroscope update and ϕ and θ from stage 1. 2: Update the heading angle using kinematic equation, Equation (22). 3: Compute the linearized state update matrix, zero matrix, and covariance matrix, Equations (13) and (23). 4: if magnetometer measurement available then 5: for each magnetometer axis do 6: The same process as the stage 1, sensor measurement update is different, Equations (24) and (25). 7: end for 8: end if
Stage 3: Navigation estimation	<ol style="list-style-type: none"> 1: The airspeed is updated by the airspeed sensor, assign initial value to the altitude angle. 2: Update the state estimate using kinematic equation Equation (29) 3: Compute the linearized state update and covariance matrix, Equations (13) and (30) 4: if GPS measurement available then 5: for each measurement state do 6: The same process as the stage 1, sensor measurement update is different, Equations (32) and (33). 7: end for 8: end if

3.2. Full-State Direct State Estimation

The hierarchical method is used to simplify the model, reduce the order of the matrix, ignore the mutual influence of the state parameters, and obtain a reliable heading estimation. This reduces the amount of calculation necessary but the accuracy is limited. The full-state method treats all parameters

as a whole, the measurement error of IMU is used as a time update, and other sensors are used as measurement update to get the complete state.

When the mission is sensitive to airspeed and altitude, and the flight controller has feasible computing power, full-state direct state estimation is a suitable method to apply. The algorithm structure is shown in Figure 8. Due to the different sampling frequencies, the corresponding state of the fast sensor measurement update using the air pressure sensor includes altitude, airspeed and side slip angle pseudo measurement [26]. The states corresponding to the slow update using GPS include location, ground speed and course.

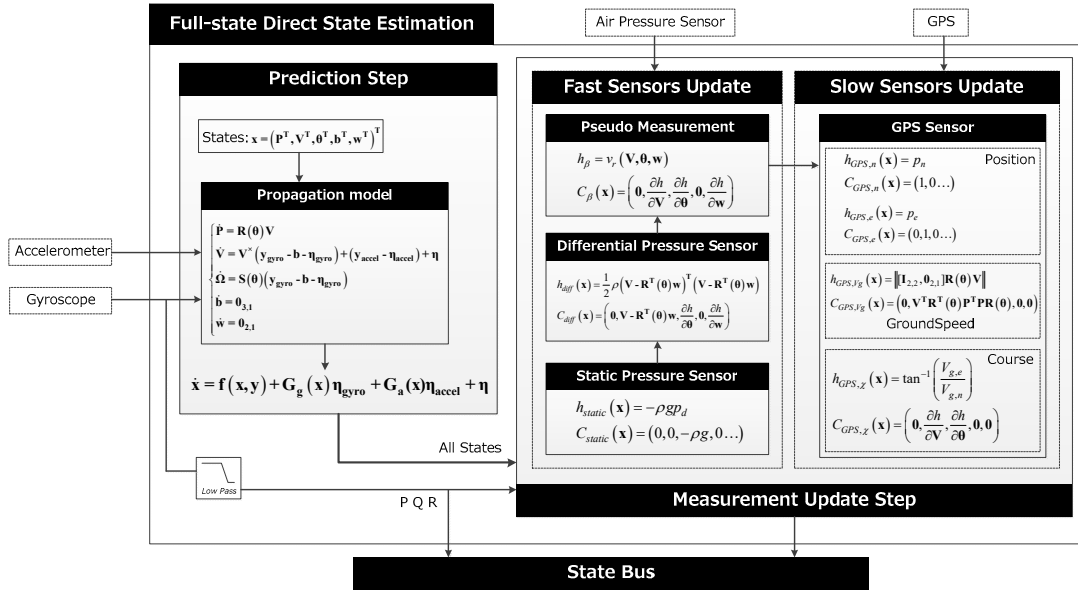


Figure 8. Full-state direct state estimation structure.

The state, measurement input of the time update process are

$$\mathbf{x} = (\mathbf{p}^T, \mathbf{v}^T, \boldsymbol{\theta}^T, \mathbf{b}^T, \mathbf{w}^T)^T \quad \mathbf{y} = (\mathbf{y}_{\text{accel}}^T, \mathbf{y}_{\text{gyro}}^T)^T$$

Using the gyro and accelerometer models, the equations of motion can be written as

$$\left\{ \begin{array}{l} \dot{\mathbf{P}} = \mathbf{R}(\boldsymbol{\theta})\mathbf{V} \\ \dot{\mathbf{V}} = \mathbf{V}^\times (\mathbf{y}_{\text{gyro}} - \mathbf{b} - \boldsymbol{\eta}_{\text{gyro}}) + (\mathbf{y}_{\text{accel}} - \mathbf{b} - \boldsymbol{\eta}_{\text{accel}}) + \boldsymbol{\eta} \\ \dot{\boldsymbol{\Omega}} = \mathbf{S}(\boldsymbol{\Theta})(\mathbf{y}_{\text{gyro}} - \mathbf{b} - \boldsymbol{\eta}_{\text{gyro}}) \\ \mathbf{b} = \mathbf{0}_{3 \times 1} \\ \dot{\mathbf{w}} = \mathbf{0}_{2 \times 1} \end{array} \right. \quad (34)$$

where $\mathbf{P} = (p_n, p_e, p_d)^T$, $\mathbf{V} = (u, v, w)^T$, $\boldsymbol{\theta} = (\phi, \theta, \psi)^T$, $\boldsymbol{\omega} = (p, q, r)^T$, \mathbf{b} is the bias vector, $\boldsymbol{\eta}$ is the noise vector, \mathbf{w} is the wind filed vector, $\mathbf{R}(\boldsymbol{\theta})$ is the direction cosine matrix, and $\mathbf{S}(\boldsymbol{\theta})$ is the transformation matrix of the angular velocity of body axes to inertial axes. Therefore, the state propagation equation is

