Mechatronics 19 (2009) 1057–1066

Contents lists available at ScienceDirect

Mechatronics

journal homepage: www.elsevier.com/locate/mechatronics

Design and implementation of a hardware-in-the-loop simulation system for small-scale UAV helicopters ☆

Guowei Cai, Ben M. Chen *, Tong H. Lee, Miaobo Dong

Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117576, Singapore

ARTICLE INFO

Keywords:
Hardware-in-the-loop simulation
UAV
Helicopter
Flight control

ABSTRACT

We present in the paper the design of a hardware-in-the-loop simulation framework and its actual implementation on our custom constructed unmanned-aerial-vehicle (UAV) helicopter systems. Real-time hardware-in-the-loop simulation is one of the most effective methods for the verification of the overall control performance and safety of the UAVs before conducting actual flight tests. In our proposed framework, four modules, which include onboard hardware, flight control, ground station and software, are integrated together to realize the hardware-in-the-loop simulation. This design is successfully utilized for simulating several flight tests including basic flight motions, full-envelope flight and multiple UAV formation flight. Results obtained show that the constructed hardware-in-the-loop simulation system is highly effective and useful.

1. Introduction

Unmanned aerial vehicles (UAVs) have recently aroused great interest and attention in the academic circle worldwide because of their great potential in both military and civil applications. Many research groups have constructed their own UAV platforms for various research purposes (see, for example [1,5,7,11–13]). Generally, the construction of UAVs is costly and time consuming. Safety is thus a primary issue that one is facing in conducting actual flight tests. As such, intensive testing and simulation, especially with the actual UAV hardware in the simulation loop, is an effective way to detect and prevent unnecessary malfunctions of hardware, software and automatic flight control systems. Hardware-in-the-loop simulation is a real-time simulation method or framework, in which the UAV platform is reacting the same way as it in the real experiment. Using such a method, researchers can effectively evaluate the reliability of the overall UAV system. In particular, the framework can be intensively used to examine the performance of designed automatic flight control algorithms. Necessary enhancements and modifications can then be done before actually testing the UAV system in the sky and thus the probability of test flight accidents can be greatly minimized.

We have recently constructed two sets of small-scale UAV helicopters, HeLion and SheLion, shown in Fig. 1, respectively. The UAVs are utilized for implementing newly developed automatic control techniques, and missions such as ground target detecting and tracking as well as multi-UAV cooperation. Both UAV helicopters are upgraded from radio-controlled hobby helicopter, Raptor 90, by equipping with a custom designed onboard avionics system [2] with a comprehensive onboard and ground station software system [6]. For the sake of safety and reliability evaluation, we propose in this work a hardware-in-the-loop simulation framework for our UAV platforms by integrating the developed hardware and software systems together with the dynamic model of the UAV helicopter. This framework includes the following four modules (1) onboard hardware module; (2) flight control module; (3) ground station module and (4) software module. Various hardware-in-the-loop experiments conducted show that the framework is effective and instrumental to real flight tests. For certain flight tests, the result of the hardware-in-the-loop simulation is able to provide safety alerts to human pilots in advance. As a result, the human pilot can overtake the automatic system and perform a timely manual control when the UAV helicopter become unstable.

The outline of this paper is as follows: In Section 2, we present the complete design procedure of our proposed hardware-in-the-loop simulation framework. All of the modules in the framework will be introduced in detail. Several simulation results, including those for basic flight motions, full-envelope flight and formation flight, are illustrated in Section 3 to verify the proposed simulation framework. Finally, in Section 4, we draw some concluding remarks.

2. Design of hardware-in-the-loop simulation framework

The framework of hardware-in-the-loop simulation is depicted in Fig. 2. This framework includes: (a) an onboard hardware

☆ This work is supported in part by Defence Science and Technology Agency (DSTA) of Singapore under a Temasek Young Investigator Award.

* Corresponding author. Tel.: +65 6516 2289; fax: +65 6779 1103.
E-mail addresses: g0301341@nus.edu.sg (G. Cai), bmchen@nus.edu.sg (B.M. Chen), eleleeth@nus.edu.sg (T.H. Lee), eledm@nus.edu.sg (M. Dong).

0957-4158/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.mechatronics.2009.06.001

Fig. 1. The UAV helicopters, HeLion and SheLion.

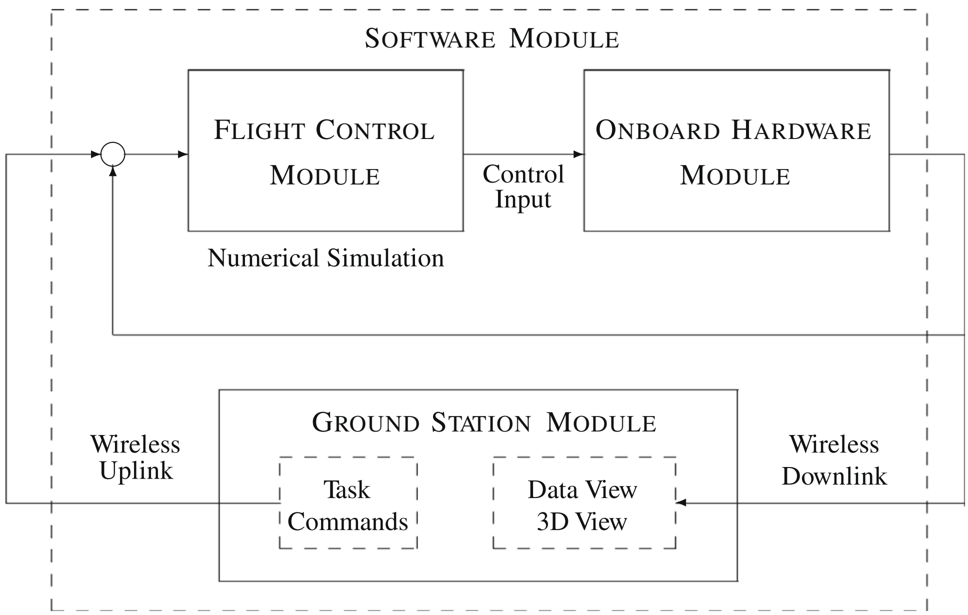

Fig. 2. Framework of the hardware-in-the-loop simulation of the UAV system.

