actuators

Article

Flying State Sensing and Estimation Method of Large-Scale Bionic Flapping Wing Flying Robot

Guangze Liu, Song Wang and Wenfu Xu *

School of Mechanical Engineering and Automation, Harbin Institute of Technology, Shenzhen 518055, China; 19b953008@stu.hit.edu.cn (G.L.); s.wang@cqu.edu.cn (S.W.)
* Correspondence: wfxu@hit.edu.cn

Abstract: A large bionic flapping wing robot has unique advantages in flight efficiency. However, the fluctuation of fuselage centroid during flight makes it difficult for traditional state sensing and estimation methods to provide stable and accurate data. In order to provide stable and accurate positioning and attitude information for a flapping wing robot, this paper proposes a flight state sensing and estimation method integrating multiple sensors. Combined with the motion characteristics of a large flapping wing robot, the autonomous flight, including the whole process of takeoff, cruise and landing, is realized. An explicit complementary filtering algorithm is designed to fuse the data of inertial sensor and magnetometer, which solves the problem of attitude divergence. The Kalman filter algorithm is designed to estimate the spatial position and speed of a flapping wing robot by integrating inertial navigation with GPS (global positioning system) and barometer measurement data. The state sensing and estimation accuracy of the flapping wing robot are improved. Finally, the flying state sensing and estimation method is integrated with the flapping wing robot, and the flight experiments are carried out. The results verify the effectiveness of the proposed method, which can provide a guarantee for the flapping wing robot to achieve autonomous flight beyond the visual range.

Keywords: bionic flapping wing robot; state sensing and estimation; complementary filtering; posture; Kalman filter; multiple sensors

check for updates

Citation: Liu, G.; Wang, S.; Xu, W. Flying State Sensing and Estimation Method of Large-Scale Bionic Flapping Wing Flying Robot. *Actuators* **2022**, *11*, 213. https://doi.org/10.3390/act11080213

Academic Editor: André Preumont

Received: 22 June 2022
Accepted: 25 July 2022
Published: 31 July 2022

1. Introduction

Flying in the blue sky like a bird has always been a dream of people. Learning about and utilizing birds' body structures and flight mechanisms so as to develop bionic flapping aircraft with high maneuverability and low energy consumption has broad application prospects [1,2]. Large-scale bionic flapping wing robots developed by imitating the flight patterns of birds in nature have unique advantages in flight efficiency, wind resistance, and bionic concealment [3,4]. In order to improve the survivability of the flapping wing robot in complex environments, it is extremely important for the position and attitude calculation and state prediction of the flapping wing robot to perform beyond visual range and long-distance flight missions. Since a single sensor is affected by environmental disturbances such as gusts and magnetic field pulses during the flight of the flapping wing robot, the position and attitude estimation information of the flapping wing robot is seriously inconsistent with the actual situation [5,6]. The use of the flying state sensing and estimation method can reasonably use the measurement information of each sensor to complement the advantages and disadvantages, thereby improving the reliability of the flapping wing robot in performing tasks.

At present, there are two main methods for solving the attitude of flapping wing robots. One is to use MEMS (Micro-Electro-Mechanical System, MEMS) (Norwood, MD, USA) sensors (including accelerometers, gyroscopes and magnetometers) to estimate the attitude. This method is suitable for large- and medium-sized flapping wing robots with strong load capacity, such as Bat Bot flapper from University of Illinois at Urbana-Champaign, and pigeon flapper from Northwestern Polytechnic University [7,8]. At the same time, Kamankesh et al. and Yu et al. [9,10] studied the stability characteristics of periodic flapping wing structures and attitude stabilization of the aircraft. Verboom et al. and Hao et al. proposed a method based on periodic filtering to estimate the attitude of the flapping wing robot [11,12]. Yang et al. and Tu et al. [13,14] used Kalman filtering and adaptive attitude update methods to study the attitude information of UAVs (unmanned aerial vehicles). He et al. [15,16] proposed several adaptive control schemes and position estimation methods. In addition, Taha, Bialy, Dou, Zhang and other researchers [17–20] have also conducted in-depth research on the attitude and control of flapping wing robots. The other is the visual image pose solution method using the ground station for image transmission. This method is relatively common in small flapping wing robots, mainly including RoboBee developed by Harvard University and Delfly micro flapping wing robot developed by Karasek [21,22].

Solving the position and attitude based on flying state sensing and estimation is an important prerequisite for the flapping wing robot to achieve attitude control. Only by accurately obtaining the state parameters of the prototype can the position and attitude be controlled, and then the prototype can be guided to fly according to the predetermined trajectory by adjusting the attitude [23,24]. In a word, solving the pose of the flapping wing robot based on the flying state sensing and estimation method can improve the adaptability of the flapping wing robot to complex environments and improve the reliability of task execution.

However, different from small- and micro-sized flapping wing flying robots, the two wings of the large flapping wing robots have a large proportion of mass and inertia, and the fluctuation of the fuselage center of mass during flight cannot be ignored. The asymmetric flapping or unbalanced force leads to serious attitude deflection, and the existence of delay aggravates the inaccuracy and time inconsistency of perceptual information. Therefore, the traditional flying state sensing and estimation methods for fixed wing and rotor are difficult to provide stable and accurate data, which seriously limits the autonomous flight ability of large bionic flapping wing flying robots. Combined with the long-term accumulation of research on large-scale bionic flapping wing flying robot in the laboratory, and based on its flight characteristics in different stages of takeoff, cruise and landing, this paper proposes a method of flying state sensing and estimation, which realizes its autonomous navigation in the whole process of flight. It was tested in the actual flight experiment to verify the effectiveness of the proposed method.

2. Flying State Sensing and Estimation Scheme Design and Sensor Calibration

2.1. Mechanical Structure Analysis and Hardware Connection Design

In this paper, the bionic flapping wing robot "Phoenix" independently developed by the laboratory is taken as the research object. The characteristic parameters and flight control hardware connection of the prototype are shown in Figure 1. The design of the transmission mechanism and steering mechanism of the flapping wing robot is shown in Figure 2. The wingspan of the prototype is 2.2 m, and the average chord length is 40 cm. The driving element is a brushless DC (direct current) motor (specification: T-motor AT2308, kv2600), which drives the crank to create circular motion after deceleration by a two-stage gear set (the two-stage gear set is composed of gear 1, gear 2 and gear 3, and gear 1 is fixed on the motor shaft). Then the linkage mechanism is used to control the flutter of the wings up and down. The transmission ratio of the first stage gear (composed of gear 1 and gear 2) is 7, the transmission ratio of the second stage gear (composed of gear 2 and gear 3) is 11. The addendum circle diameter of gear 1 is 7.5 mm, the addendum circle diameter of gear 2 is 48 mm, and the addendum circle diameter of gear 3 is 73 mm. The lift and thrust generated by the periodic flapping of wings are the main source of power for the "Phoenix" flapping robot.

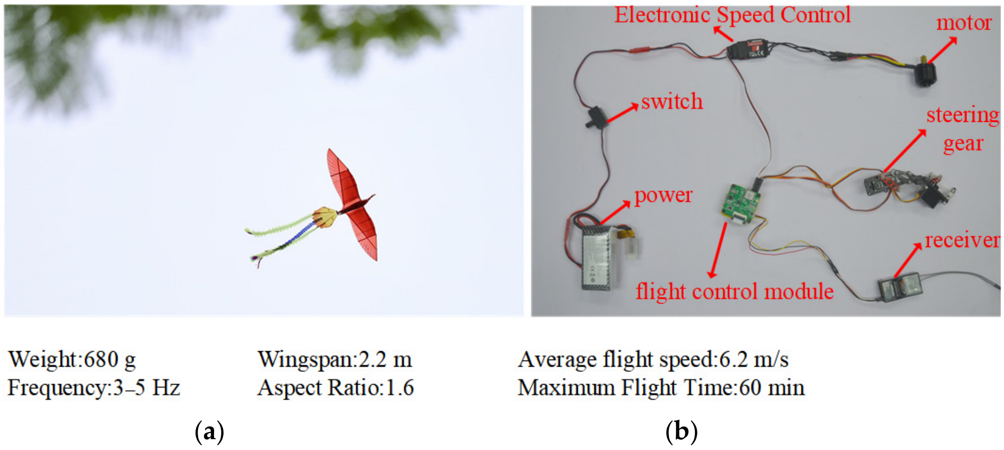


Figure 1. The characteristic parameters and flight control hardware connection of the "Phoenix" flapping wing robot. (**a**) Flapping wing robot and its characteristic parameters. (**b**) flight control hardware connection.

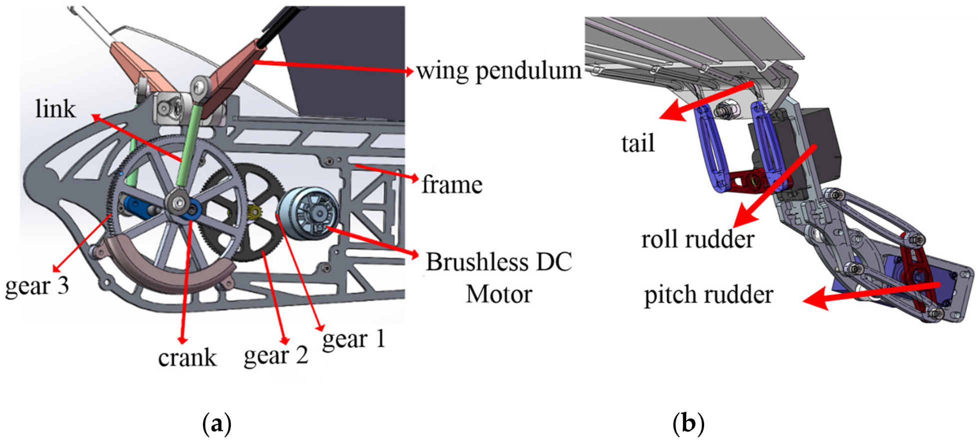


Figure 2. Transmission mechanism and steering mechanism of flapping wing robot. (**a**) Transmission mechanism. (**b**) steering mechanism.

