Design and Implementation of Multiple-Rotorcraft-Flying-Robot Testbed
Zheng Wang, Feng Gu, Yuqing He, Jianda Han, and Yuechao Wang

Abstract—In this paper, a multiple-rotorcraft-flying-robot (MRFR) testbed is introduced. Firstly, physical structure, as well as hardware and software system, is described in detail. Subsequently, in order to realize the autonomous flight of each single rotorcraft-flying-robot (RFR), modeling, identification and flight control experiments are conducted. Then, formation flying demonstration is presented to verify the basic functions of the new-designed MRFR testbed. Finally, two points should be emphasized: 1) both cooperation/coordination algorithm of MRFR and advanced flight control algorithm of single RFR can be verified on this testbed; 2) experiments on this testbed can approach real flight test much better because three real RFR systems are used.

I. INTRODUCTION
Due to presenting some attracting high maneuverability such as hovering and emergency turning, etc., rotorcraft flying robot (RFR) has been expected to play important role in more and more applications, such as aiding disaster recovery efforts in mines and after earthquakes/tsunami. Simultaneously, it has been reached a common view that working efficiency of multiple rotorcraft flying robots (MRFR) through good cooperation and coordination is much higher than single RFR. Thus, most recently, many researchers are devoting themselves to the topic of MRFR.

However, with the rapid development of theoretical researches, for example, formation control[1], cooperative observation[2], consensus algorithms[3], etc., experimental tests have gradually been one of the most important bottleneck in the field of MRFR. Up to now, there are only few research groups having conducted experimental studies of MRFR, such as: 1) formation flight experiments in university of UC Berkeley using the BErkeley AeRobot (BEAR) Rotorcraft-based Unmanned Aerial Vehicles (RUAVs)[4]; 2) formation reconstruction and collision avoidance test of National University of Singapore using a group of UAV helicopters for verifying a collision avoidance formation control scheme[5].

The main difficulties of MRFR’s experimental tests are as following three aspects: 1) experiments of MRFR is very dangerous because high performance autonomous control of single RFR is still an open problem that does not be solved perfectly; 2) experiments of MRFR is extraordinarily complicated since too much factors, such as disturbances, unknown aero-dynamics parameters, should be considered beforehand; 3) experiments of MRFR is so time-consuming and expensive that few research groups can afford them.

Thus, several indoor experimental testbeds have been developed to solve these problems as far as possible. For example: the 3-DOF helicopter testbed in the Flight Systems and Control Laboratory in University of Toronto[6], and the flying testbed developed in UCLA[7]. However, these existing testbeds were mainly designed to test cooperation and coordination of different bodies and ignore most of locomotion characteristics of real RFR systems. Based on this, a new indoor MRFR testbed is developed in this paper. The most attracting advantage of this testbed is that it is composed of three passive three-degree-of-freedom arms driven by three real electric rotorcraft flying robots, which make it presents more characteristics of real RFR systems.

The remainder of this paper is organized as follows: First, in section II, the overall design scheme is presented. Second, hardware and software structures are introduced in section III. Subsequently, some experiments are conducted and results are analyzed to verify the validity of the new developed testbed in section IV, which is then followed by the conclusions.

II. BASIC DESIGN SCHEME

A. Mechanism Design
The new developed MRFR testbed is shown as Fig.1. It is mainly composed of a main vertical axis, three passive arms and three real electric rotorcraft flying robot fixed at the end of each arm. Each arm has three rotational joints called vertical, horizon and yaw, respectively, where horizon and yaw joint can both rotate continuously without any limitation, and vertical joint can only move vertically from -15° to 15°. Also, some weights are used to balance the additional weight of RFR, such as controller and sensors.

Embedded controller of every RFR is attached on the passive arm to drive the RFR and measure the sensor data. The main sensors are encoders equipped in every joint. Also, some other state and environment detection sensors can be also used in this testbed, such as accelerometer, camera, LIDAR and so on. A commutating brush is equipped at every rotational joint to prevent electric wire circling around the joint.
As a conclusion of this sub-section, we point out the basic features of this testbed:
1) All the joints are passive and driven by real RFRs;
2) Collision can be avoided by the physical limitation of the testbed, thus experimental risk can be relieved greatly;
3) The testbed is versatile because not only cooperation and coordination of MRF experiments but also advanced control algorithm of single RFR can be tested, as well as, some other experiments about RFR can be conducted by increasing some sensors, such as target tracking.

Fig.1. Sketch of the MRFR testbed (1. main vertical axis; 2. passive arm; 3. electric rotorcraft flying robot; 4. vertical joint; 5. horizon joint; 6. yaw joint; 7. balancing equipment; 8. encoders; 9. commutating brush; 10. controller)

B. Electrical Design

A two levels control structure as Fig.2 is used for the control system, including flight control level and ground control level.