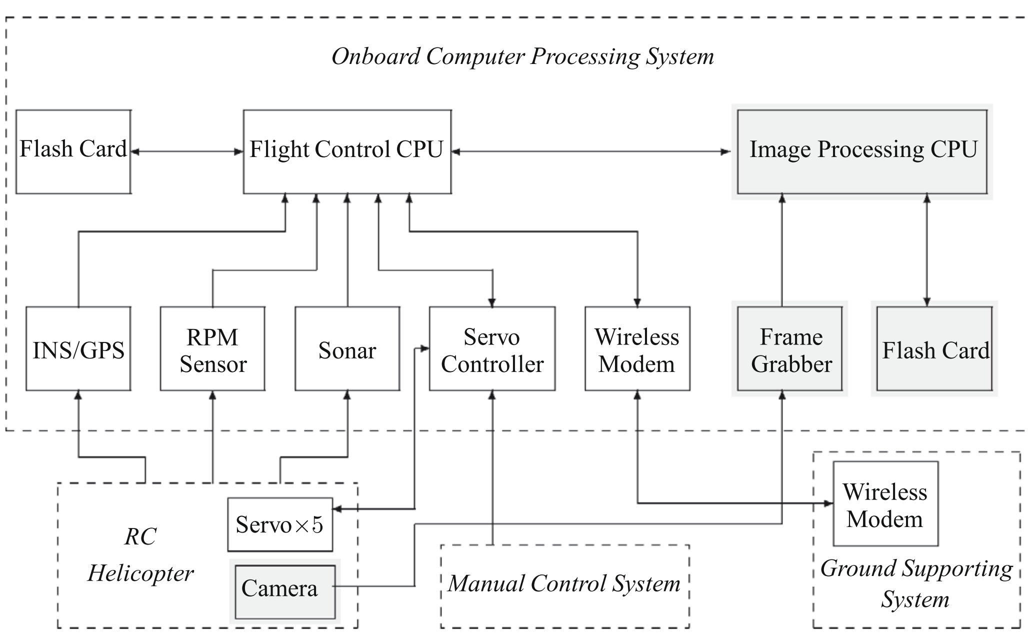

Fig. 3. Hardware configuration of the UAV helicopter system, SheLion.

module that activates helicopter servo actuators and output sensors; (b) a flight control module for executing automatic control algorithms; (c) a ground station module for generating task commands and monitoring the helicopter through data view and 3D view interfaces; and (d) a software module for integrating all the previous three modules to perform real-time hardware-in-the-loop simulation. In what follows, we proceed to give a detailed description of all these modules.

2.1. Onboard hardware module

Shown in Fig. 3 is the overall hardware configuration of one of our UAVs, SheLion. The onboard hardware module consists of the following three main components: (1) an onboard computer processing system; (2) several servo actuators of helicopter; and (3) a wireless modem. It is noted that the modules related to image processing (marked in grey in Fig. 3), i.e., the image processing CPU, the frame grabber, the flash card and the camera, are not included in the hardware-in-the-loop simulation scheme. In our proposed framework, flight control execution is done by the onboard computer system. Such an arrangement distinguishes from some traditional hardware-in-the-loop simulation designs, in which the simulation of control algorithms is performed on another flight simulation computer. This separation, to some degree, is caused by insufficient computational capability of selected onboard processing computer systems. For example, in simulation systems reported in [8] and [13], the onboard computer processing units are running at a very low frequency (less than 300 MHz) and thus cannot afford the computational load of simulating control algorithms with higher order helicopter dynamic model. The processing units used on our UAV systems, HeLion and SheLion, have a working frequency higher than 600 MHz, which enables us to execute flight control algorithms onboard directly as they are done in actual flight tests. Such a design completely prevents another loop involving an additional flight simulation computer, and hence reduces unnecessary work required for software programming and hardware adjustment.

The second hardware component is the onboard servo actuators, which are used to drive the helicopter to realize roll, pitch, yaw and heave motions. In the hardware-in-the-loop simulation, observing and measuring the responses, such as moving directions

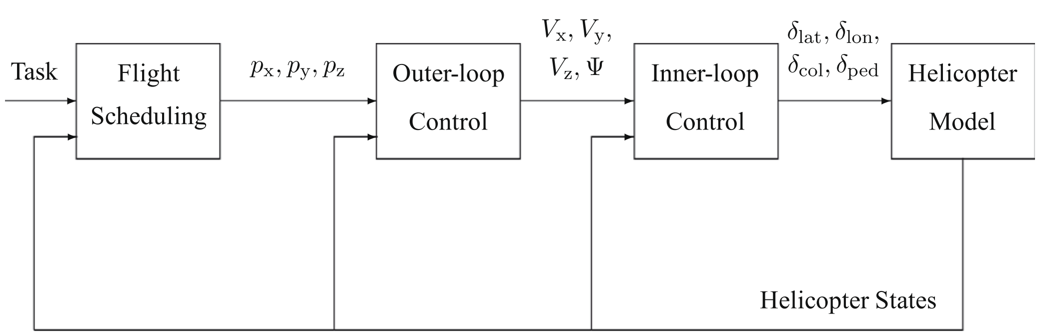

Fig. 4. Automatic flight control structure of the UAV helicopter system.