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{y}) + \mathbf{G}_g(\mathbf{x})\boldsymbol{\eta}_{\text{gyro}} + \mathbf{G}_a(\mathbf{x})\boldsymbol{\eta}_{\text{accel}} + \boldsymbol{\eta} \quad (35)$$

The time update model of the state and the Jacobian matrices are as follows:

$$f(x,y) = \begin{pmatrix} \mathbf{R}(\theta)\mathbf{V} \\ \mathbf{V}^\times(\mathbf{y}_{gyro}-\mathbf{b})+(\mathbf{y}_{accel}-\mathbf{b}) + \mathbf{R}^T(\theta)\mathbf{g} \\ \mathbf{S}(\theta)(\mathbf{y}_{gyro}-\mathbf{b}) \\ \mathbf{0}_{3 \times 1} \\ \mathbf{0}_{2 \times 1} \end{pmatrix} \tag{36}$$

$$\mathbf{A}(x,y) = \frac{\partial f}{\partial x} = \begin{pmatrix} \mathbf{0}_{3 \times 3} & \mathbf{R}(\theta) & \frac{\partial[\mathbf{R}(\theta)\mathbf{V}]}{\partial \theta} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 2} \\ \mathbf{0}_{3 \times 3} & -(\mathbf{y}_{gyro}-\mathbf{b})^\times & \frac{\partial[\mathbf{R}^T(\theta)\mathbf{g}]}{\partial \theta} & -\mathbf{V}^\times & \mathbf{0}_{3 \times 2} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \frac{\partial[\mathbf{S}(\theta)(\mathbf{y}_{gyro}-\mathbf{b})]}{\partial \theta} & -\mathbf{S}(\theta) & \mathbf{0}_{3 \times 2} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 2} \\ \mathbf{0}_{2 \times 3} & \mathbf{0}_{2 \times 3} & \mathbf{0}_{2 \times 3} & \mathbf{0}_{2 \times 3} & \mathbf{0}_{2 \times 2} \end{pmatrix} \tag{37}$$

The fast sensor measurement update includes three parts, static and differential pressure sensor and pseudo side-slip angle. According to Equations (4) and (5), the measurement model and associated Jacobian matrices are given by

$$h_{static}(x) = -\rho g p_d \quad h_{diff}(x) = \frac{1}{2} \rho V_a^2 = \frac{1}{2} \rho (\mathbf{V}-\mathbf{R}^T(\theta)\mathbf{w})^T (\mathbf{V}-\mathbf{R}^T(\theta)\mathbf{w}) \tag{38}$$

$$C_{static}(x) = (0, 0, -\rho g, 0 \dots) \quad C_{diff}(x) = \left(\mathbf{0}_{3,1}, \mathbf{V}-\mathbf{R}^T(\theta)\mathbf{w}, \frac{\partial h}{\partial \theta}, \mathbf{0}_{3,1}, \frac{\partial h}{\partial \mathbf{w}} \right) \tag{39}$$

With the given sensors, the side-slip angle or side-to-side velocity is unobservable and drift occurs, but it is important when the control system needs to obtain reliable lateral information to control the flight path [12]. To correct this parameter, a pseudo measurement is proposed on the side-slip angle by assuming that it is zero. Therefore, the model for the pseudo-sensor is given by

$$h_\beta = v_r(\mathbf{V},\theta,\mathbf{w}) = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} (\mathbf{V}-\mathbf{R}^T(\theta)\mathbf{w}) \tag{40}$$

The associate Jacobian matrix is given by

$$C_\beta(x) = \left(\mathbf{0}_{1 \times 3}, \frac{\partial h}{\partial \mathbf{V}}, \frac{\partial h}{\partial \theta}, \mathbf{0}_{1 \times 3}, \frac{\partial h}{\partial \mathbf{w}} \right) \tag{41}$$

The GPS measurement is updated at a low frequency, including north and east position, ground speed and heading. Similarly, the measurement model and the corresponding Jacobian matrices are as follows:

$$h_{GPS,n}(x) = p_n \quad h_{GPS,e}(x) = p_e \tag{42}$$

$$h_{GPS,V_g}(x) = \|[I_{2,2}, 0_{2,1}]\mathbf{R}(\theta)\mathbf{V}\| \quad h_{GPS,\chi}(x) = \tan^{-1}(V_{g,e}/V_{g,n})$$

$$C_{GPS,n}(x) = (1, 0, 0 \dots) \quad C_{GPS,e}(x) = (0, 1, 0 \dots) \tag{43}$$

$$C_{GPS,V_g}(x) = (\mathbf{0}, \mathbf{V}^T \mathbf{R}^T(\theta) \mathbf{P}^T \mathbf{P} \mathbf{R}(\theta), \mathbf{0}, \mathbf{0}) \quad C_{GPS,\chi}(x) = (\mathbf{0}, \frac{\partial h}{\partial \mathbf{V}}, \frac{\partial h}{\partial \theta}, \mathbf{0}, \mathbf{0})$$

The comparison between the real and the estimate state of all parameters is shown in Figure 9. Algorithm 2 is a summary of full-state direct state estimation method.

Compared with Figures 5 and 6, the accuracy of full-state method is obviously superior in position, speed, and heading. When entering the steady state, the altitude deviation is small, the error precision is less than 8%, the GPS-related state measurement accuracy is higher, and the position error precision is approximately 3 m, ensuring sufficient trajectory tracking accuracy. Due to the increase of accuracy of the estimated ground speed and course, this method is suitable for reliable trajectory tasks or fixed-area hovering, such as animal photography or precise takeoff and landing.