The steering mechanism is composed of two parallelogram mechanisms, which are driven by two steering engines, respectively, and are responsible for steering adjustment of the flapping wing robot. The specification of the pitching steering engine is GDW DS298MG, the torque is 6.5 kg.cm, and the speed is 0.12 s/60°. The specification of the yaw steering engine is Emax ES08MD, the torque is 2.0 kg.cm, and the speed is 0.1 s/60°. The flight control module is powered by the signal line of the electronic speed controller. After completing the tasks of sensor data acquisition and processing, position and attitude solution, and position and attitude control, it outputs three control signals to control the adjustment of motor speed, pitch and roll attitude adjustment, respectively.

2.2. Design of Flying State Sensing and Estimation Scheme

The design of the flying state sensing and estimation function is shown in Figure 3, and the black solid arrows indicate the flow direction of data. First, the RC (remote control) sends control commands, including the selection of flight mode, sensor calibration, etc. After receiving the sensor calibration and filtering command, the flight control module will run the calibration task to calibrate the original data of IMU (inertial measurement unit), barometer and magnetometer. The calibrated data will enter the flying state sensing and estimation task module after filtering.

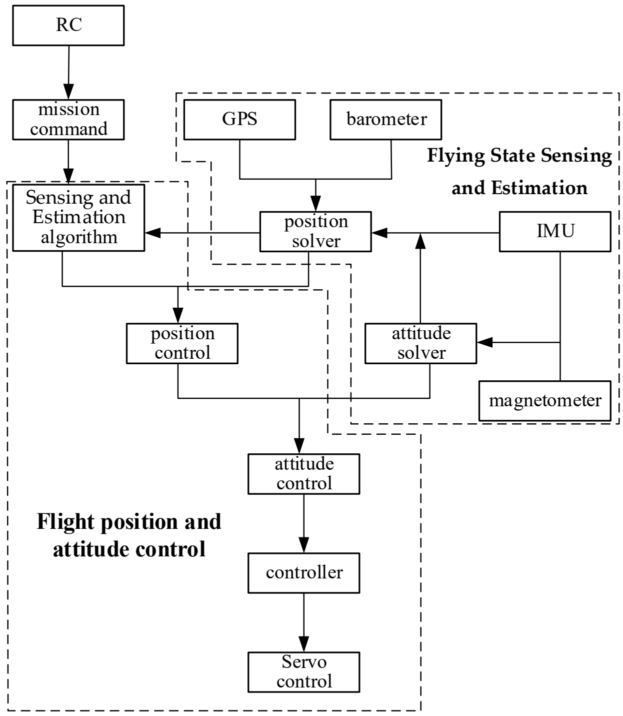


Figure 3. Functional block diagram of flying state sensing and estimation.

In the flying state sensing and estimation task module, the attitude of the "Phoenix" flapping wing robot is solved by fusing IMU and magnetometer data, and the position of the "Phoenix" flapping wing robot is solved by fusing GPS, a barometer and inertial navigation. When the position and attitude information of the "Phoenix" flapping wing robot is obtained, the flapping wing robot will run the corresponding guidance algorithm according to its own position information and enter the position and attitude control link. The position control outputs the desired attitude, and then combines the estimated attitude information of the flapping wing robot into the position and attitude control links of autonomous takeoff, cruise and landing flight to output three-way attitude control quantities. It can be seen from Figure 2 that the roll and yaw attitudes of the "Phoenix" flapping wing robot are coupled with each other. The yaw control quantities and the roll control quantities need to be converted into a control quantities output through the preset logic of the mixer, and then used for the control of the roll servo. The pitch control quantities is used to control the pitch servo.

2.3. Sensor Calibration and Filtering

Due to an assembly error and the influence of the surrounding environment, the original data output by the sensor is not accurate enough. Therefore, it is necessary to calibrate and filter the sensor data to obtain relatively accurate sensor data.

(1) Calibration of the sensor. The calibration of the sensor mainly includes the calibration of the gyroscope, magnetometer and accelerometer. The gyroscope is a device that detects the angular motion of an object, and its error can be calibrated by calculating the average value and Kalman filter by collecting the angular velocity data of three-axis output of gyroscope in static state. The magnetometer is an electronic device that obtains the heading angle by measuring the strength of the geomagnetic field. Here, the ellipsoid fitting based on the least squares method is used to calibrate the magnetometer. The calibration principle of the magnetometer can be found in the literature [12,25]. The effect comparison after calibration is shown in Figure 4. It can be seen that the center of the sphere formed by the raw magnetometer data is not at the origin, which indicates that there is an obvious offset error in the magnetometer.

The fitted data are more evenly distributed on the spherical surface and the center of the sphere is located at the origin, which indicates that the ellipsoid fitting method has achieved better results.

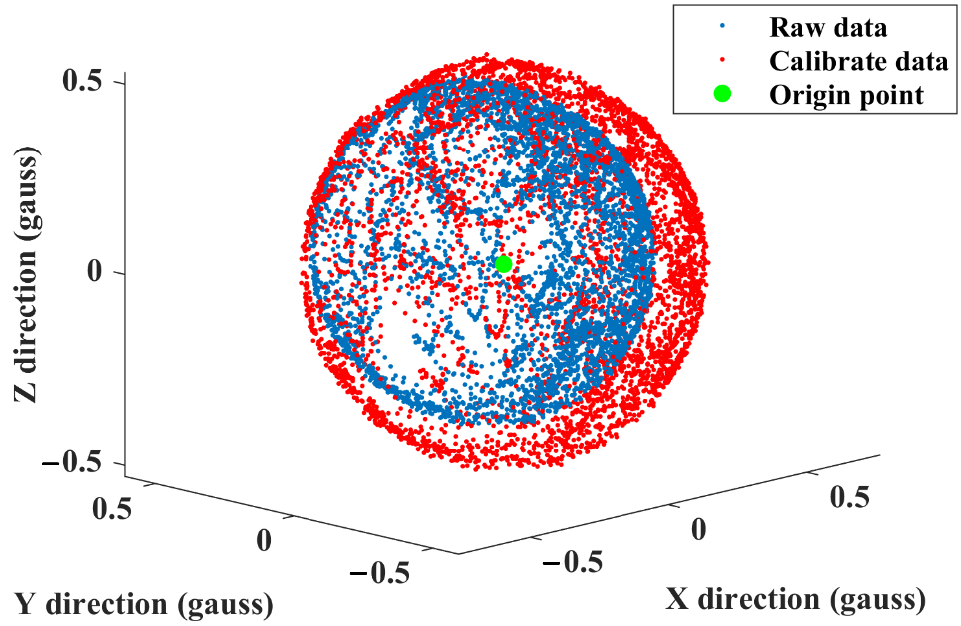


Figure 4. Ellipsoid fitting effect of magnetometer.

For the accelerometer, the ellipsoid fitting method based on nine parameters is used to calibrate it, and the comparison of calibration results is shown in Figure 5. The calibration principle of accelerometer can be found in the literature of Verboom and Hao [11,26]. It can be seen that after the accelerometer is calibrated, the data envelope forms a sphere, and the center of the sphere is closer to the origin, which shows that the ellipsoid fitting method is effective in correcting the systematic error of the accelerometer. Then, keeping the prototype in a horizontal state, the raw data is collected and the calibrated data output by the IMU and the magnetometer when stationary are shown in Table 1. It can be seen that the influence of the sensor system error is basically eliminated by the calibration.

(2) Sensor filtering. During the flapping process of the flapping wing robot, a lot of vibration noise will be mixed with the flapping of the wings, which will seriously affect the attitude and position estimation, so filtering processing is required.

Table 1. Data comparison table before and after sensor calibration.

Static Measurement		Raw Data	Calibrated Data
Accelerometer (m/s^2)	X	0.42	0.15
	Y	0.04	−0.20
	Z	10.19	−9.69
	Module	10.21	9.70
Gyroscope (°/s)	X	2.38	−0.04
	Y	1.70	0.005
	Z	1.34	0.05
	Module	3.17	0.06
Magnetometer	X	0.15	0.16
	Y	0.49	0.47
	Z	0.14	0.14
	Module	0.53	0.51

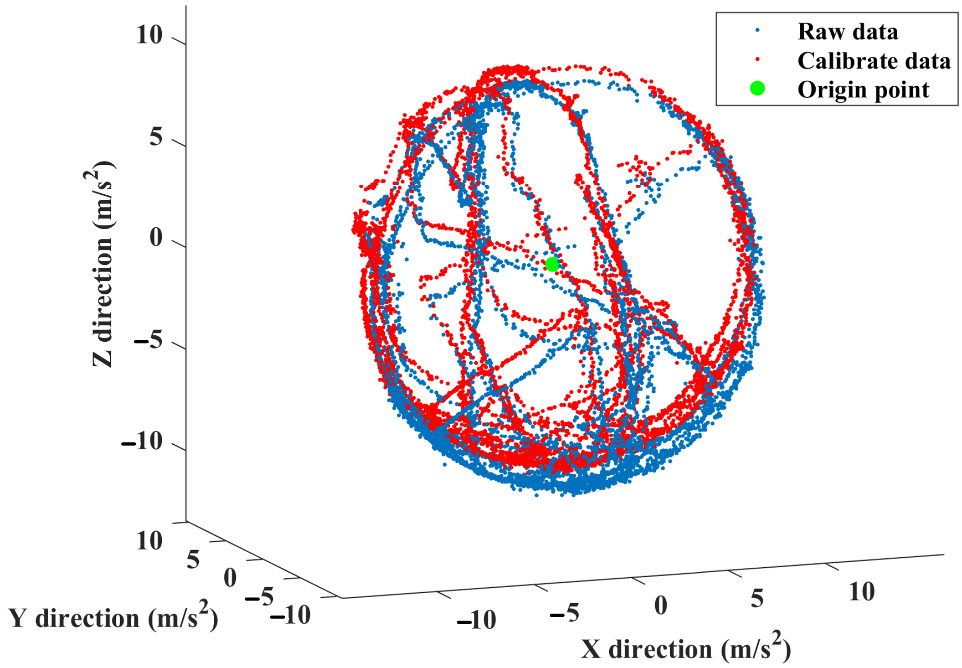


Figure 5. Ellipsoid fitting effect of accelerometer.

The comparisons of the accelerometer and gyroscope signals before and after calibration filtering are shown in Figures 6 and 7, respectively. It can be seen that the fluctuation of the horizontal acceleration and the three-axis angular velocity data curve after calibration filtering is significantly reduced, and the signal-to-noise ratio and anti-interference ability are enhanced. At the same time, because the magnetometer signal is not easily affected by mechanical vibration, the sliding mean filter is used to suppress the magnetic field pulse noise.