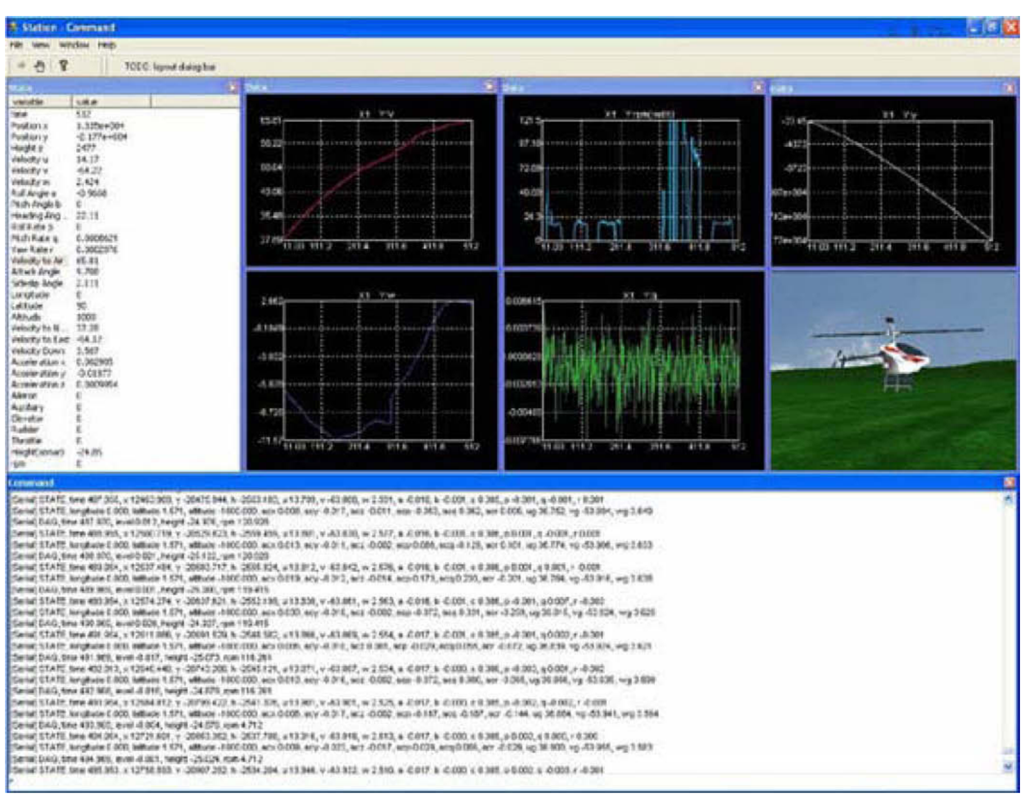

Fig. 5. Data view interface of the ground station system.

and amplitudes, of the servo actuators are an effective way to evaluate the performance of designed automatic control algorithms.

The last main hardware component included in the hardware-in-the-loop simulation is a pair of wireless modems, with one being installed on the helicopter and the other being attached to the ground station, which provide a full-duplex wireless communication channel for the downlink and uplink of data between the onboard system and the ground station.

2.2. Flight control module

The flight control module is for implementing automatic control algorithms. The structure of this module is shown in Fig. 4, which consists of three hierarchical control loops, i.e., (1) an inner-loop control block, which is to internally stabilize the helicopter dynamics; (2) an outer-loop control block, which is to control the position of the helicopter; and lastly, (3) a flight scheduling block, which is to navigate the UAV to follow flight paths generated by the ground station.

More specifically, the inner-loop control block is to track the reference signals of velocities and heading angle of the helicopter along axes of its body frame, i.e., V_x, V_y, V_z and Ψ, and to guarantee the internal stability of the UAV helicopter. The output signals of the inner-loop control are δ_{lat}, δ_{lon}, δ_{col} and δ_{ped}, which are used to respectively drive the aileron, elevator, collective pitch and rudder servo actuators of the helicopter.

Fig. 6. 3D view interface of the ground station system.

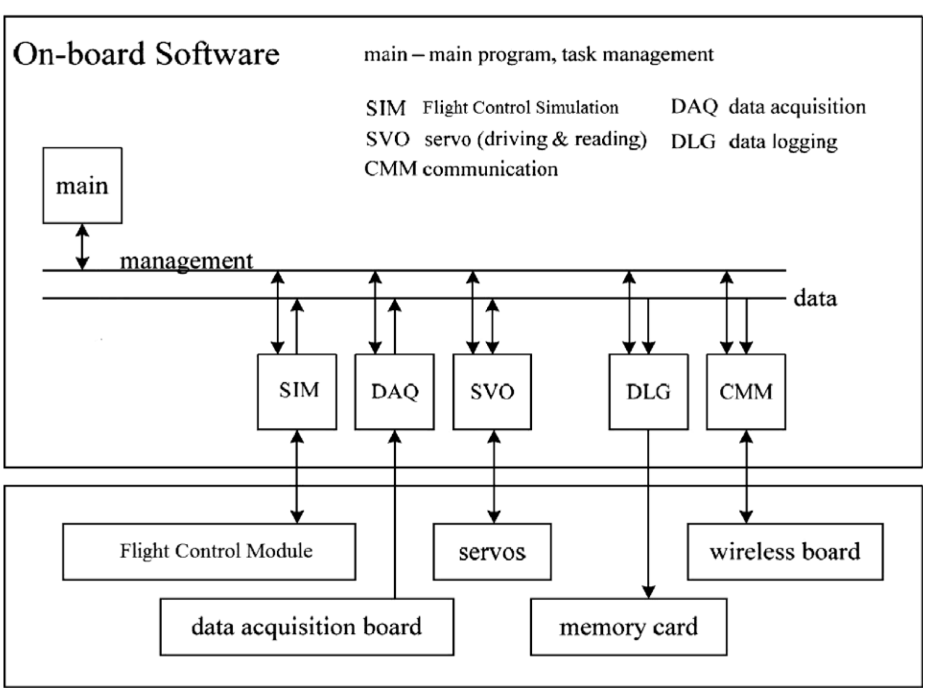

Fig. 7. The structure of the onboard software system.

The outer-loop control is to control the helicopter to appropriate position specified by p_x, p_y and p_z, respectively, in the ground frame, i.e., the commonly used NED (north-east-down) frame. Its output signals are used as references for the inner-loop control.

Lastly, the flight scheduling block is an organizer to schedule all necessary flight paths for designated missions or tasks for the UAV helicopter, which can either be predefined or generated online by the ground station.

2.3. Ground station module

The ground station module is for direct observation and monitoring purposes. In our hardware-in-the-loop simulation framework, the performance of the overall system including automatic flight control is evaluated on the ground station based on the downlink data of the onboard hardware and processing units through the wireless modems. Two friendly and easy-to-use user interfaces for data view are implemented, one is a normal data view as shown in Fig. 5 and the other is a 3D view depicted in Fig. 6. Another important function that the ground station is capable of performing is to generate real-time commands and tasks for the helicopter, if necessary.