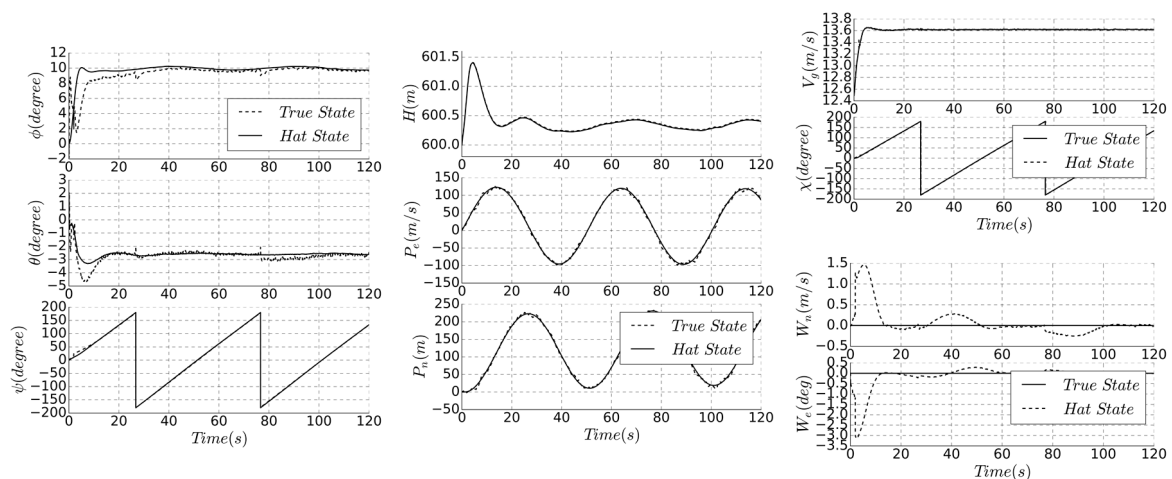


Figure 9. Comparison of true and estimated state.

Algorithm 2. Full-state direct state estimation

	1: Obtain ω and \mathbf{a} from gyroscope and accelerometer, set initial $\hat{\mathbf{x}}$.
Time update	2: Update the state estimate using the kinematic equation, Equation (35). 3: Compute the linearized state update matrix and covariance variable, Equations (13) and (37).
	1: if air pressure sensor measurement available, then
	2: for static and differential pressure sensor measurement do
	3: Calculate sensor output, linearized output equations, and Kalman gain, Equations (15), (38) and (39).
Fast sensor update	5: Update state estimate and covariance, Equations (16) and (17)
	6: end for
	7: Calculate pseudo side slip angle output and linearized output equations, Equations (40) and (41).
	8: Using zero to update state estimate, Equations (15)–(17)
	9: end if
	1: if new GPS measurement available, then
	2: for each measurement state do
Slow sensor update	3: Calculate location, groundspeed and course output and linearized output equations, Equations (42) and (43)
	4: Using measurement to update state estimate, Equations (15)–(17)
	5: end for
	6: end if

3.3. Full-State Indirect State Estimation

The direct state estimate method focuses on the observation of all measurement states, while for the indirect estimation method the idea is to filter the error states, which should satisfy the linear and Gaussian assumptions better than the direct implementation. The biggest difference of indirect method is that the error estimation vector is introduced, which is predicted by time update, and the estimation of states is achieved by observation and compensation of errors. Let \mathbf{x} be the true state, $\hat{\mathbf{x}}$ the estimated state, then $\tilde{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}}$ is the error state. Similar to the previous section, if $\hat{\mathbf{x}}$ satisfies

$$\dot{\hat{\mathbf{x}}} = \mathbf{f}(\hat{\mathbf{x}}, \mathbf{y}) \quad (44)$$

The state error evolves according to Equation (35), $\tilde{\mathbf{x}}$ can be written as

$$\dot{\tilde{\mathbf{x}}} = \mathbf{f}(\hat{\mathbf{x}}, \mathbf{y}) + \mathbf{G}(\mathbf{x})\boldsymbol{\eta}_s + \boldsymbol{\eta}_i - \mathbf{f}(\hat{\mathbf{x}}, \mathbf{y}) \quad (45)$$

Using the Tylor series expansion up to the linear term, \mathbf{f} and \mathbf{G} are equivalent to

$$\begin{aligned} \mathbf{f}(\mathbf{x}, \mathbf{y}) &\approx \mathbf{f}(\hat{\mathbf{x}}, \mathbf{y}) + \mathbf{A}(\hat{\mathbf{x}}, \mathbf{y})\tilde{\mathbf{x}} \\ \mathbf{G}(\mathbf{x})\boldsymbol{\eta}_s &\approx \mathbf{G}(\hat{\mathbf{x}})\boldsymbol{\eta}_s \end{aligned} \quad (46)$$

Equation (45) can be simplified as

$$\dot{\tilde{x}} = \mathbf{A}(\hat{x}, y) + \mathbf{G}(\hat{x})\eta_s + \eta_i \tag{47}$$

The basic idea is to run the indirect filter on the state error, then to correct the state estimate at each step by adding in the error, and the state error is reset to zero after each correction. In the time update process, the state error propagation equation can be written as

$$\dot{\tilde{x}} = \mathbf{A}(\hat{x}, y)\tilde{x} \tag{48}$$

In the measurement update process, it is also necessary to introduce a measurement of the state error, and Equation (17) can be rewritten as

$$\begin{aligned} \tilde{x}^+ &= \tilde{x}^- + \mathbf{L}(\mathbf{z} - \mathbf{h}(\hat{x}^-, \mathbf{u}) - \mathbf{C}(\hat{x}^-, \mathbf{u})\tilde{x}^-) \\ \hat{x}^- &= \hat{x}^+ + \tilde{x}^+ \end{aligned} \tag{49}$$

The state error is set to zero after each estimation, that is $\tilde{x} = 0$. The algorithm is outlined in Figure 10.