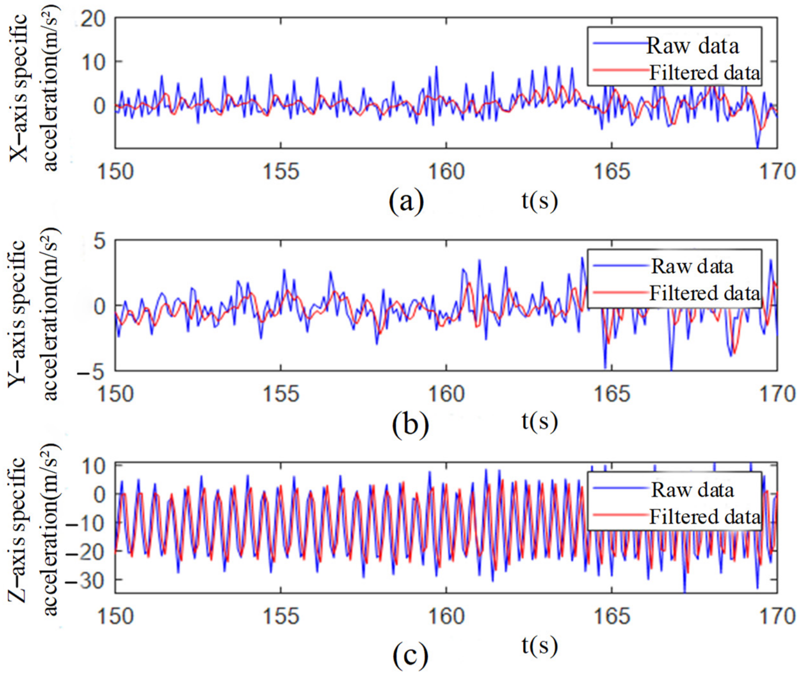


Figure 6. Comparison diagram of accelerometer signal before and after calibration and filtering. (**a**) Specific acceleration in X-axis. (**b**) Specific acceleration in Y-axis. (**c**) Specific acceleration in Z-axis.

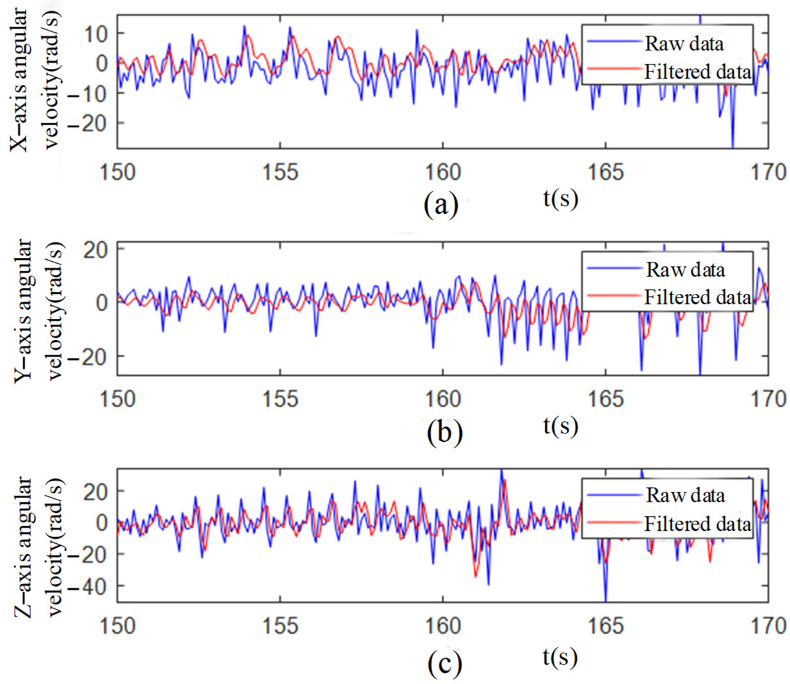


Figure 7. Comparison diagram of gyro signal before and after calibration and filtering. (**a**) Angular velocity of X-axis. (**b**) Angular velocity of Y-axis. (**c**) Angular velocity of Z-axis.

3. Solution Method of Position and Attitude for Flying State Sensing and Estimation

3.1. Coordinate System Definition and Attitude Kinematics

In order to describe the state of the flapping wing robot in three-dimensional space, including the position of its own center of mass and its attitude in space, the coordinate system fixed with flapping wing robot is defined as the body system $\{O_B X_B Y_B Z_B\}$. The origin of the body system is selected at the position of the centroid of the flapping wing robot. The reference coordinate system that characterizes the attitude change of the flapping wing robot is called the inertial frame $\{O_I X_I Y_I Z_I\}$, as shown in Figure 8.

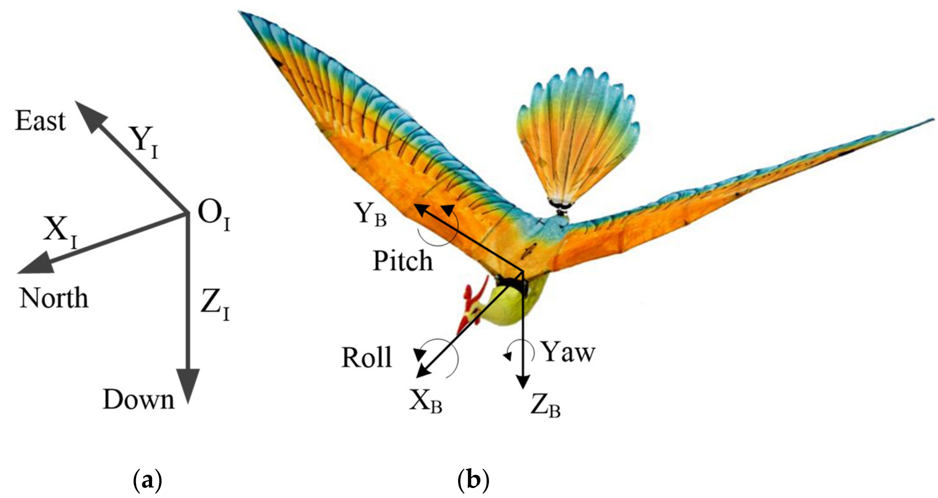


Figure 8. Schematic diagram of inertial frame and body system orientation. (**a**) Schematic diagram of inertial system. (**b**) Schematic diagram of body system.

The rotation matrix ${}^{I}_{B}\boldsymbol{R}$ is used to describe the attitude change of the body system {B} relative to the inertial frame {I}, which is called the rotation transformation matrix of the body system {B} relative to the inertial frame {I}. Similarly, the rotation matrix ${}^{B}_{I}\boldsymbol{R}$ is used to describe the attitude of the inertial frame relative to the body system. Due to the large range of attitude changes of the flapping wing robot, in order to avoid the appearance of attitude jumping, the unit quaternion $q = [q_0\ q_1\ q_2\ q_3]$ is used to represent the attitude of the flapping wing robot. The quaternion represents the rotation matrix ${}^{\mathrm{I}}_{\mathrm{B}}\boldsymbol{R}$ of the body system relative to the inertial frame as shown in Equation (1)

$$ {}^{\mathrm{I}}_{\mathrm{B}}\boldsymbol{R} = \begin{bmatrix} 2(q_0^2+q_1^2)-1 & 2(q_1q_2-q_0q_3) & 2(q_1q_3+q_0q_2) \\ 2(q_1q_2+q_0q_3) & 2(q_0^2+q_2^2)-1 & 2(q_2q_3-q_0q_1) \\ 2(q_1q_3-q_0q_2) & 2(q_2q_3+q_0q_1) & 2(q_0^2+q_3^2)-1 \end{bmatrix} \tag{1} $$

Quaternion was used to solve the differential equation of the attitude of flapping wing robot, as shown in Equation (2), and the first-order Runge–Kutta method was also used to update the quaternion differential equation. Then the quaternion expression of the current moment was obtained as shown in Equation (3). Due to the instability of the gyroscope's bias, the bias error changes gradually over time. Therefore, the key to attitude calculation is to compensate the change of gyroscope bias in real time. In this paper, an explicit complementary filtering algorithm is used to filter the gyroscope data to eliminate the interference of the deviation instability on the attitude solution.

$$ \dot{\hat{\boldsymbol{q}}} = \frac{1}{2}\hat{\boldsymbol{q}} \otimes \left(\hat{\boldsymbol{\omega}}^b + \delta\right) \tag{2} $$

where $\dot{\hat{\boldsymbol{q}}}$ is the derivative of quaternion, $\hat{\boldsymbol{q}}$ is the estimated value of the quaternion at any time; $\hat{\boldsymbol{\omega}}^b$ is the angular velocity vector under the body system; δ is the error compensation term; $\otimes$ is the Quaternion multiplication.

$$ \hat{\boldsymbol{q}} = \hat{\boldsymbol{q}}_{\mathrm{init}} + \dot{\hat{\boldsymbol{q}}}\Delta t \tag{3} $$

where $\hat{\boldsymbol{q}}_{\mathrm{init}}$ is the quaternion at the initial moment; Δt is the integration time.

3.2. Periodic Equivalent Strategy

On the basis of the coordinate system in Figure 8, the whole-body coordinate system $\{O_{sys}X_{sys}Y_{sys}Z_{sys}\}$ is added, as shown in Figure 9. $\boldsymbol{r}_{sys}$ is the position coordinate vector of the center of mass of the whole-body system relative to the inertial reference frame. $\boldsymbol{r}_0$ is the position vector of the center of mass of the fuselage relative to the center of mass of the whole-body system. $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ are the position vectors of the left and right wings relative to the center of mass of the whole system, respectively. The mass of the fuselage is m_0, and the masses of the left and right wings are m_1 and m_2, respectively.

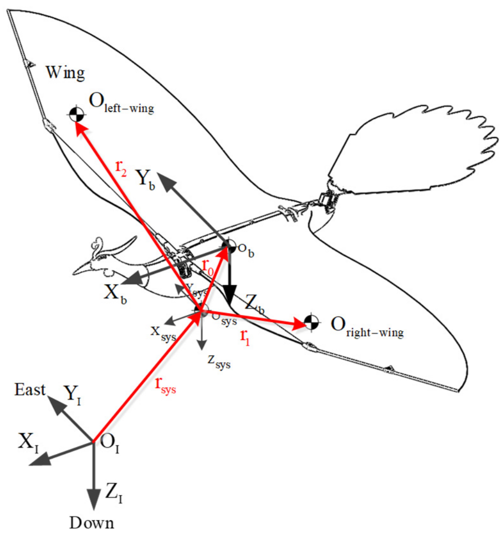


Figure 9. Vector position diagram of aircraft centroid.