The detailed functions of ground control level is as follows,
1) Monitor flight states of each RFR system;
2) Online regulate motion control parameters of RFRs;
3) Execute some cooperative/coordinate algorithm.

While the basic functions of flight control level are as follows,
1) Implement motion control law of each single RFR system;
2) Transmit state of each RFR system to the ground control computer.

III. HARDWARE SYSTEM

Except for the main body and RFRs, the MRFR testbed is composed of three sub-systems (as Fig.3): sensor system, flight control system and ground control system. In this section, we will introduce these three sub-systems in detail.

A. Sensor system

Three kinds of basic sensors are equipped in this testbed to measurement the states of the platform and RFR: encoder (measure position/velocity of each RFR) and two kinds of photoelectric switch (one is used to obtain reference position for horizon motion; another is for measuring rotor speed).

1) Encoders and joint position/velocity measurement. Since each arm has three joints, three increment encoders (Delta ES3-35CB6942, with the resolution of 0.144°) are assembled at rotation centers of the arm joint as shown in Fig.4, where Encoder-1, Encoder-2 and Encoder-3 are used for measuring vertical angle, horizon angle and yaw angle, respectively.

2) Photoelectric switch and zero reference position. To obtain absolute position in horizon, a photoelectric switch (Omron E3Z-G), called zero checking switch, is equipped on each horizon joint, as shown in Fig.4. Once the photoelectric switch is triggered for the first time, horizon position is
calibrated at initial position denoted as “0”.

Also, the output of each zero checking switch is dealt with by the Single Chip Micyoco system where the horizon state is calculated cooperating with horizon encoder.

With the position encoder and zero checking switch, as well as the Single Chip Micyoco system, we can obtain all required states of the platform.

3) Photoelectric and rotor speed. Stable rotor speed of main/tail rotor for a RFR system is important for its high performance control. In this testbed, a photoelectric switch (Omron E3Z-D61), called speed measurement switch, is applied for each RFR system to measure the rotor speed.

![Photoelectric switch for measuring rotor speed](image)

The speed measurement switch is installed in the tail of the RFR as shown in Fig.5. It can output pulse signals that is proportional to the rotation speed of the tail rotor and main rotor. The accuracy of rotor speed measurement is better by measuring the tail rotor because the tail rotor speed is faster than the main rotor and thus can be conducted faster. The experiments result shows the feasibility of this idea.

**B. Flight Control System (FCS)**

FCS (Fig.6) for each RFR system is implemented by a TI DSP 28xx series EVM which includes enhanced Pulse Width Modulator (EPWM), Serial Communication Interface (SCI) and Enhanced Quadrature Encoder Pulse (EQEP) etc. The Ti 2000 series DSP with 150MHz operating frequency has the adequate computing ability to realize flight control algorithm.

**C. Ground Control System (GCS)**

A personal computer (Intel Core TM i5 M450, 2.40 GHz) is used as a GCS for the whole testbed. An RS485 communication card (Advantech PCI-1612) is used for driving the RS485 serial bus to communicate with all the three FCSs.

**IV. SOFTWARE SYSTEM**

**A. Software structure of FCS**

The whole programming flow diagram of FCS is as Fig.7 and the basic sample time of FCS is 100ms.

![Program flow graph](image)

One of the most important functions of FCS is to realize the flight control of each single RFR system. Thus we will introduce the basic structure of flight controller in this sub-section. Flight control of each single RFR is shown as Fig.8 which is composed of three independent closed loops: rotor speed loop, vertical position loop and horizon position loop. In the earlier experiments, we do not use the rotation joint of yaw and thus the control structure does not include the control loop of yaw motion.

![Single RFR control structure](image)

Rotors speed loop is designed to keep a constant rotation speed of main rotor; vertical position and horizon velocity loop are used to regulate the movement of each arm. With respect to horizontal loop, a two-level control structure with speed inner loop and position outer loop is selected because tail rotor is the main driver for horizon motion but the force originated from it is small relative to the large inertia.

**B. Software structure of GCS**

GCS system, whose interface is shown as Fig. 9, is programmed using Visual C++ 6.0. The interfaces of GCS can be divided into three regions: control parameter regulation panel, desired formation regulation panel, and state monitoring panel. Flight control parameters of each RFR system can be regulated through parameter regulation panel. Desired formation can be input manually in the region of
desired formation and then transmitted to every FCS through GCS. Finally, all RFRs’ feedback states transmitted to the GCS are displayed in the state monitoring part.