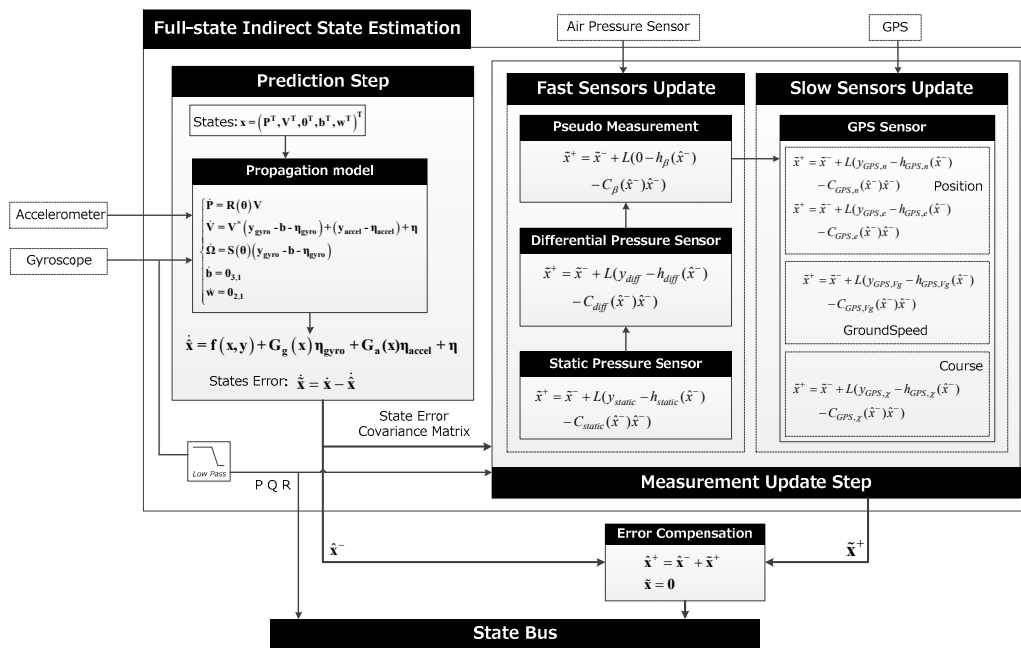


Figure 10. Full-state indirect state estimation structure.

As the supplement to the direct filter, it is more computationally intensive and more complex, but the elimination of measurement bias is better than the first two methods. Due to the large size of the state matrix, the computation power of low-cost flight controller is a challenge. Figure 11 shows the simulation results of the indirect method, Table 2 compares the accuracy of the different algorithms, and Algorithm 3 is a summary of the method.

According to the above, it can be seen that the estimation accuracy of indirect method is better than that of direct method, and the estimation accuracy of wind field is higher. The indirect method has further improved the estimation accuracy of the altitude and position. Compared with Figure 9, the altitude deviation is small at the beginning, the deviation after steady state is reduced, and the result is more stable. Therefore, this method is suitable for large-scale solar-powered UAV or mission with high trajectory tracking accuracy.

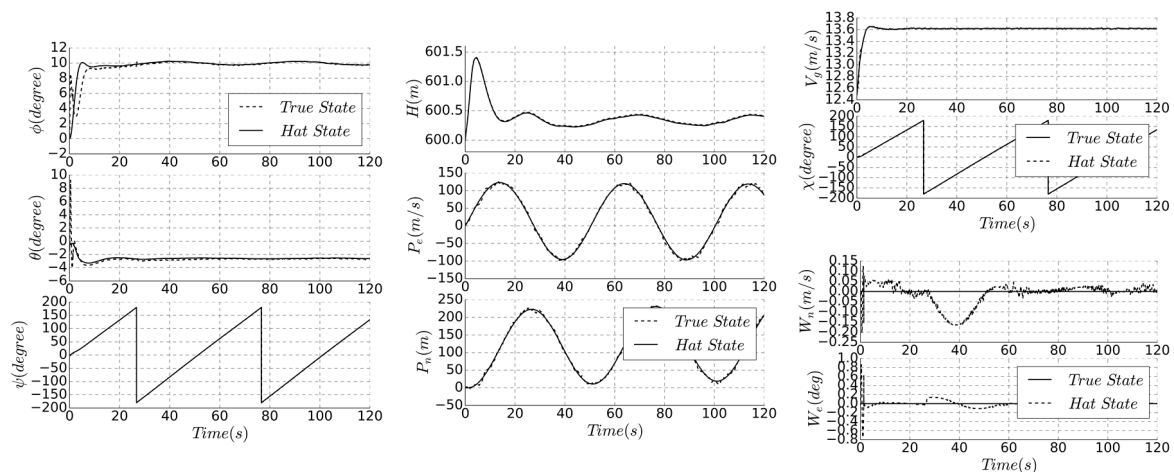


Figure 11. Comparison of true and estimated state.

Table 2. Comparison of the estimate accuracy of three algorithms.

	Roll Angle (deg)	Pitch Angle (deg)	Yaw Angle (deg)	Position (m)	Height (m)	Airspeed (m/s)	Ground Speed (m/s)	GPS Course (deg)	Wind (m/s)
Three-stage series	2.52	3.13	7.19	6.04	5.3	2.4	2.9	10.45	0.3
Full-state direct	1.24	0.4	1.58	2.93	0.98	0.8	0.68	1.62	0.3
Full-state indirect	0.48	0.3	0.61	2.9	0.98	0.5	0.37	1.48	0.15

Algorithm 3. Full-state indirect state estimation

Time update	<ol style="list-style-type: none"> 1: Obtain ω and \mathbf{a} from gyroscope and accelerometer, set initial $\hat{\mathbf{x}}$ and $\bar{\mathbf{x}} = \mathbf{0}$. 2: Update the state estimate using the kinematic equation and linearized state, Equations (35) and (37). 3: Propagate $\bar{\mathbf{x}}$ and \mathbf{P} according to Equation (Equations (13) and (48))
Fast sensor update	<ol style="list-style-type: none"> 1: if air pressure sensor measurement available then 2: for static and differential pressure sensor measurement do 3: Calculate sensor output, linearized output equations, and Kalman gain, Equations (15), (38) and (39). 4: Update state covariance, state error and estimate state, Equations (16) and (49). 5: end for 6: Calculate pseudo side slip angle output and linearized output equations, Equations (40) and (41). 7: Using zero to update state estimate, Equations (15), (16) and (49) 8: end if
Slow sensor update	<ol style="list-style-type: none"> 1: if new GPS measurement available then 2: for each measurement state do 3: Calculate location, groundspeed and course output and linearized output equations (Equations (42) and (43)) 4: Using measurement to update state error (Equations (15), (16) and (49)) 5: end for 6: end if 7: State error reset zero, $\bar{\mathbf{x}} = \mathbf{0}$.