Because the center of mass of the fuselage fluctuates relative to the center of mass of the whole system during the flapping process of the aircraft, and the sensor is fixedly connected to the fuselage, the height measured by the sensor is not the actual flying height of the aircraft. The relationship between the center of mass of the fuselage and the center of mass of the whole system is:

$$m_0\boldsymbol{r}_0 + m_1\boldsymbol{r}_1 + m_2\boldsymbol{r}_2 = (m_0 + m_1 + m_2)\boldsymbol{r}_{\text{sys}} \tag{4}$$

The position vector of the body's center of mass can be simplified as:

$$\begin{aligned} r_0 &= \frac{(m_0+m_1+m_2)\boldsymbol{r}_{\text{sys}} - m_1\boldsymbol{r}_1 - m_2\boldsymbol{r}_2}{m_0} \\ &= \frac{m_0+m_1+m_2}{m_0}\boldsymbol{r}_{sys} - \frac{m_1}{m_0}\boldsymbol{r}_1 - \frac{m_2}{m_0}\boldsymbol{r}_2 \end{aligned} \tag{5}$$

Equation (5) can be rewritten as

$$\boldsymbol{r}_0 = \overset{\wedge}{\boldsymbol{r}}_{\text{sys}} - \overset{\wedge}{\boldsymbol{r}}_1 - \overset{\wedge}{\boldsymbol{r}}_2 \tag{6}$$

where

$$\begin{cases} \overset{\wedge}{\boldsymbol{r}}_{\text{sys}} = \frac{m_0+m_1+m_2}{m_0}\boldsymbol{r}_{\text{sys}} \\ \overset{\wedge}{\boldsymbol{r}}_1 = \frac{m_1}{m_0}\boldsymbol{r}_1 \\ \overset{\wedge}{\boldsymbol{r}}_2 = \frac{m_2}{m_0}\boldsymbol{r}_2 \end{cases} \tag{7}$$

Then the fluctuation equation of the center of mass of the fuselage relative to the center of mass of the whole system is:

$$\begin{aligned} \boldsymbol{r}_0 - \overset{\wedge}{\boldsymbol{r}}_{\text{sys}} &= -\overset{\wedge}{\boldsymbol{r}}_1 - \overset{\wedge}{\boldsymbol{r}}_2 \\ &= \Delta\text{hsin}(2\pi f t) \end{aligned} \tag{8}$$

The fluctuation curve of the center of mass of the fuselage relative to the center of mass of the whole system is obtained, as is shown in Figure 10.

Then the change equation of the center of mass of the fuselage is:

$$h(t) = H(t) \pm \Delta\text{hsin}(2\pi f t) \tag{9}$$

where $H(t)$ is the height change of the center of mass of the whole system. Δh is the fluctuation range of the center of mass of the fuselage relative to the center of mass of the whole system. f is the flapping frequency of the wings.

(1) When the aircraft is in cruise flight

$\left|\overset{\wedge}{\boldsymbol{r}}_{\text{sys}}\right| = H(t)$ is a constant value, then the variation curve of the center of mass of the fuselage relative to the center of mass of the whole system is shown in Figure 11.

(2) When the aircraft is in maneuvering flight

$\left|\overset{\wedge}{\boldsymbol{r}}_{\text{sys}}\right| = H(t)$ a is a variable value, then the variation curve of the center of mass of the fuselage relative to the center of mass of the whole system is shown in Figure 12.

3.3. Attitude Solution Based on Explicit Complementary Filtering

Because the traditional complementary filtering algorithm is not suitable for the flapping wing system with severe vibration [9,27], a more widely practical display complementary filtering algorithm was used to solve the attitude of the "Phoenix" flapping wing robot. It combines the data of the accelerometer, magnetometer and gyroscope, and uses the estimation error of the accelerometer and the magnetometer to a certain constant vector to correct the zero-drift change of the three axes of the gyroscope so that the angular velocity data is closer to the real value, thereby suppressing the attitude divergent.

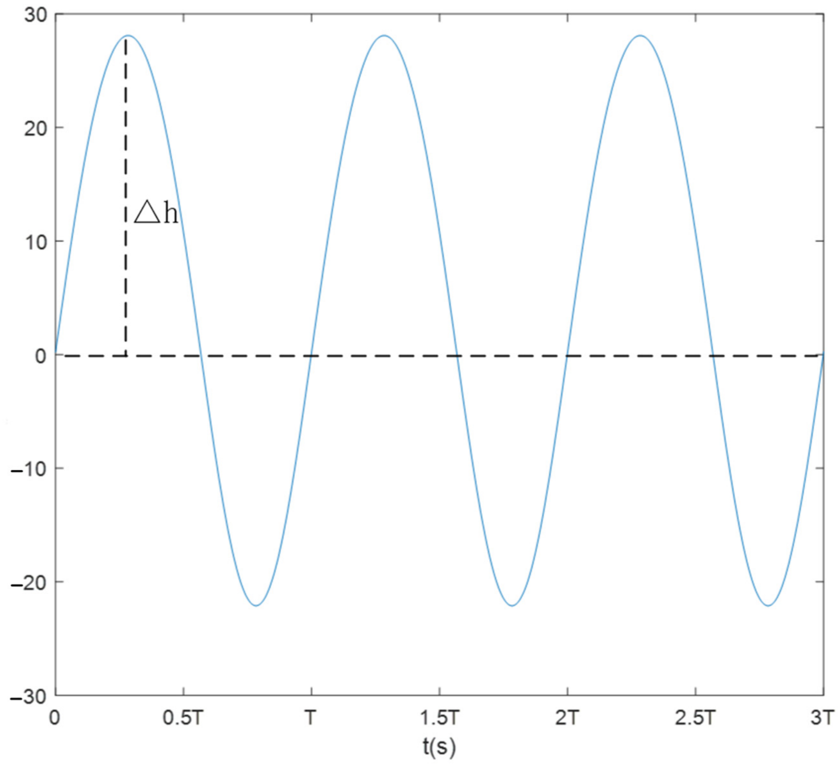


Figure 10. Height fluctuation curve of fuselage centroid relative to the centroid of the whole machine system.

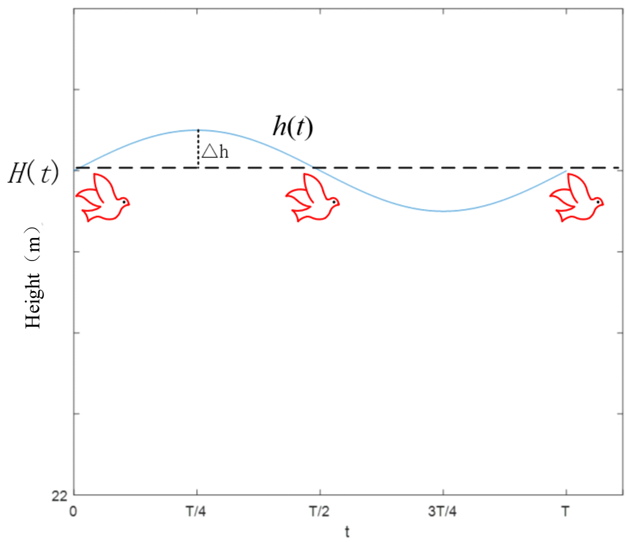


Figure 11. Variation curve of centroid height and periodic equivalent height during cruise flight.

3.3.1. Gravitational Acceleration Compensation

The unit vector of the estimated gravitational acceleration in the inertial frame is shown in Equation (10).

$$^{I}\bar{g} = \frac{^{I}g}{\|^{I}g\|} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^{T} \tag{10}$$

where ^{I}g and $\|^{I}g\|$ are the gravitational acceleration vector and modulo length (m/s^2) in the inertial frame, respectively. With the help of the rotation matrix (1), the representation of the estimated unit vector of the gravitational acceleration in the body system can be obtained, which is given by the Equation (11).

$$\overline{v} = {}^{\mathrm{B}}_{\mathrm{I}}\boldsymbol{R} * \overline{\boldsymbol{g}} = {}^{\mathrm{I}}_{\mathrm{B}}\boldsymbol{R}^{\mathrm{T}} * \overline{\boldsymbol{g}} = \left[2(q_1q_3 - q_0q_2)\ 2(q_2q_3 + q_0q_1)\ 2(q_0^2 + q_3^2) - 1\right]^{\mathrm{T}} \tag{11}$$

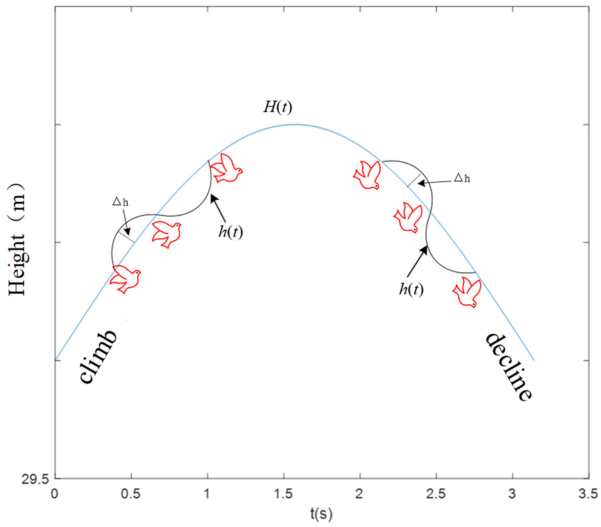


Figure 12. Centroid height variation curve and periodic equivalent height during maneuvering flight.

The vector of the specific acceleration measured by the accelerometer under the body system is shown in Equation (12),

$${}^{\mathrm{B}}\boldsymbol{f} = {}^{\mathrm{B}}\boldsymbol{a} - {}^{\mathrm{B}}\boldsymbol{g} \tag{12}$$

where ${}^{\mathrm{B}}\boldsymbol{a}$ is the inertial acceleration vector under the body system; ${}^{\mathrm{B}}\boldsymbol{g}$ is the gravitational acceleration vector under the body system.