Fig.9. Interface of Ground Control Station

Fig.10 is the program diagram of GCS. To avoid communication conflicts, communication between each single RFR system and GCS is implemented in order in every time interval (1s). This ordered send-receive communication threading is independent of the Windows message threading, so multithreading program is designed to guarantee the continuous.

V. EXPERIMENT

A. Single RFR System Identification

Since advanced control strategies depend on system dynamical model, we firstly conduct experiments to obtain each single RFR system parameters through off-line identification algorithms. Firstly, it is not difficult for us to produce a linear model structure (as Eq.1) by simplified system model of a real RFR system,

\[
\begin{align*}
\dot{\theta} &= q \\
\dot{\varphi} &= r \\
\dot{q} &= \frac{(T_{\alpha} + mgl \cos \theta)}{J_s} \\
\dot{r} &= \frac{r}{J_s}
\end{align*}
\]  

(1)

where \(\theta\) and \(\varphi\) are the vertical and horizon positions, \(q\) and \(r\) are the vertical and horizon velocity, \(T_{\alpha}\) and \(T_{r}\) are the main and tail rotor thrust, respectively. Because \(\theta\) is small \((\theta \in [-15, 15])\), \(\cos \theta\) in Eq.1 can be approached by using 1-\(\theta\), and thus we have

\[
\dot{q} = \frac{(T_{\alpha} + mgl \theta - mgJ_{\alpha})}{J_s}
\]  

(2)

Further, based on aero-dynamics of rotorcraft flying robot[8], \(T_{r}\) can be denoted as following linear equation

\[
T_{r} = \alpha X_{\alpha} + \beta X_{\alpha} + \gamma_{\alpha} u_{\alpha} + \gamma_{\alpha} u_{\alpha}
\]  

(3)

and

\[
\dot{\alpha}_{\alpha} = -\tau \alpha_{\alpha} + \kappa u_{\alpha}
\]  

(4)

where \(\tau\) and \(\kappa\) are delay coefficient and input coefficient, respectively. So each single RFR system model can be expressed as a 2-input 6-state linear equation,

\[
\dot{x} = Ax + Bu
\]  

(5)

where \(x = [\theta, \varphi, q, r, \alpha_{\alpha}, \alpha_{\alpha}, \alpha_{\alpha}]^T\), \(u = [u_{\alpha}, u_{\alpha}]^T\),

\[
A = \begin{bmatrix}
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\]

\[
B = \begin{bmatrix}
0 & 0 \\
0 & 0 \\
0 & 0 \\
0 & 0 \\
0 & 0 \\
0 & 0
\end{bmatrix}
\]

Using Matlab system identification toolbox, the unknown parameters can be easily estimated and the results are as following and Fig.11,

\[
A = \begin{bmatrix}
0.0016 & 0 & 0 & 0 & 0 & 0 \\
0.0016 & 0 & 0 & 0 & 0 & 0 \\
0.0052 & 0 & 0 & 0 & 0 & 0 \\
0.0015 & 0 & 0 & 0 & 0 & 0 \\
0.0015 & 0 & 0 & 0 & 0 & 0 \\
0.0015 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\]

The average model fitness is

\[
\text{fitness} = 100\% \times \left(1 - \frac{y_{\text{sim}} - y_{\text{real}}}{y_{\text{sim}} - y_{\text{real}}} \right) = 91.7\%
\]  

(6)

That means the estimated system equation approached real system’s behavior perfectly.

Since linear model (5) is not accurate enough to describe the RFR dynamic characters, especially for the flight mode of high maneuverability, identification and estimation algorithms of nonlinear dynamic model can be tested and compared on the new testbed in the future.
B. Single RFR Linear Control

With the linear system equation (5) obtained in section V-A, we can design linear controller of each single RFR system through linear system theory. And the experimental results of step response are as shown in Fig.11, where the desired value of rotor speed, vertical position and horizon velocity are respective 160rpm, 0°, and 10°/s (red and dash line). From Fig. 12, it is clear that each RFR system can be stabilized and the regulation time is about 5~10s.

If the yaw joint is used, each single RFR system will be high nonlinear, coupling and under-actuated, that means the new plant also apply a good testbed to test test and compare different nonlinear control algorithms which can be hardly test on outdoor RFR.