4. Simulation

In order to verify the effectiveness of the proposed algorithm, the simulation of the complete mission path is carried out for the same UAV simulation model. A typical climb-cruise-hover-descend is used as the mission path, and the hovering process performs specific tasks, and the estimation accuracy in each flight phase and state switching is obtained. The flight trajectory and estimated parameters are shown in Figures 12 and 13.

All three algorithms can achieve autonomous flight in the mission process, and the altitude and trajectory accuracy are different. As the static pressure sensor and airspeed sensor are only processed by low-pass filtering, the static bias of three-stage series method will appear in the estimation of altitude and airspeed, which is 5.3 m and 0.9 m/s, respectively, and the error of flight trajectory will also appear in the heading estimation, which is approximately 10 degrees, resulting in a position error of 6 m in the hover phase.

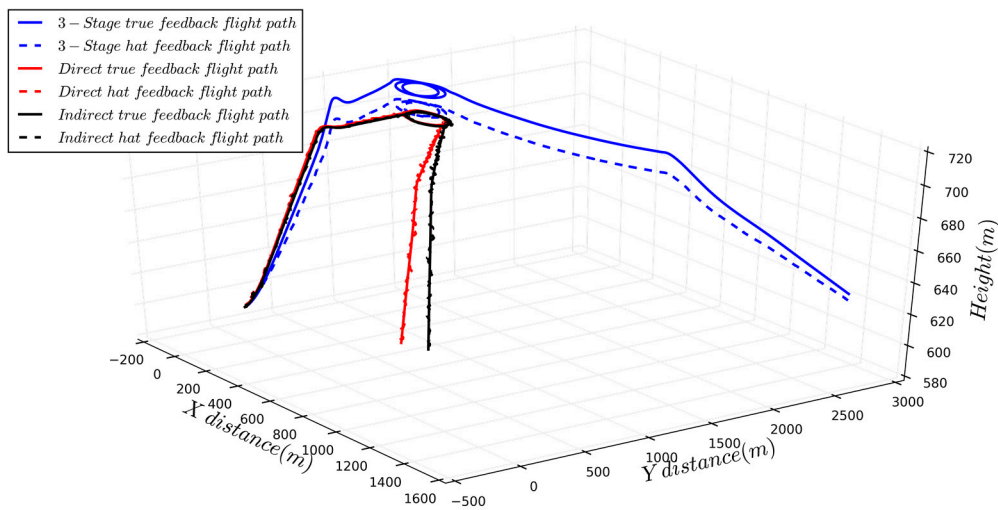


Figure 12. Flight trajectory comparison of different state estimation algorithms.

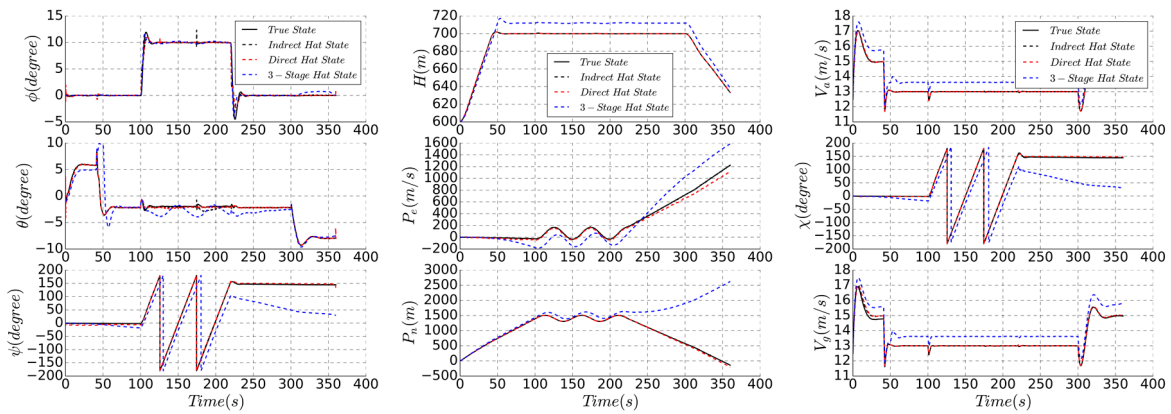


Figure 13. Different mission phases simulation.

Full-state estimation method fuses the measured data of pressure sensor and GPS, and has higher accuracy of position and velocity estimation, and there is only a small error in altitude, airspeed, and groundspeed. Full-state methods can consider the mutual influence between estimated states, and the parameter is not abruptly changed during the state switching, which is obvious in the three-stage series method. The altitude and position accuracy of the direct method are 1.2 degrees and 3.4 m, respectively. The indirect method achieves similar precision, but the estimated result is more stable. Since this method is based on the elimination of state error, the altitude estimation accuracy is high and the steady-state process estimation accuracy fluctuation is small.

Table 3 compares the three algorithms from five aspects: Algorithm complexity, hardware requirements, number of estimable parameters, estimation accuracy and application. The indirect full-state algorithm has the highest accuracy, but the hardware requirements are also highest. The direct full-state algorithm is more compromised, and the three-stage series has the adequate heading accuracy, with the lowest controller computation requirements and simplest complexity, but the estimated parameters are limited.