When the flapping robot is in equilibrium,${}^{\mathrm{B}}\boldsymbol{a} \approx 0$,then the output signal of the accelerometer is the estimation of the gravitational acceleration, that is,${}^{\mathrm{B}}\hat{g} = -{}^{\mathrm{B}}f$. After normalization, the Equation (13) will be obtained,

$$\hat{v} = \frac{{}^{\mathrm{B}}\hat{g}}{\|{}^{\mathrm{B}}\hat{g}\|} = \frac{-{}^{\mathrm{B}}f}{\|{}^{\mathrm{B}}f\|} \tag{13}$$

Then the unit vector $\overline{v}$ of the estimated gravitational acceleration under the body system is cross-multiplied with the unit vector $\hat{v}$ of the measured gravitational acceleration to obtain the gravity error correction vector, as shown in Equation (14), which is used to compensate the zero-drift change of the gyroscope.

$$e_1 = \hat{v} \times \overline{v} \tag{14}$$

3.3.2. Magnetic Deflection Compensation

In this paper, the magnetometer is used to correct the yaw angle, and the divergence of the yaw angle is suppressed by the invariance of the magnetic field resultant vector in the horizontal plane. Firstly, the geomagnetic field signal measured under the body system is converted to the inertial frame through the rotation matrix, as shown in Equation (15).

$${}^{\mathrm{I}}\boldsymbol{m} = {}^{\mathrm{I}}_{\mathrm{B}}\boldsymbol{R} \times {}^{\mathrm{B}}\boldsymbol{m} \tag{15}$$

where ${}^{\mathrm{B}}\boldsymbol{m}$ is the geomagnetic vector under the body system; ${}^{\mathrm{I}}\boldsymbol{m}$ is the geomagnetic vector in the inertial frame.

The equation for solving the geomagnetic declination in the inertial frame is shown in Equation (16).

$$\hat{\psi}_0 = \arctan(\mathrm{m}_{\mathrm{y}}^{\mathrm{I}}/m_{\mathrm{x}}^{\mathrm{I}}) \tag{16}$$

The geomagnetic declination of Shenzhen is $\hat{\psi}_0 = -0.052$. Thus, it can be obtained that the difference between the measured geomagnetic declination and the predicted geomagnetic declination is $\Delta\psi_0 = \hat{\psi}_0 - \overline{\psi}_0$, which is rewritten into a vector form, as shown in Equation (17).

$$\boldsymbol{e} = \begin{bmatrix} 0 & 0 & \Delta\psi_0 \end{bmatrix}^{\mathrm{T}} \tag{17}$$

Convert the geomagnetic error vector into the body system, as shown in Equation (18).

$$\boldsymbol{e}_2 = {}_{\mathrm{I}}^{\mathrm{B}}\boldsymbol{R}\boldsymbol{e} \tag{18}$$

Adding the gravitational acceleration error vector and the geomagnetic error vector, and then the error compensation vector was obtained through the proportional–integral link, as shown in Equation (19). The error compensation vector and the angular velocity vector output by the gyroscope are added to obtain the corrected angular velocity vector, which is then substituted into the quaternion differential equation for integration. The quaternion is then iteratively updated to finally obtain the precise attitude information of the "Phoenix" flapping wing robot.

$$\delta = K_{\mathrm{p}}(\boldsymbol{e}_1 + \boldsymbol{e}_2) + K_{\mathrm{I}}\int(\boldsymbol{e}_1 + \boldsymbol{e}_2) \tag{19}$$

where K_{p} and K_{I} are proportional coefficient and integral coefficient, respectively.

When the "Phoenix" flapping wing robot performs the attitude calculation task, it needs to obtain the accelerometer and magnetometer data to estimate the initial attitude of the "Phoenix" flapping wing robot. In the equilibrium state of the "Phoenix" flapping wing robot, the three-axis accelerometer data is collected to calculate the initial roll angle and pitch angle of the flapping wing robot, as shown in Equations (20) and (21).

$$\varphi = \arctan(a_{\mathrm{y}}^{\mathrm{b}}/a_{\mathrm{x}}^{\mathrm{b}}) \tag{20}$$

$$\theta = \arcsin(a_{\mathrm{x}}^{\mathrm{b}}) \tag{21}$$

where φ is the roll angle of flapping wing robot, and θ is the pitch angle.

The geomagnetic vector obtained by the magnetometer under the body system is projected into the horizontal plane of the inertial frame, as shown in Equations (22) and (23).

$$m_{\mathrm{x}} = m_{\mathrm{x}}^{\mathrm{b}}\cos\theta + m_{\mathrm{y}}^{\mathrm{b}}\sin\theta\sin\varphi - m_{\mathrm{z}}^{\mathrm{b}}\sin\theta\cos\varphi \tag{22}$$

$$m_{\mathrm{y}} = m_{\mathrm{y}}^{\mathrm{b}}\cos\varphi + m_{\mathrm{z}}^{\mathrm{b}}\sin\varphi \tag{23}$$

Initialize the yaw angle (looking from the tail to the nose, the right is positive) using the magnetometer data in the north (X-axis) and east (Y-axis) directions, as shown in Equation (24).

$$\psi = \mathrm{atan2}(-m_{\mathrm{y}}, m_{\mathrm{x}}) - \mathrm{mag\_decline} \tag{24}$$

where mag_decline is the magnetic declination angle, which is about −0.052 rad in Shenzhen.

Then, the initial Euler angle is converted into a quaternion by Equation (25), and then the quaternion vector at the initial moment can be obtained by normalizing.

$$\begin{cases} q_0 = \cos(\psi/2)\cos(\theta/2)\cos(\varphi/2) + \sin(\psi/2)\sin(\theta/2)\sin(\varphi/2) \\ q_1 = \cos(\psi/2)\cos(\theta/2)\sin(\varphi/2) - \sin(\psi/2)\sin(\theta/2)\cos(\varphi/2) \\ q_2 = \cos(\psi/2)\sin(\theta/2)\cos(\varphi/2) + \sin(\psi/2)\cos(\theta/2)\sin(\varphi/2) \\ q_3 = -\cos(\psi/2)\sin(\theta/2)\sin(\varphi/2) + \sin(\psi/2)\cos(\theta/2)\cos(\varphi/2) \end{cases} \tag{25}$$

After the quaternion initialization is completed, the offset of the three-axis gyroscope is compensated by displaying the complementary filtering algorithm to obtain the error compensation vector δ, and then the quaternion is updated by using Equations (2) and (3). If the solved quaternion is a finite real number, the solution is considered successful, otherwise the solution result is discarded. After the quaternion solution is completed, the normalization processing is carried out, and the corresponding rotation matrix ${}^{\mathrm{I}}_{\mathrm{B}}\boldsymbol{R}$ is updated. In this way, the attitude solution task of one cycle (0.01 s) is completed.

3.4. Position Solution Based on Kalman Filter

The position solution of the "Phoenix" flapping wing robot in the horizontal direction requires the use of sensor data. The GPS sensor output data has long-term stability and does not diverge over time, but the data refresh rate is low and is easily affected by surrounding obstructions. The INS (inertial navigation system) can be used in any complex environment, with high accuracy in a short time, but the position measured by INS will produce integral drift for a long time. A barometer and GPS can be used to estimate altitude information. In a windless and stable flight environment, the barometer is used to perceive the altitude information of the "Phoenix" flapping wing robot. When the flapping robot is disturbed by strong winds or its own flapping frequency is too fast, it will cause large fluctuations in the surrounding air pressure, resulting in inaccurate data output of the barometer [20,23]. When the output data measured by the barometer fluctuates unreasonably, the altitude data output by GPS is selected as the height measurement information of the flapping robot.

In summary, it is difficult for each navigation system to be used independently to meet the reliability requirements. A feasible way to improve the accuracy of the navigation system is the integrated navigation technology, that is, two or more navigation systems are used to measure and solve the same navigation information to form a more accurate measurement value.

3.4.1. Design of Kalman Filter

In this study, the Kalman filter is used to fuse multi-sensor data to solve the position and velocity information of the "Phoenix" flapping wing robot [13,28]. The current state can be calculated only by inputting the measured value of the state vector in the current solution cycle (which may contain multiple sensor data) and the estimated value of the previous solution cycle.

Setting $x = [p\ v\ b]^{\mathrm{T}}$ as the northward state vector of the flapping wing robot, where p represents the position; v represents the velocity; b represents the inertial acceleration bias. The solution using the Kalman filter includes prediction process and update process. The prediction process is shown in Equations (26) and (27).

$$\hat{\boldsymbol{x}}'_k = \boldsymbol{F}\hat{\boldsymbol{x}}_{k-1} + \boldsymbol{B}a_k + \boldsymbol{\Gamma}\zeta_{k-1} \tag{26}$$

where $\boldsymbol{F}$ is the state transition matrix; B is the control matrix; $\boldsymbol{\Gamma}$ is the process noise driving matrix. $\hat{\boldsymbol{x}}'_k$ and $\hat{\boldsymbol{x}}_{k-1}$ are the prior state vector at time k and the posterior state vector at time (k–1), respectively; a_k is the inertial acceleration in the inertial frame at time k; ζ_{k-1} is the zero mean white noise.

$$\hat{\boldsymbol{P}}'_k = \boldsymbol{F}\hat{\boldsymbol{P}}_{k-1}\boldsymbol{F}^{\mathrm{T}} + \boldsymbol{Q} \tag{27}$$

where $\hat{\boldsymbol{P}}'_k$ is the prior estimated state error covariance matrix at time k; $\hat{\boldsymbol{P}}_{k-1}$ is the posterior estimated state error covariance matrix at time k–1; $\boldsymbol{Q}$ is the process excitation noise covariance matrix.