2.4. Software module

The software module is to integrate all the previous three modules together. The module adopted in the hardware-in-the-loop

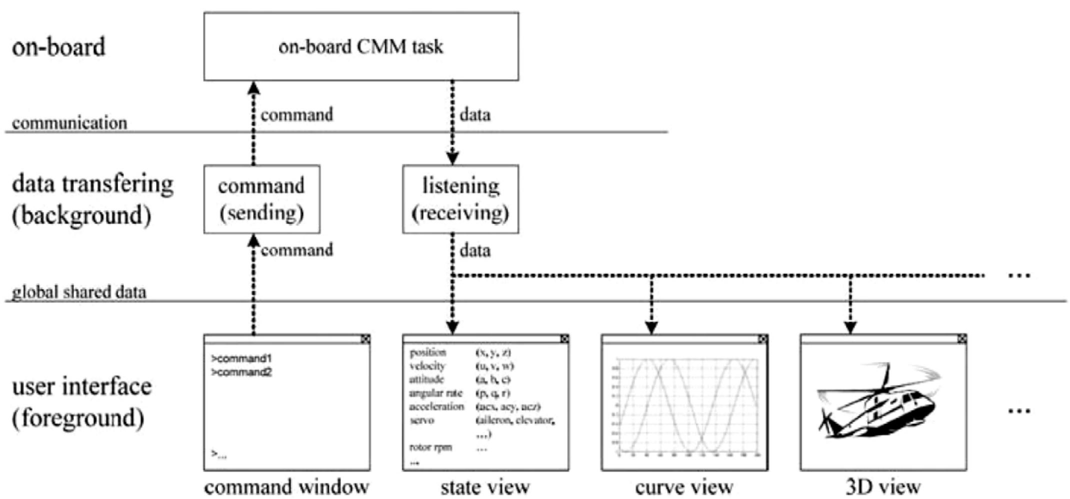

Fig. 8. The structure of the ground station software system.

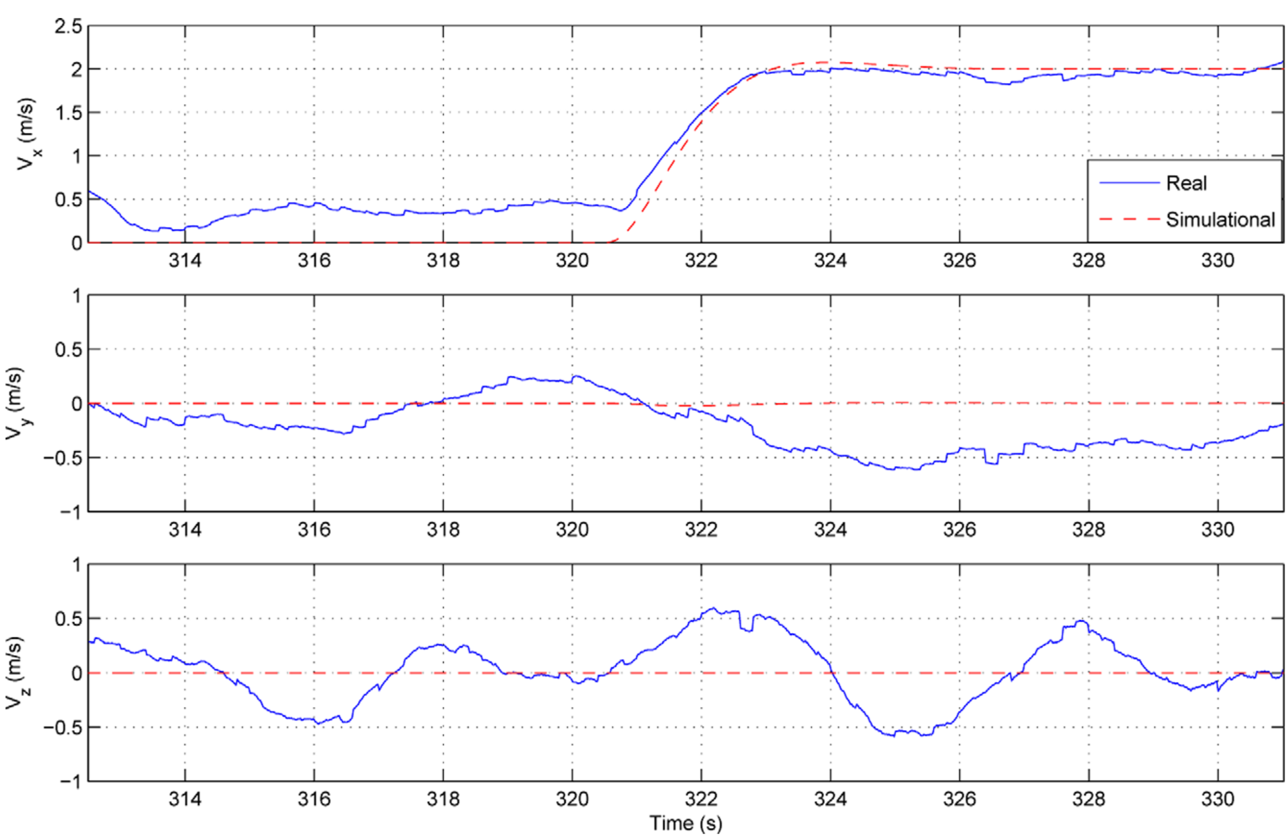

Fig. 9. Velocities of hardware-in-the-loop simulation and actual flight tests.