Table 3. Comparison of different state estimation algorithms.

	Algorithm Complexity	Hardware Requirement	Estimated Parameters	Estimation Accuracy	Application
Three-stage series	★★	★	14	★★	Low-cost, small UAV, lower mission requirements.
Full-state direct	★★☆	★☆	19	★★★	Low-cost UAV with high trajectory tracking accuracy.
Full-state indirect	★★★	★★	19	★★★★	Mission trajectory requires high precision and large-scale UAV.

5. Field Experiment

A full-scale electronic UAV, with a wingspan of 3 m and a weight of 3 kg without solar cells, was equipped with a low-cost flight controller and the three-stage series estimation algorithm were loaded for the full mission flight verification (Figure 14). The cost of the flight controller is less than \$3000, and the whole system is less than \$8000. The experimental site is located at (108°56′3.07″ E, 34°58′58.85″ N), with an altitude of 630 m, an average wind speed of 3.5 m/s, a ground temperature of 10 °C, and during autumn.



Figure 14. Field flight test of a small hand-launched UAV.

The UAV has no landing gear and it takes off through hand-launch from the car. The fuselage touches the ground when landing. The predicted state during the calculation is the altitude, position, and groundspeed. In the Figure 15a,c are three-dimensional and two-dimensional flight trajectories, respectively, and (b) is parameters and commands recorded by flight controller. The directly measured states include altitude, airspeed, longitude and latitude; the IMU data are shown in the dotted line in Figure 16. Through the control of pitch and roll angle, altitude and airspeed control results, the three-stage series state estimation algorithm can provide a reliable state for flight controller. The calculation frequency is 50 Hz, and the IMU is also sampled based on the same frequency; the GPS measurement frequency is 10 Hz, which is the same as the data transmission frequency; and the data link frequency band is 915 MHz. The variation range is 8 m, the airspeed is 9–15 m/s, the groundspeed is 6–16 m/s, the roll angle accuracy is 6.5 degrees, the pitch angle is 3 degrees, the yaw angle is 9.3 degrees, and the trajectory tracking precision is nearly 23 m.

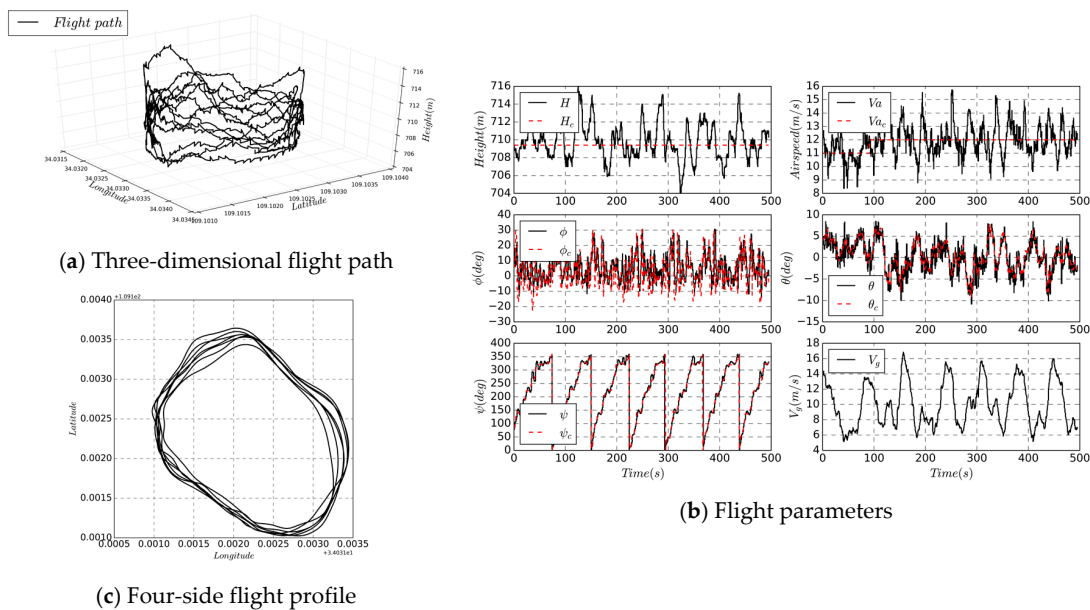


Figure 15. Flight path and state parameters during cruise process.

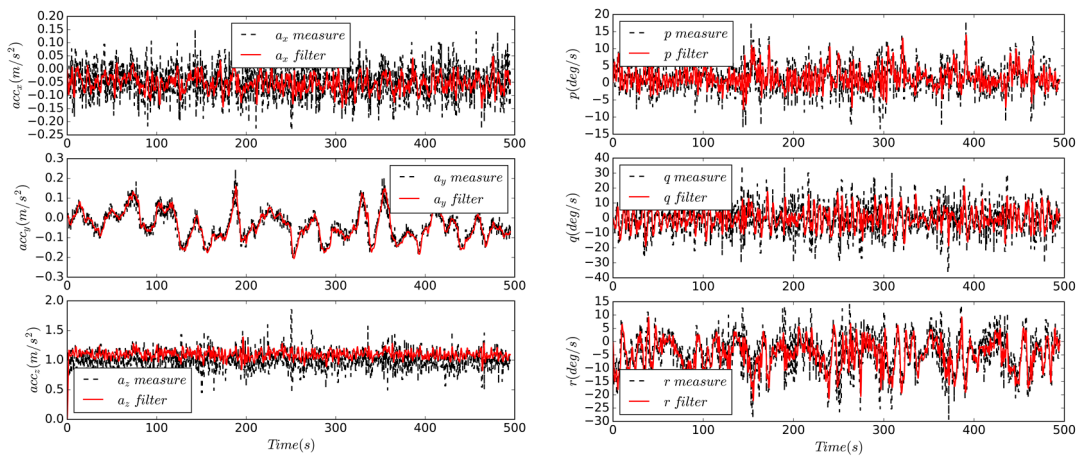


Figure 16. Comparison of IMU measurement and filtered data.