The update process of the Kalman filter is shown in Equations (28)–(31). Equations (26) and (28) are collectively referred to as the Kalman filter system equation of the "Phoenix" flapping wing robot.

$$\boldsymbol{z}_k = \boldsymbol{H}\hat{\boldsymbol{x}}'_k + \boldsymbol{v}_k \tag{28}$$

$$\hat{\boldsymbol{x}}_k = \hat{\boldsymbol{x}}'_k + \boldsymbol{K}(\boldsymbol{z}_k - \boldsymbol{H}\hat{\boldsymbol{x}}'_k) \tag{29}$$

$$\boldsymbol{K} = \frac{\hat{\boldsymbol{P}}_k^{'}\boldsymbol{H}^T}{\boldsymbol{H}\boldsymbol{P}_k^{'}\boldsymbol{H}^T + \boldsymbol{R}} \tag{30}$$

$$\hat{\boldsymbol{P}}_{\boldsymbol{k}} = (\boldsymbol{I} - \boldsymbol{K}\boldsymbol{H})\hat{\boldsymbol{P}}_{\boldsymbol{k}}^{'} \tag{31}$$

where $\boldsymbol{z}_k$ is measurement vector, $\boldsymbol{z}_k = \left[p_k^m \ v_k^m\right]^{\mathrm{T}}$; p_k^m and v_k^m are the position and velocity measured by the sensor, respectively; v_k is the measurement error vector, $v_k = [p_e' v_e']^{\mathrm{T}}$; p_e' and v_e' represent the measurement error of position and velocity, respectively. $\boldsymbol{I}$ is the three order unit matrix; $\boldsymbol{K}$ is the Kalman gain matrix; $\boldsymbol{H}$ is the measurement matrix; $\boldsymbol{R}$ is the measurement noise covariance matrix.

The design of the Kalman filter focuses on three covariance matrices, namely the initial state error covariance matrix $\hat{\boldsymbol{P}}_0$, the process excitation noise covariance matrix $\boldsymbol{Q}$ and the measurement noise covariance matrix $\boldsymbol{R}$. Here, the initial state error covariance matrix $\hat{\boldsymbol{P}}_0$ is a third-order square matrix, the elements on the diagonal are the variance of the initial estimation error of the state vector, and the elements on the non-diagonal line are the covariance of the initial estimation error of the state vector.

When setting the process excitation noise covariance matrix, it can be seen from the system model that the errors of position and speed are related to the change of acceleration, and the relationship is shown in Equations (32) and (33). Assuming that the variance of the acceleration is $\boldsymbol{D}(a)$, the expressions of the position and velocity variances are shown in (34) and (35) according to the properties of the variance, thus determining the elements on the diagonal of the $\boldsymbol{Q}$ matrix. The calculation of covariance is given by Equation (36), and the correlation coefficient is taken as $\rho = \pm 1$. When the direction of position and velocity is the same as the direction of acceleration, the correlation coefficient $\rho = 1$, otherwise $\rho = -1$. In summary, under the condition that the state quantity is positively correlated with the driving quantity, the final expression of the process excitation noise $\boldsymbol{Q}$ matrix is shown in Equation (37).

$$\Delta v = a\Delta t \tag{32}$$

$$\Delta p = v_0\Delta t + \frac{1}{2}a\Delta t^2 \tag{33}$$

$$D(v) = D(at) = \Delta t^2 D(a) \tag{34}$$

$$D(p) = D(\frac{1}{2}a\Delta t^2) = \frac{1}{4}\Delta t^4 D(a) \tag{35}$$

$$\mathrm{cov}(X, Y) = \rho\sqrt{D(X)D(Y)} \tag{36}$$

where $D(X)$ and $D(Y)$ represent the variance of X variable and Y variable, respectively; ρ represents the correlation coefficient, $-1 \leq \rho \leq 1$;

$$\boldsymbol{Q} = \begin{bmatrix} \frac{1}{4}\Delta t^4 & \frac{1}{2}\Delta t^3 & \frac{1}{2}\Delta t^2 \\ \frac{1}{2}\Delta t^3 & \Delta t^2 & \Delta t \\ \frac{1}{2}\Delta t^2 & \Delta t & 1 \end{bmatrix} D(a) \tag{37}$$

The measurement noise covariance is mainly related to the measurement accuracy of the sensor, and the measurement error can be determined according to the horizontal (vertical) positioning accuracy factor output by the GPS sensor. It is also assumed that the GPS measurements of position and velocity are independent of each other, then the non-diagonal elements of the measurement noise covariance matrix $\boldsymbol{R}$ are set to 0, and the diagonal elements are related to the positioning accuracy factor. The final expression of the measurement noise covariance moment $\boldsymbol{R}_{\mathrm{H}}$ in the horizontal direction is shown in Equation (38).

$$\boldsymbol{R}_{\mathrm{H}} = \begin{bmatrix} D(hdop) & 0 \\ 0 & D(vdop) \end{bmatrix} \tag{38}$$

where $D(hdop)$ and $D(vdop)$ represent the variance of the horizontal position and horizontal velocity positioning accuracy factor, respectively. The measurement noise covariance $\boldsymbol{R}_{\mathrm{v}}$ in the vertical direction is related to the vertical positioning factor of the barometer or GPS. When the barometer is used to measure the height information of the "Phoenix" flapping wing robot, the expression of $\boldsymbol{R}_{\mathrm{v}}$ is shown in Equation (39).

$$R_{\mathrm{V}} = D(baro) \tag{39}$$

where $D(baro)$ is the variance of the height measured by the barometer.

When the error of the barometer output data is large, GPS is used to sense the height change of the flapping wing robot, and the expression of $\boldsymbol{R}_{\mathrm{v}}$ is shown in Equation (40).

$$R_{\mathrm{V}} = D(vdop) \tag{40}$$

where $D(vdop)$ is the variance of the vertical positioning precision factor of GPS.

3.4.2. Improvement of the Kalman Filter Algorithm

(1) H∞ filter

If the prior information of the covariance matrices $\boldsymbol{Q}$, $\boldsymbol{R}$, and $\boldsymbol{P}_0$ in the Kalman filter equation is uncertain, when the flapping robot is disturbed by gusts, it will easily lead to inaccuracy of the system model and unclear statistical characteristics of noise. At this time, it is necessary to appropriately increase the value on the diagonal of the process noise covariance matrix $\boldsymbol{Q}$ to increase the utilization weight of the real-time measured value and reduce the utilization weight of the predicted value. This process has great randomness, and it is impossible to determine the appropriate value of the process noise covariance matrix $\boldsymbol{Q}$ to optimize the performance of the filter. H∞ filtering is an effective method to solve the above problems [19,22], which is of great help to improve the robustness of the system. The discrete H∞ filter system equations and algorithms are shown in Equations (41) and (42), respectively.

$$\begin{aligned} \hat{\boldsymbol{x}}'_{k+1} &= \boldsymbol{F}\hat{\boldsymbol{x}}_k + \boldsymbol{B}\boldsymbol{a}_k + \boldsymbol{w}_k \\ \boldsymbol{z}_k &= \boldsymbol{H}\hat{\boldsymbol{x}}'_k + \boldsymbol{v}_k \\ \boldsymbol{y}_k &= \boldsymbol{L}_k\hat{\boldsymbol{x}}'_k \end{aligned} \tag{41}$$

where y_k is the estimated vector, which can be a linear combination of state variables, and it is necessary to ensure that $\boldsymbol{L}_k$ is full rank; w_k, v_k are system noise and measurement noise, respectively.

$$\begin{aligned} \hat{\boldsymbol{x}}'_{k+1} &= \boldsymbol{F}\hat{\boldsymbol{x}}'_k + \boldsymbol{F}\boldsymbol{K}\left(\boldsymbol{z}_k - \boldsymbol{H}\hat{\boldsymbol{x}}'_k\right) \\ \boldsymbol{K} &= \boldsymbol{P}'_k\left(\boldsymbol{I} + \boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{H}\boldsymbol{P}'_k\right)^{-1}\boldsymbol{H}\boldsymbol{R}^{-1} \\ \boldsymbol{P}'_{k+1} &= \boldsymbol{F}\boldsymbol{P}'_k\left(\boldsymbol{I} + \boldsymbol{H}^{T}\boldsymbol{R}^{-1}\boldsymbol{H}\boldsymbol{P}'_k\right)^{-1}\boldsymbol{F}^{\mathrm{T}} + \boldsymbol{Q} \end{aligned} \tag{42}$$

After many experiments, the values of each covariance matrix are as follows. Among them, initial state error covariance matrix and process noise matrix in the X direction:

$$\hat{\boldsymbol{P}}_0^{\mathrm{x}} = \begin{bmatrix} 8 & 0 & 0 \\ 0 & 0.18 & 0 \\ 0 & 0 & 6.4 \times 10^{-6} \end{bmatrix}, \ \boldsymbol{Q}_{\mathrm{x}} = \begin{bmatrix} 1.15 & 0 & 0 \\ 0 & 0.8 & 0 \\ 0 & 0 & 1 \times 10^{-6} \end{bmatrix}$$

Initial state error covariance matrix and process noise matrix in Y direction:

$$\hat{\boldsymbol{P}}_0^{\mathrm{y}} = \begin{bmatrix} 32 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 6.4 \times 10^{-6} \end{bmatrix}, \ \boldsymbol{Q}_{\mathrm{y}} = \begin{bmatrix} 5.8 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 1 \times 10^{-6} \end{bmatrix}$$

Horizontal measurement noise matrix:

$$\boldsymbol{R}_{\mathrm{H}} = \begin{bmatrix} 1 & 0 \\ 0 & 0.04 \end{bmatrix}$$

Initial state error covariance matrix and process noise matrix in Z direction:

$$\hat{\boldsymbol{P}}_0^{\mathrm{Z}} = \begin{bmatrix} 12.5 & 0 & 0 \\ 0 & 3.38 & 0 \\ 0 & 0 & 1 \times 10^{-4} \end{bmatrix}, \ \boldsymbol{Q}_{\mathrm{z}} = \begin{bmatrix} 3.5 & 0 & 0 \\ 0 & 2.0 & 0 \\ 0 & 0 & 1 \times 10^{-6} \end{bmatrix}$$

When using barometer data to represent height measurements, the measurement noise in the Z direction is:

$$R_{\mathrm{V}} = 1.524$$

When using the GPS output data to represent the height measurement value, the measurement noise in the Z direction is:

$$R_{\mathrm{V}} = 9$$

(2) Suppression of filter divergence

In the actual flight experiment, it was found that with the prolongation of flight time, the navigation parameters output by the Kalman filter tended to be stable, but the deviation from the real value became larger and larger. This is because the torsion, bending and other deformations will occur during the flapping process of the wings, which makes it difficult to establish an accurate mechanism model to analyze the force of the flapping robot during flight. Therefore, the physical model error is the main factor leading to the divergence of the filter. The basic idea of model error correction is to change the value of the elements in the covariance matrix to affect the size of the elements of the Kalman gain matrix, and then reasonably assign the weights of the predicted value and measured value of the state quantity. Here, the attenuation memory method is used to suppress the filter divergence.