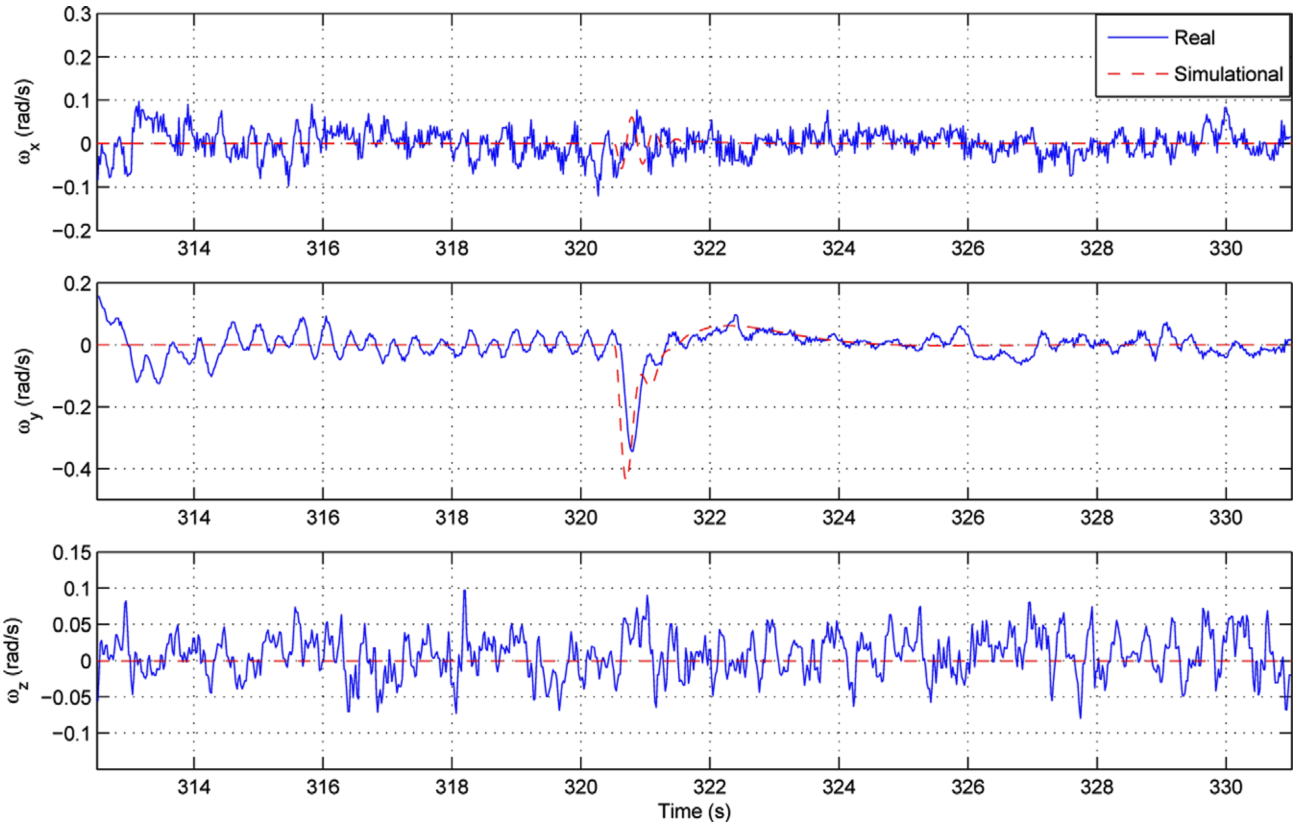

Fig. 10. Angular rates of hardware-in-the-loop simulation and actual flight tests.

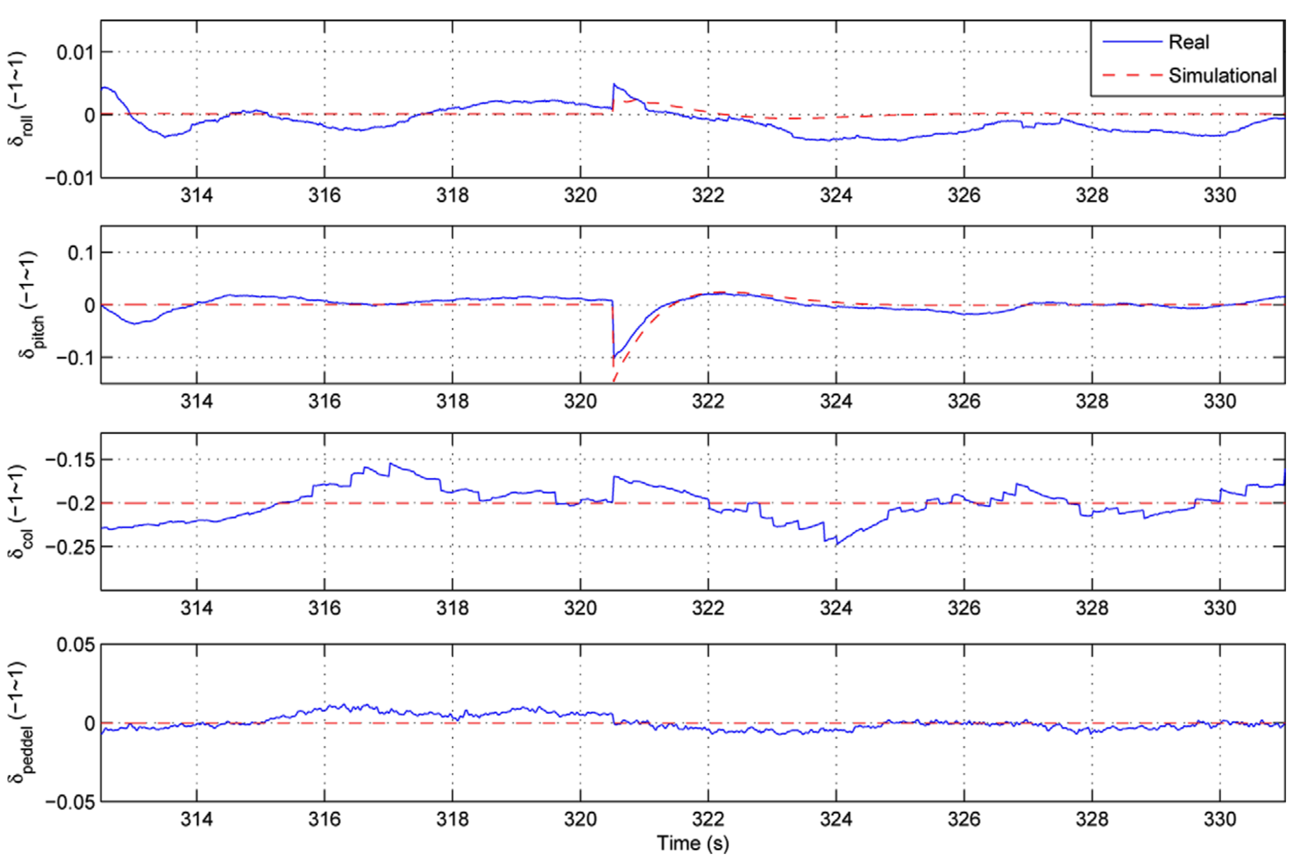

Fig. 11. Input signals of hardware-in-the-loop simulation and actual flight tests.