The field flight test can not only verify the feasibility of the state estimation algorithm, but also calibrate some parameters in the modeling process, such as sensor measurement bias, noise and covariance matrix \mathbf{R} . The Finite Impulse Response (FIR) filter is used to filter the flight data [27], as shown in Figure 16. The measurement result of the IMU is regarded as the raw data, and the filtered parameter is regarded as the real state. The bias and noise of the sensor model can be obtained by the following equations:

$$\begin{aligned} \eta_{correct} &= \frac{X_{m_{\min}}^{\max} - X_{f_{\min}}^{\max}}{2} \\ \beta_{correct} &= E(X_m) - E(X_f) \end{aligned} \quad (50)$$

where $\eta_{correct}$ is the corrected measurement noise and $\beta_{correct}$ is the corrected measurement bias. The subscripts m and f represent measured and filtered data.

The statistical data are of the measured noise and deviation of the IMU is shown in Table 4. For the noise of the accelerometer and the bias of the gyroscope, the value in the y direction is smaller than the x and z directions. The calibration of the covariance matrix \mathbf{R} in the state estimation process is shown as follows:

$$\mathbf{R}_{accel} = \begin{bmatrix} (0.0818)^2 & 0 & 0 \\ 0 & (0.0193)^2 & 0 \\ 0 & 0 & (0.251)^2 \end{bmatrix} \quad \mathbf{R}_{gyro} = \begin{bmatrix} (0.0723)^2 & 0 & 0 \\ 0 & (0.0906)^2 & 0 \\ 0 & 0 & (0.0898)^2 \end{bmatrix} \quad (51)$$

Table 4. The statistics of IMU measurement noise and bias in the cruising process.

	Accelerometer (m/s ²)			Gyroscope (deg/s)		
	x Direction	y Direction	z Direction	p	q	r
Noise	0.0818	0.0193	0.2514	4.1424	5.915	5.146
Bias	0.00424	0.00315	−0.0875	−0.1074	0.0383	0.3928

For the velocity vector, $\mathbf{V} = (u, v, w)^T$, both u and w are greater than v during the mission process. A penalty factor τ can be introduced in the matrix \mathbf{R}_{accel} to get a more general expression, rewritten as

$$\mathbf{R}_{accel} = \begin{bmatrix} R_{x_0} & \tau R_{y_0} & R_{z_0} \end{bmatrix}^T \quad (52)$$

6. Conclusions

This paper begins with the cost of the flight controller of a hand-launched solar-powered UAV without ailerons and landing gear, and proposes three novel state-estimation algorithms according to

whether they are hierarchical, direct or indirect. The UAV trajectory tracking accuracy is improved by three algorithms and the platform cost is reduced.

Firstly, the measurement process of the low-cost sensors is modeled, and the simulation of the model is completed based on the statistical values of the measurement error and bias. The measurement of the accelerometer is high frequency, the gyroscope and the magnetometer are in middle frequency, and the GPS measurement is low frequency.

The three-stage series method reduces the longitudinal accuracy, and focuses on the estimation of heading angle to improve the trajectory precision, which is simple, with a clear hierarchy, limited estimation parameters, and is suitable for application on low-cost small UAV platforms. The full-state method can also obtain side-slip angle or side-to-side velocity estimation to improve the control precision of the aileron free UAV. For the solar-powered UAV, a method of fixing the magnetometer externally can be selected to ensure heading accuracy. The full-state direct method has a higher estimation accuracy and more estimation parameters, but the increase of the algorithm calculation requires a higher-level controller, suitable for high control precision and harsh takeoff and landing conditions. The indirect method has the highest accuracy estimation and the highest requirements for flight controllers, which can be applied to large solar-powered UAV platforms.

Simulations are presented to validate the proposed methods, and the results show the three-stage series method has a position accuracy of approximately 6 m, while the direct and indirect methods are both 3.4 m; however, the stability and state switching of the indirect method are better. Afterwards, the three-stage series algorithm is loaded on the full-scale low-cost electric UAV. The results illustrate that the three-stage series algorithm can provide a reliable state for the controller and achieve stable control, the yaw angle precision is 9.3 degrees, and the trajectory tracking precision is nearly 23 m. Finally, the sensor model and covariance matrix \mathbf{R} corrected according to the recorded and filtered data of the IMU during flight.

As a next step, we will conduct research on a long-endurance, high-reliable flight control system, and adaptability to harsh environmental condition. With the long time reliability test of the hardware, the service life of the low-cost solar-powered UAV can be obtained. The completion of this series of experiments makes it closer to civilian applications.

Author Contributions: A.G. and Z.Z. conducted the sensor and whole system modeling; X.Z. contributed the different state estimation algorithms; A.G. and F.B. performed the experiments; Z.Z. and X.Z. analyzed the experiment results; A.G. and Z.Z. wrote the paper.

Funding: This research was funded by the [Civil Aircraft Specific Project of China] grant number [MJ-2015-F-009] and [Key R&D Projects in Shaanxi Province of China] grant number [2018ZDCXL-GY-03-04].