The attenuation memory method filtering equations are shown in Equations (43)–(45). Compared with Equation (27), the filtering equation only has one more scalar factor λ in Equation (43). Since $\lambda > 1$, the prior state error covariance matrix $\widetilde{\boldsymbol{P}}_k'$ is always larger than $\boldsymbol{P}_k'$, so there is always a Kalman gain matrix $\boldsymbol{K}' > \boldsymbol{K}$. This shows that the use of the attenuation filter algorithm for the new measurement value is more weighted than the Kalman filter, which will cause the filter gain matrix to converge to a larger value, thus reducing the impact of model error. The larger the value of the scalar factor λ is, the higher the trust in the latest measurement value and the lower the trust in the historical measurement value. Through actual debugging, it is determined that $\lambda = 1.02$.

$$\widetilde{\boldsymbol{P}}_k' = \lambda^2 \boldsymbol{F} \widetilde{\boldsymbol{P}}_{k-1} \boldsymbol{F}^{\mathrm{T}} + \boldsymbol{Q} \tag{43}$$

where λ is the scalar factor, $\lambda > 1$.

$$\boldsymbol{K}' = \widetilde{\boldsymbol{P}}_k' \boldsymbol{H}^{\mathrm{T}} \left(\boldsymbol{H} \widetilde{\boldsymbol{P}}_k' \boldsymbol{H}^{\mathrm{T}} + \boldsymbol{R} \right)^{-1} \tag{44}$$

$$\widetilde{\boldsymbol{P}}_k = \widetilde{\boldsymbol{P}}_k' - \boldsymbol{K}' \boldsymbol{H} \widetilde{\boldsymbol{P}}_k' \tag{45}$$

4. Flight Test Verification

4.1. Verification Experiment of Attitude Solution

In order to verify the actual effect of the complementary filtering algorithm, the flight control module is installed at the center of mass of the "Phoenix" flapping wing robot, as shown in Figure 13. The flight control module also integrates the industrial-grade attitude calculation module Xsens-MTi3 and the MPU6050 chip at the same time. Among them, the MPU6050 is a traditional motion tracking chip, which is packaged with a three-axis gyroscope, a three-axis accelerometer and a DMP (Motion Digital Processor, DMP). The performance comparison between the chip and the Xsens-MTi3 module is shown in Table 2. It can be seen that the Xsens-MTi3 module has higher attitude estimation capability and attitude calculation accuracy, but it is expensive. Therefore, from the perspective of cost, MPU6050 is more suitable for attitude estimation of the "Phoenix" flapping wing robot.

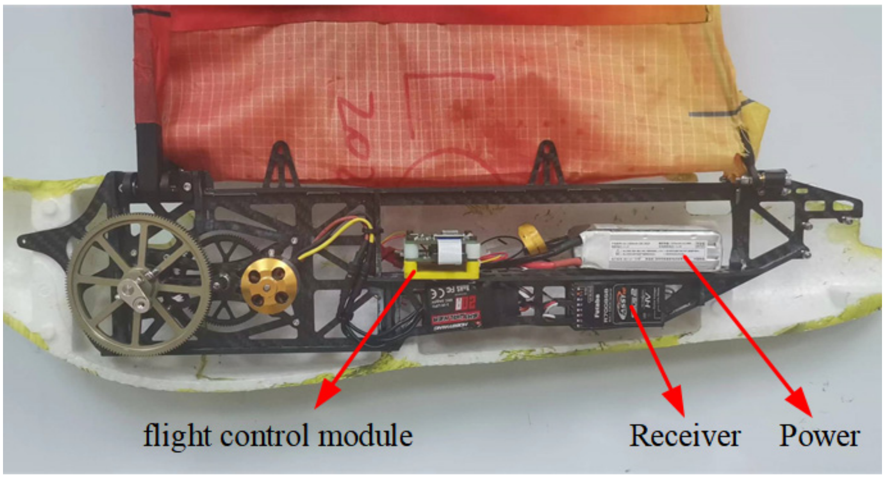


Figure 13. Installation diagram of flight control module.

Table 2. Comparison between MPU6050 and Xsens-MTi3.

	MPU6050	Xsens-MTi3
size	$4 \times 4 \times 0.9$ mm	$12 \times 12 \times 2$ mm
weight	<1 g	8 g
Calculating power	200 Hz	1 kHz
accuracy	Ø 10°	<1°
cost	8~12¥	1000~9000¥

After the flight control module is installed, the flapping wing robot is controlled by the remote controller to fly for a period of time. The attitude information solved by the display complementary filtering algorithm and the attitude data output by Xsens-MTi3 are recorded by the SD (Secure Digital, SD) card. The comparison of attitude changes over time is shown in Figure 14.

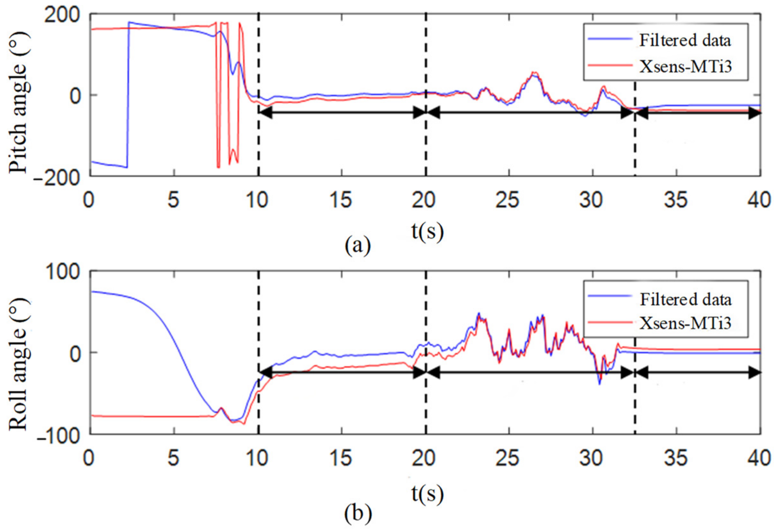
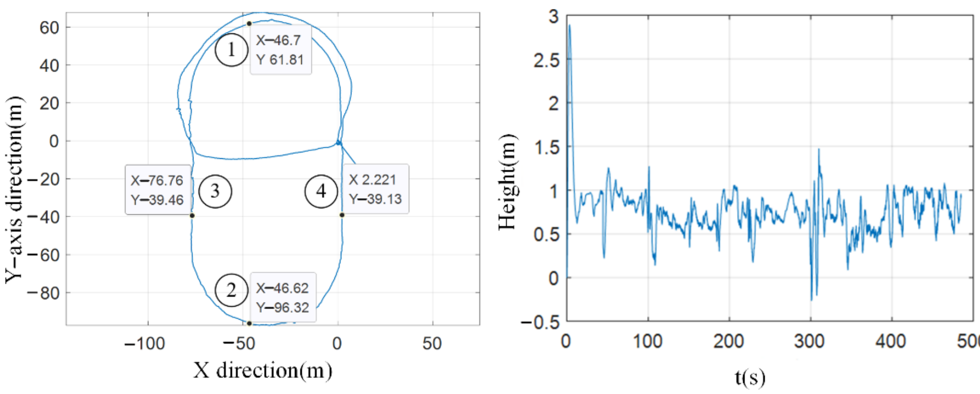


Figure 14. Comparison diagram of attitude solution. (**a**) Comparison of pitch angle changes. (**b**) Comparison of rolling angle changes.

It can be seen that the initial attitude solution by complementary filtering gradually converges with time. During the flight process of the "Phoenix" flapping wing robot, the results of the display complementary filtering algorithm and the output of Xsens-MTi3 are basically consistent, and the attitude estimation is always convergent during the whole flight process. The results show that the attitude calculation accuracy is less than 5 degrees, which basically meets the stability and rapidity indicators of the system design.

4.2. Verification Experiment of Position Solution

The flight test was conducted in an open sports field with no obstructions and strong magnetic field interference. The SD card was used to record the flight trajectory of the "Phoenix" flapping wing robot, and the accuracy and stability of the position calculation algorithm were evaluated. First of all, in order to verify the accuracy of the position calculation, let the "Phoenix" flapping robot flap at a certain frequency, making a circle around the middle of the sports field (the length of the middle ring of the sports field is about 158 m and the width is about 78 m). Then, walk around the outside of the sports field for half a circle, and the obtained plane trajectory is shown in Figure 15 (left). From the figure, the northward distance between points 1 and 2 is 158.13 m, and the eastward distance between points 3 and 4 is 78.98 m. It can be seen that the estimation accuracy of the horizontal position is less than 1.5 m, which meets the relevant requirements. The variation of height with time is shown in Figure 15 (right). Taking the height at the beginning of the detour as the benchmark, it can be seen that the fluctuation of the vertical position is within 1 m, which also meets the requirements.



(**a**) (**b**)

Figure 15. Plane track of detour playground and variation of height. (**a**) Plane track of detour playground. (**b**) Variation of height.

Then, the "Phoenix" flapping wing robot is controlled by the remote control to fly over the playground for a period of time, and the position and speed information during the flight are recorded. The variation of speed with time during flight is shown in Figure 16. It can be seen that the maximum horizontal resultant velocity speed of the "Phoenix" flapping wing robot is about 10 m/s, and the average horizontal resultant velocity speed is 7 m/s, which has strong maneuverability. In the vertical direction, due to the reciprocating vibration of the fuselage, the speed changes very drastically. The maximum climbing speed is about 2.5 m/s, and the maximum descending speed is about 5 m/s.

Based on the above two experiments, it can be seen that in the whole flight process, the Kalman filter algorithm does not have any divergence in the estimation of the horizontal position, and the stability is strong. The estimation of the flying height of the flapping wing robot tends to be stable on the whole. the Kalman filter algorithm is used to solve the position and speed of the "Phoenix" flapping wing robot, which meets the requirements of accuracy and stability, and has received better results.