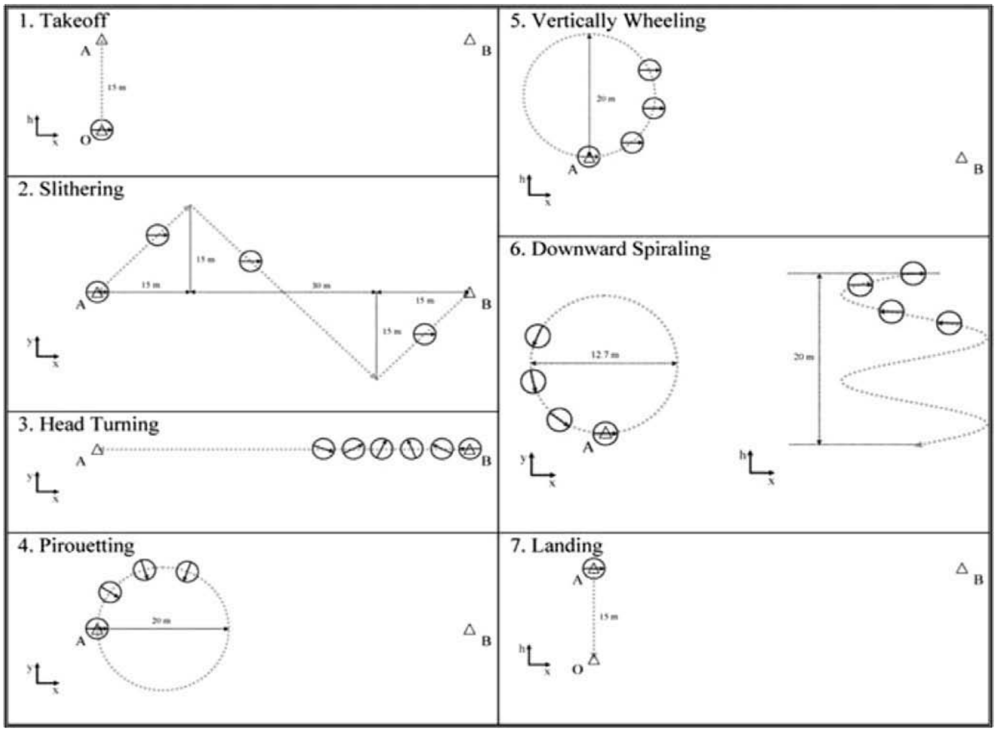

Fig. 12. Flight schedule of the full-envelope flight test.

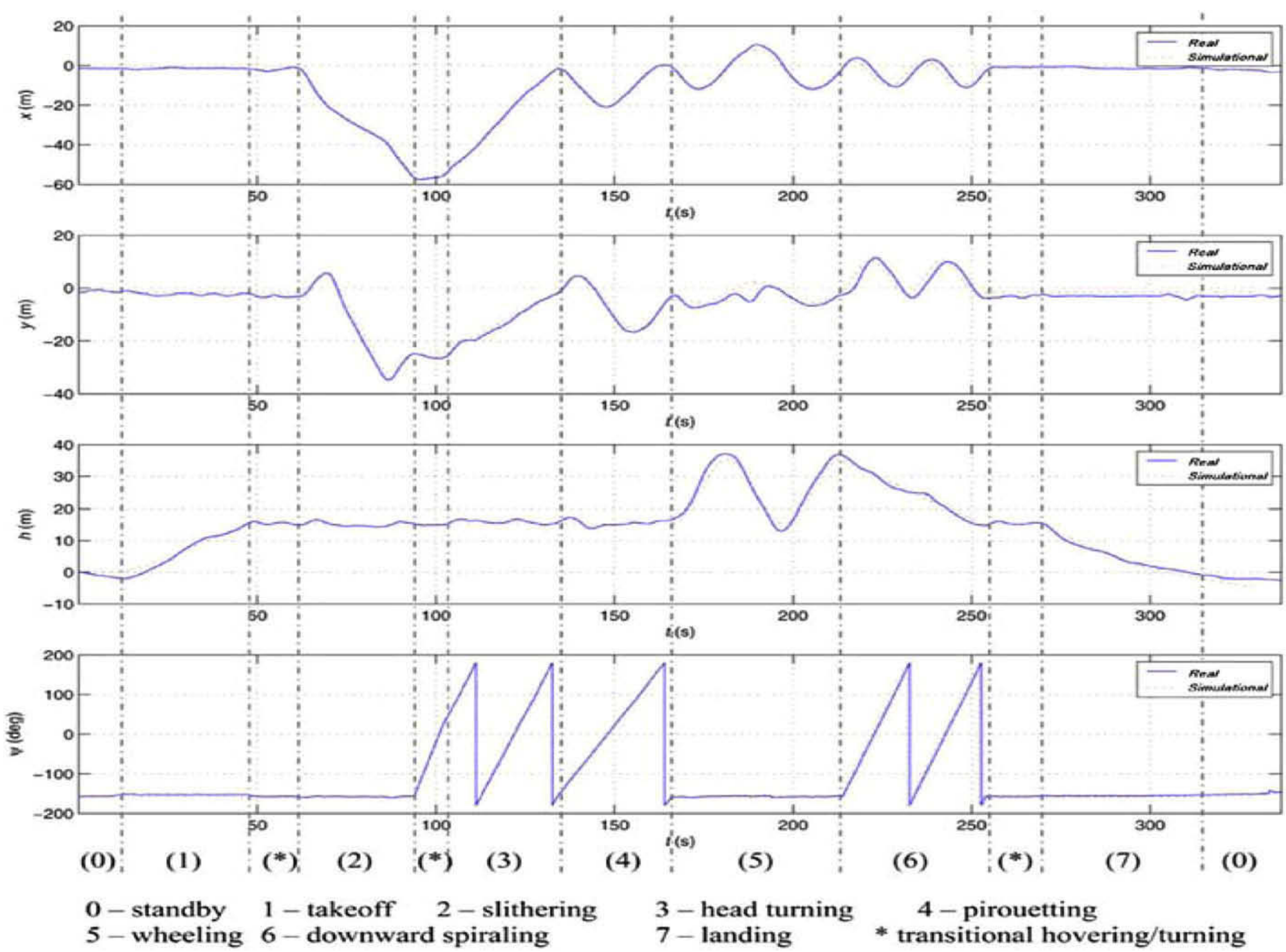

Fig. 13. Results of hardware-in-the-loop simulation and real flight test for full-envelope flight case.

simulation framework consists of two parts, i.e., the onboard software system and the ground station software system.