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Colella, N.; Wenneker, G. Pathfinder: Developing a solar rechargeable aircraft. *IEEE Potentials* **1996**, *15*, 18–23. [[CrossRef](#)]
2. Oettershagen, P.; Melzer, A.; Mantel, T. Design of small hand-launched solar-powered UAVs: From concept study to a multi-day world endurance record flight. *J. Field Robot.* **2017**, *34*, 1352–1377. [[CrossRef](#)]
3. Cocconi, A. *AC Propulsion's Solar Electric Powered SoLong UAV*; Technical Report; AC Propulsion: San Dimas, CA, USA, 2005. Available online: <https://archive.org/detail/ACPropulsionSolongUAV2005> (accessed on 5 June 2005).
4. Noth, A. Design of Solar Powered Airplanes for Continuous Flight. Ph.D. Thesis, Swiss Federal Institute of Technology (ETH), Zürich, Switzerland, September 2008.
5. Oettershagen, P.; Stastny, T.; Hinzmann, T. Robotic technologies for solar-powered UAVs: Fully autonomous updraft-aware aerial sensing for multiday search-and-rescue missions. *J. Field Robot.* **2017**, *35*, 1–29.
6. Huang, Y.; Wang, H.L.; Yao, P. Energy-optimal path planning for Solar-powered UAV with tracking moving ground target. *Aerosp. Sci. Technol.* **2016**, *53*, 241–251. [[CrossRef](#)]

7. Ebeid, E.; Skriver, M.; Terkildsen, K.H.; Jensen, K.; Schultz, U.P. A Survey of Open-Source UAV Flight Controllers and Flight Simulators. *Microprocess. Microsyst.* **2018**, *61*, 11–20. [[CrossRef](#)]
8. Beard, R.W. *Embedded UAS Autopilot and Sensor Systems. Encyclopedia of Aerospace Engineering*; John Wiley & Sons, Ltd.: Chichester, UK, 2010; pp. 4799–4814.
9. Peng, X.D.; Zhang, T.M.; Li, J.Y.; Chen, Y. Attitude estimation algorithm of agricultural small-UAV based on sensors fusion and calibration. *Acta Autom. Sin.* **2015**, *41*, 854–860.
10. Arifianto, O.; Farhood, M. Development and Modeling of a Low-Cost Unmanned Aerial Vehicle Research Platform. *J. Intell. Robot. Syst.* **2015**, *80*, 139–164. [[CrossRef](#)]
11. Hwang, S.H.; Kim, D.S. Hybrid Helmet Attitude Tracking System using Adaptive Complementary Filter. *Measurement* **2019**, *146*, 186–194. [[CrossRef](#)]
12. Beard, R.W.; Mclain, T.W. *Small Unmanned Aircraft: Theory and Practice*; Princeton University Press: Princeton, NJ, USA, 2012.
13. Paulsen, T.H. Low Cost/High Precision Flight Dynamics Using the Square-Root Unscented Kalman Filter. Masters' Thesis, Department of Electrical and Computer Engineering, Brigham Young University, Provo, UT, USA, 2009; pp. 12–19.
14. Heikki, H.; Arto, V. A DCM Based Attitude Estimation Algorithm for Low-Cost MEMS IMUs. *Int. J. Navig. Obs.* **2015**, *2015*, 503814.
15. Strohmeier, M.; Montenegro, S. Coupled GPS/MEMS IMU attitude determination of small UAVs with COTS. *Electronics* **2017**, *6*, 15. [[CrossRef](#)]
16. Ko, N.; Youn, W.; Choi, I.; Song, G.; Kim, T. Features of Invariant Extended Kalman Filter Applied to Unmanned Aerial Vehicle Navigation. *Sensors* **2018**, *18*, 2855. [[CrossRef](#)] [[PubMed](#)]
17. Hu, L.; Yang, W.; He, J.; Zhou, H.; Zhang, Z.; Luo, X.; Zhao, R.; Tang, L.; Du, P. Roll angle estimation using low cost MEMS sensors for paddy field machine. *Comput. Electron. Agric.* **2019**, *158*, 183–188. [[CrossRef](#)]
18. Pei, X.B. Low Cost Integrated Navigation and Stability Control of New Multi Rotor UAV. Ph.D. Thesis, Changchu Insitute of Optics, Fine Mechanics and Physicas, Chinese Academy of Science, Changchun, China, 2018; pp. 25–40.
19. Rankin, J. GPS and differential GPS: An error model for sensor simulation. In Proceedings of the IEEE 1994 Position Location and Navigation Symposium, Las Vegas, NV, USA, 11–15 April 1994; pp. 260–266.
20. Eldredge, A.M. Improved State Estimation for Miniature Air Vehicles. Master's Thesis, Brigham Young University, Provo, UT, USA, 20 August 2006.
21. Di, L.; Fromm, T.; Chen, Y.Q. A data fusion system for attitude estimation of low-cost miniature UAVs. *J. Intell. Robot. Syst.* **2012**, *65*, 621–635. [[CrossRef](#)]
22. Han, S.; Wang, J. A Novel Method to Integrate IMU and Magnetometers in Attitude and Heading Reference Systems. *J. Navig.* **2011**, *64*, 727–738. [[CrossRef](#)]
23. Hong, H.; Wang, M.; Florian, H.; Tang, S. Fast real-time three-dimensional wind estimation for fixed-wing aircraft. *Aerosp. Sci. Technol.* **2017**, *69*, 674–685. [[CrossRef](#)]
24. Goh, S.T.; Abdelkhalik, O.; Zekavat, S.A. A Weighted Measurement Fusion Kalman Filter implementation for UAV navigation. *Aerosp. Sci. Technol.* **2013**, *28*, 315–323. [[CrossRef](#)]
25. Rhudy, M.B.; Fravolini, M.L.; Porcaccia, M. Comparison of wind speed models within a Pitot-free airspeed estimation algorithm using light aviation data. *Aerosp. Sci. Technol.* **2019**, *86*, 21–29. [[CrossRef](#)]
26. Ma, Z.Y.; Zhu, X.P.; Zhou, Z. On-ground lateral direction control for an unswept fly-wing UAV. *Aeronaut. J.* **2019**, *123*, 416–432. [[CrossRef](#)]
27. Wang, B.K.; Liu, D.T.; Wang, W.J. A hybrid approach for UAV flight data estimation and prediction based on flight mode recognition. *Microelectron. Reliab.* **2018**, *84*, 253–262. [[CrossRef](#)]