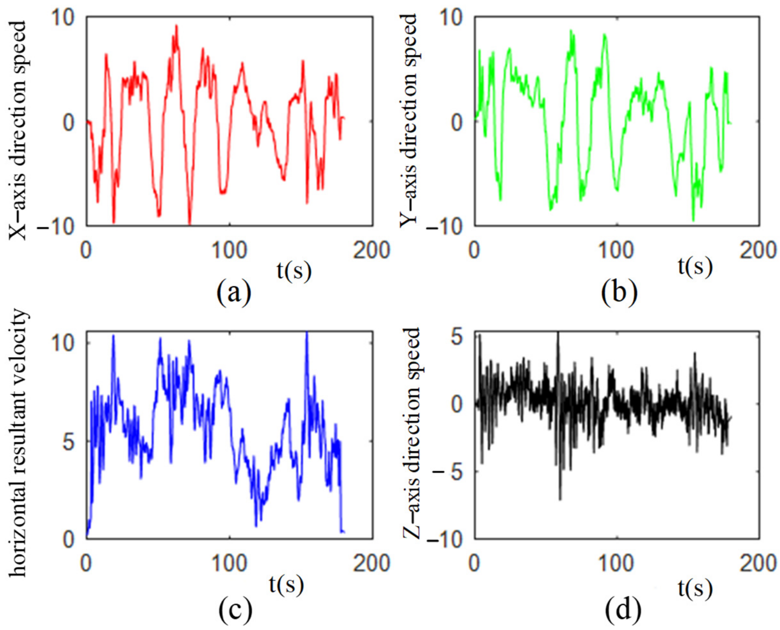


Figure 16. Flight speed variation diagram. (**a**) Speed variation in X direction. (**b**) Speed variation in Y direction. (**c**) Speed variation of horizontal resultant speed. (**d**) Speed variation in Z direction.

5. Conclusions

In view of the large mass ratio of the wings of the large bionic flapping wing flying robot and the fluctuation of the fuselage centroid during the flight, the traditional state perception and estimation methods struggle to provide stable and accurate data. This paper takes the "Phoenix" large-scale bionic flapping wing flying robot with a wingspan of 2.2 m as the research object, studies its integrated navigation method of attitude and position, and develops a corresponding flight controller to carry out flight experiment verification. A three-axis attitude estimation method for flapping wing flying robot based on the fusion of inertial sensor and magnetometer data is proposed, which solves the problem of attitude divergence caused by periodic heaving and pitching motions. The developed explicit complementary filtering algorithm fuses the IMU and magnetometer data to solve the attitude information of the flapping wing robot. The results obtained by the explicit complementary filtering algorithm are compared with the high-precision attitude module Xsens-MTi3. The results show that the attitude calculation accuracy is less than 5 degrees, which indicates that the use of low-cost sensors has achieved high-precision position and attitude estimation.

The position and velocity of Phoenix flapping wing robot in three-dimensional space are solved by using integrated navigation technology and the Kalman filter to fuse inertial navigation, GPS and barometer data. The Kalman filter is optimized by using the filtering algorithm, which improves the robustness of the Kalman filter. Aiming at the problem of filter divergence, the attenuation memory method is used to correct the model error, which improves the accuracy of position estimation. Finally, a complete position calculation process is planned and flight experiments are carried out. The results show that the accuracy and stability of the position solution meet the requirements when the sensor signal is good. At present, this method is mainly studied from the perspective of multi rigid bodies, and satisfactory results are obtained. In the future, the flexible vibration characteristics and fluid structure coupling characteristics of the wing will be considered, and on this basis, the method will be further optimized to improve the accuracy of state perception and estimation, ensuring real-time performance.

Author Contributions: G.L. contributed the central idea and wrote the initial draft of the paper. S.W., and W.X. contributed to refining the ideas, and revised this paper. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the Shenzhen excellent scientific and technological innovation talents training project (basic research on outstanding young people). Research on flight dynamics and cooperative control of large Bionic Flapping Wing robot formation, No. RCJC2020071414436040.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Ajanic, E.; Feroskhan, M.; Mintchev, S.; Noca, F.; Floreano, D. Bioinspired wing and tail morphing extends drone flight capabilities. *Sci. Robot.* **2020**, *5*, eabc2897. [CrossRef] [PubMed]
2. Lau, G.A. Stunt flying hawk-inspired drone. *Sci. Robot.* **2020**, *5*, eabe8379. [CrossRef] [PubMed]
3. Wang, S.P.; Du, C.P.; Zheng, Y. Path planning algorithm of flapping wing aircraft based on reinforcement learning. *J. Control Decis.* **2022**, *37*, 851–860.
4. Wang, Y.P. *Research on Autonomous Formation Flight Control Method of Large Bionic Flapping Wing Flying Robot*; Harbin University of Technology: Harbin, China, 2021; pp. 12–23.
5. Li, G.; Chen, X.; Zhou, F.; Liang, Y.; Xiao, Y.; Cao, X.; Zhang, Z.; Zhang, M.; Wu, B.; Yin, S.; et al. Self-powered soft robot in the mariana trench. *Nature* **2021**, *591*, 66–71. [CrossRef] [PubMed]
6. Di Luca, M.; Mintchev, S.; Su, Y.; Shaw, E.; Breuer, K. A bioinspired separated flow wing provides turbulence resilience and aerodynamic efficiency for miniature drones. *Sci. Robot.* **2020**, *5*, eaay8533. [CrossRef]
7. Xuan, J.L.; Song, B.F.; Song, W.P.; Yang, W.Q.; Xue, D.; Liang, S.R. Progress of chinese dove and future studies on flight mechanism of birds and application system. *Trans. Nanjing Univ. Aeronaut. Astronaut.* **2020**, *37*, 663–675.
8. Ramezani, A.; Chung, S.J.; Hutchinson, S. A biomimetic robotic platform to study flight specializations of bats. *Sci. Robot.* **2017**, *2*, eaal2505. [CrossRef] [PubMed]
9. Kamankesh, Z.; Banazadeh, A. Stability analysis for design improvement of bio-inspired flapping wings by energy method. *Aerosp. Sci. Technol.* **2021**, *111*, 106558. [CrossRef]
10. Yu, B.; Cheng, Y.Y.; Jin, X.Z.; Du, H.B. Robust finite time attitude stabilization of rigid spacecraft with angular velocity constraints. *Control. Decis.* 2021; *in press*.
11. Verboom, J.L.; Tijmons, S.; Wagter, C.D.; Remes, B.; Babuska, R.; de Croon, G.C.H.E. Attitude and altitude estimation and control on board a flapping wing micro air vehicle. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 5846–5851.
12. Zhu, X.; Zhao, T.; Cheng, D.; Zhou, Z. A three-step calibration method for tri-axial field sensors in a 3D magnetic digital compass. *Meas. Sci. Technol.* **2017**, *28*, 055106. [CrossRef]
13. Yang, W.; Wang, L.; Song, B. Dove: A biomimetic flapping-wing micro air vehicle. *Int. J. Micro Air Veh.* **2018**, *10*, 70–84. [CrossRef]
14. Tu, Z.; Yang, Y.; Fei, F.; Zhang, J.; Deng, X. Realtime on-board attitude estimation of high-frequency flapping wing mavs under large instantaneous oscillation. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 6806–6811.
15. He, W.; Mu, X.; Zhang, L.; Zou, Y. Modeling and trajectory tracking control for flapping-wing micro aerial vehicles. *IEEE/CAA J. Autom. Sin.* **2021**, *8*, 148–156. [CrossRef]
16. He, W.; Ding, S.Q.; Sun, C.Y. Research progress on modeling and control of flapping wing aircraft. *Acta Autom. Sin.* **2017**, *43*, 685–696.
17. Taha, H.E.; Kiani, M.; Hedrick, T.L.; Greeter, J.S. Vibrational control: A hidden stabilization mechanism in insect flight. *Sci. Robot.* **2020**, *5*, eabb1502. [CrossRef] [PubMed]
18. Bialy, B.J.; Chakraborty, I.; Cekic, S.C.; Dixon, W.E. Adaptive boundary control of store induced oscillations in a flexible aircraft wing. *Automatica* **2016**, *70*, 230–238. [CrossRef]
19. Dou, L.Q.; Mao, Q.; Su, P.H. Reentry attitude control of reusable launch vehicle based on adaptive fuzzy h∞ control. *J. Control Decis.* **2018**, *33*, 1181–1189.
20. Zhang, J.; Cheng, B.; Yao, B.; Deng, X. Adaptive robust wing trajectory control and force generation of flapping wing MAV. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 5825–5857.
21. Jafferis, N.T.; Helbling, E.F.; Karpelson, M.; Wood, R.J. Untethered flight of an Insect-sized flapping-wing microscale aerial vehicle. *Nature* **2019**, *570*, 491–495. [CrossRef] [PubMed]

22. Karasek, M.; Percin, M.; Cunis, T.; van Oudheusden, B.W.; De Wagter, C.; Remes, B.D.; de Croon, G.C. Accurate position control of a flapping-wing robot enabling free-flight flow visualisation in a wind tunnel. *Int. J. Micro Air Veh.* **2019**, *11*, 1756829319833683. [CrossRef]
23. Bailing, T.; Pinpin, L.; Hanchen, L.; Qun, Z. Trajectory and attitude cooperative control of multiple uavs in complex environment. *Acta Aeronaut. Et Astronaut. Sin.* **2020**, *41*, 36–43.
24. Wang, Z.W.; Wang, F.J.; Di, C.C.; Shi, Z.; Yang, G. Nonlinear alignment method of land inertial navigation system. *Acta Aeronaut. Et Astronaut. Sin.* **2018**, *39*, 283–293.
25. Wu, H.; Pei, X.; Li, J.; Gao, H.; Bai, Y. An improved magnetometer calibration and compensation method based on Levenberg–Marquardt algorithm for multi-rotor unmanned aerial vehicle. *Meas. Control.* **2020**, *53*, 276–286. [CrossRef]
26. Zhao, G.L.; Xu, D.F.; Ge, L.Z.; Guo, Q.R. Research on calibration method of MEMS accelerometer based on kalman filter. *Transducer Microsyst. Technol.* **2021**, *40*, 25–27.
27. Zhang, Z.; Wang, J.B.; Song, B.; Tong, G.-f. Adaptive complementary filtering algorithm for imu based on mems. In Proceedings of the 2020 Chinese Control And Decision Conference (CCDC), Hefei, China, 22–24 August 2020; pp. 5409–5416.
28. Khamseh, H.B.; Ghorbani, S.; Janabi, F. Unscented kalman filter state estimation for manipulating unmanned aerial vehicles. *Aerosp. Sci. Technol.* **2019**, *92*, 446–463. [CrossRef]