The structure of the onboard software system is depicted in Fig. 7. A multi-thread running scheme is adopted and it contains one MAIN thread and five task threads, namely, (1) the SIM thread for flight control simulation for the helicopter; (2) the DAQ thread for data acquisition board; (3) the SVO thread for driving the servo actuators; (4) the CMM thread for dual-directional wireless communication with the ground station; and (5) the DLG thread for logging data into the onboard compact flash card. In each loop, all the task threads are controlled by the main thread and executed in the following sequence: SIM → DAQ → SVO → CMM → DLG. We note that the SIM thread is corresponding to the flight control simulation module described earlier. We also refer interested readers to [6] for the detailed implementation of this software scheme using QNX Neutrino [10].

The ground station software system is run on a laptop with the Windows XP Professional. The framework of the ground station software system, shown in Fig. 8, has two layers, i.e., the background layer and the foreground layer. Data transferring runs in background. This layer collaborates with the onboard system (i.e., the CMM task thread) through the wireless channel, receiving data from and sending commands to the onboard system. The foreground layer consists of a variety of views displaying the in-flight data. Four kinds of views are provided in our system, i.e., the state view, curve view, 3D view and command view. The state view shows basic states of the helicopter in a manner of a list of texts. The curve view shows in-flight data in 2D coordination graphs, in which data are generally displayed as a function of time. The 3D view is to reconstruct the actual motion of the helicopter in a more realistic 3D style. Once again, the detailed realization of the ground station software system and its execution scheme can be found in [6].

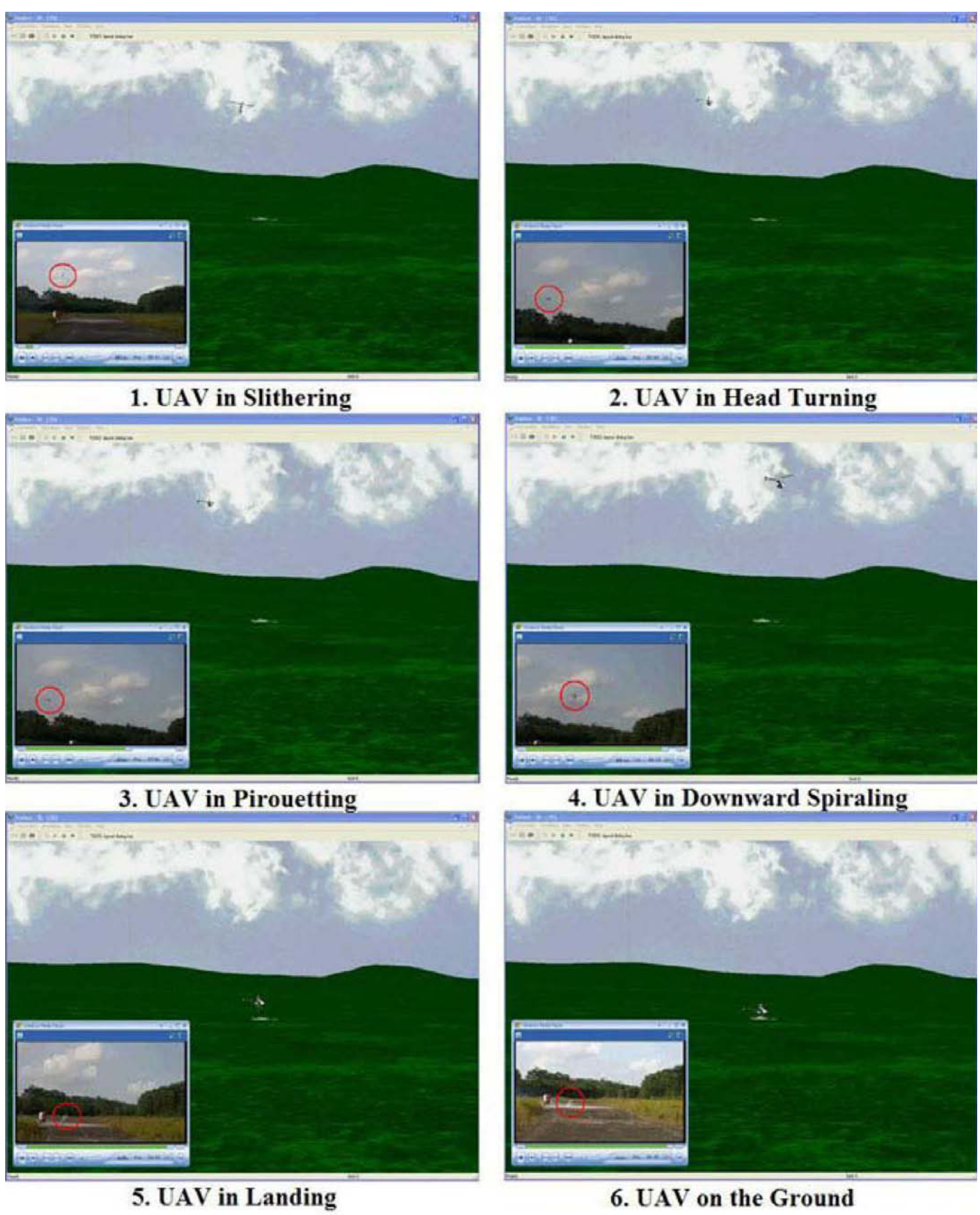

Fig. 14. Comparison between virtual 3D flight and actual flight.

3. Simulation of UAV helicopter systems

In this section, we provide the simulation results using the proposed hardware-in-the-loop simulation system for the following three cases: (i) basic motion flight test, (ii) full-envelope flight test, and lastly, (iii) multiple UAV formation flight test. The dynamic model of the helicopter used in simulation is adopted from that given in [3], and the control algorithm implemented in our simulation and flight tests is adopted from a newly developed nonlinear control technique, i.e., the so-called composite nonlinear feedback control, reported in [4,9]. For easy reference and verification, simulation results are also to be compared to those obtained from actual flight tests.

3.1. Basic motion flight test

This case is to test some basic motions, such as hovering and forward flight of the helicopter. These motions are very important as they are the basic elements of complicated fully autonomous flight. Ensuring the stability and control performance for these flight conditions is a necessary step before conducting any further test on the UAV system. Two basic flight motions are examined in this test. More specifically, the helicopter is switched to an automatic hover mode for about 8 s and then is commanded for a forward flight with a speed of 2 m/s. The results of the corresponding hardware-in-the-loop simulation and actual flight test are shown in Figs. 9–11. It is clear that the hardware-in-the-loop simulation results match pretty well with those obtained from the actual flight tests.