C. Multiple RFR Formation Control

In order to present the feasibility and validity of testing coordination/cooperation algorithm using the new developed testbed, we will list some formation control experimental results in this sub-section. The formation control structure is as Fig.13 where the red blocks are flight controller described in Fig.8. Here we use a often-referred-to leader-follower formation control structure and linear formation control algorithm using relative kinematics[9]. In our experiment, only the horizon relative position controllers are designed to present a simply formation problem. Taking the middle RFR as the leader and the upper and lower RFR as followers (Fig.14), the horizon relative motion can be controlled by the relative kinematics method as follow,

\[ u_j = -u_{\gamma_d} + u_l - k_r (p_{r_{\gamma}} - p_l) \]  

(7)

where \( p_{r_{\gamma}} \) and \( u_{\gamma_d} \) are desired relative horizon position and velocity; \( p_{\gamma}=p_{l}-p_l \) is the measured relative horizon position; \( p_l \) and \( p_f \) are the measured horizon position of leader and follower RFR; and \( u_l \) is the measured horizon velocity of leader RFR. \( k_r \) is the linear control parameter to be designed for follower RFR.

Formation procedure, as shown in Fig.14, is divided into five processes where the desired states, leader velocity \( (n_{l,\gamma}) \) and follower relative position \( (p_{r1,\gamma} \) and \( p_{r2,\gamma}) \), are represented as follow.
\( v_l = 0^\circ / s, p_{1l} = 90^\circ, p_{2l} = -90^\circ \quad t \in (0, 30) \)
\( v_l = 0^\circ / s, p_{1l} = 0^\circ, p_{2l} = 0^\circ \quad t \in [30, 60) \)
\( v_l = 10^\circ / s, p_{1l} = 0^\circ, p_{2l} = 0^\circ \quad t \in (60, 120) \)
\( v_l = 0^\circ / s, p_{1l} = 0^\circ, p_{2l} = 45^\circ \quad t \in [120, 180) \)
\( v_l = 10^\circ / s, p_{1l} = 0^\circ, p_{2l} = 45^\circ \quad t \in [180, 210) \)

The experiment results are shown as Fig.15 where the red lines describe the desired states in GCS and the blue lines denote measured states. From Fig. 14(b), it can be clearly seen that formation can be kept perfectly and the feasibility of formation control test is shown.

Fig.14. Experiment of formation control

![Experiment of formation control](image)

The experiment results are shown as Fig.15 where the red lines describe the desired states in GCS and the blue lines denote measured states. From Fig. 14(b), it can be clearly seen that formation can be kept perfectly and the feasibility of formation control test is shown.

Fig.15. Experiment result of formation control

VI. CONCLUSION

An extensible MRFR testbed, which can be used to test both single RFR high performance flight control algorithm and cooperation/coordination algorithm of MRFR, is introduced in this paper. The hardware sub-system, the software-sub-system and the control structure are presented in detail. Also, experiments results, including system identification, flight control of single RFR, and formation control of MRFR, showed the feasibility and validity of the new developed platform.

REFERENCES

[1] Z. Wang, Y.Q. He and J.D. Han, “Multi-Unmanned Helicopter Formation Control on Relative Dynamics,” in *Proc. IEEE Mechatronics and Automation*, Changchun, 2009, pp. 4381-4386.

[2] F. Gu, Y.Q. He, J.D. Han and Y.C. Wang, “ESMF based multiple UAVs active cooperative observation method in relative velocity coordinates,” in *Proc. IEEE Decision and Control*, 2009, Shanghai, pp. 3008-3013.

[3] J.A. Fax, and R.M. Murray, “Information flow and cooperative control of vehicle formations,” *IEEE Trans. Automation and Control*, vol 49(9), pp. 1465-1476, 2004.

[4] E. Shaw, H. Chung, J. Hedrick and S. Sastry, “Unmanned helicopter formation flight experiment for the study of mesh stability,” *Lecture Notes in Economics and Mathematical System*, vol 588, pp. 37-56, 2007.

[5] B. Yun, B.M. Chen, K.Y. Lum and H. Lee, “Design and implementation of a leader-follower cooperative control system for unmanned helicopters,” *Journal of Control Theory and Applications*, vol 8(1), pp. 61-68, 2010.

[6] S. Sarpalli, G. S. Sukhatme and J. F. Montgomery, “An Experimental Study of the Autonomous Helicopter Landing Problem,” *Springer Tracts in Advanced Robotics*, vol 5, pp. 466-475, 2003.

[7] J. Shan, H. Liu, “Development of an experimental testbed for multiple vehicles formation flight control,” in *Proc. IEEE Control Applications*, Toronto, 2005, pp. 160-164.

[8] M. Béjar, A. Ollero and F. Cuesta, “Modeling and control of autonomous helicopters,” *Journal of Advances in Control Theory and Applications*, vol 353, pp. 1-29, 2007.

[9] A. K. Das, R. Fierro, V. K. Kumar, J. P. Ostrowski, J. Spletzer and C. J. Taylor, “A vision-based formation control framework,” *IEEE Trans. Robotics and Automation*, vol. 18(5), pp. 813-825, 2002.