3.2. Full-envelope flight test

In this test, the helicopter is commanded to fly a set of complicated flight paths to examine the overall flight performance of the UAV. The flight patterns included in this test are: (1) automatic takeoff, (2) slithering, (3) head turning, (4) pirouetting, (5) vertically wheeling, (6) downward spiraling, and (7) automatic landing. Fig. 12 shows a graphical illustration of these motions, where **O** is the takeoff and landing point, **A** and **B** are two reference points in the sky, and lastly, **x–h** and **x–y** represent vertical and horizontal planes, respectively.

The results for both the hardware-in-the-loop simulation and real flight tests are given in Fig. 13, in which the dotted lines represent the hardware-in-the-loop simulation result and the solid lines are for actual flight test. Once again, they are pretty well matched. In fact, the hardware-in-the-loop simulation is of a great help and instrumental for us in conducting the actual flight test. We notice that the overall hardware-in-the-loop simulation procedure is performed smoothly without any sudden jump. This is an indication that the designed controller is physically implementable.

As the full-envelope flight covers a wide range of flight motions, we choose this case to verify and illustrate that the 3D-view function. In the whole flight schedule, we select several flight points and compare the emulated 3D-view flight with the recorded video clip. Shown in Fig. 14 is such comparison, in which the video clips are depicted at the left corner with the actual UAV helicopter being marked by a red circle. The results clearly indicate that the emulated 3D flight in the hardware-in-the-loop simulation system is capable of accurately predicting the actual flight paths, and thus is an effective and useful function for simulating real flight situations.

3.3. Multiple UAV formation flight test

At moment for a safety reason, we use the ground station as a virtual leader UAV and command an actual UAV helicopter as a follower to follow the virtual leader to fly certain flight patterns in our multiple UAV formation flight test. We intentionally choose a counter example here to show that the proposed hardware-in-the-loop simulation system is effective in predicting potential dangers in the real environment. Shown in Fig. 15 is a leader–follower trajectory in horizontal plane simulated by our hardware-in-the-loop simulation system. In this experiment, the follower is required to maintain a constant distance of 7 m away with a same heading angle. Because of the improperly designed leader trajectory, the follower has to experience a very fast turning period (180° heading change in 2 s), which is quite dangerous for the real

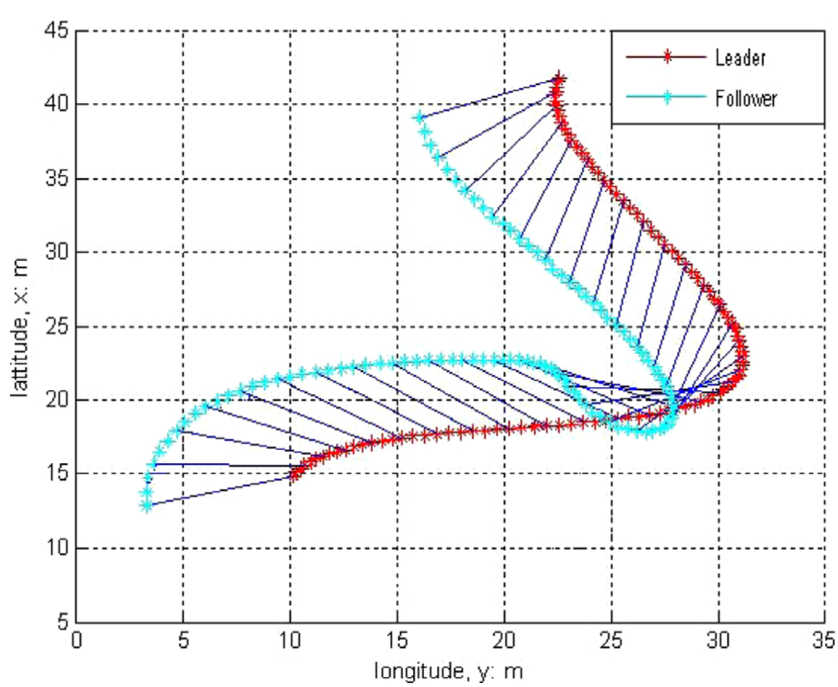

Fig. 15. Flying trajectories of leader and follower in hardware-in-the-loop simulation.

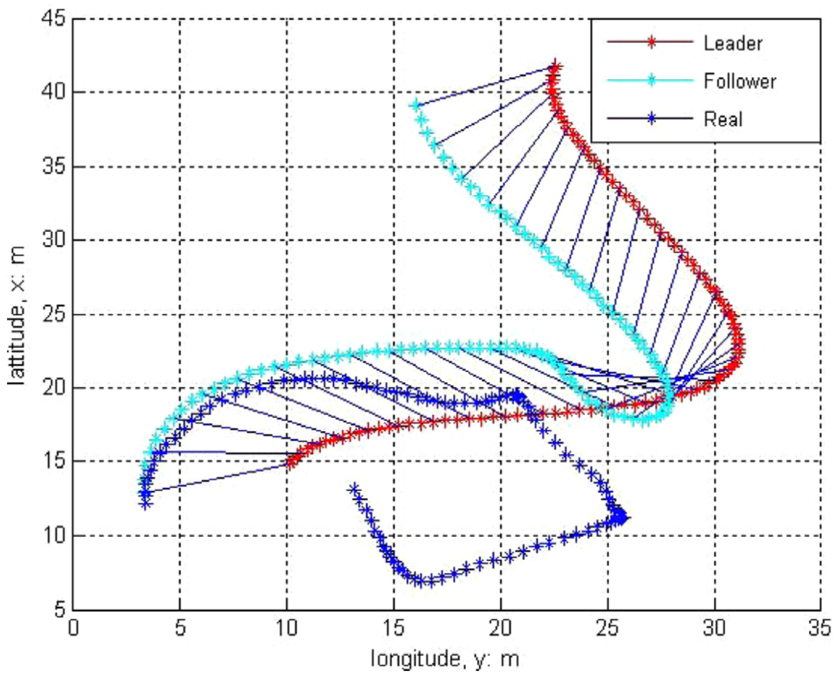

Fig. 16. Flying trajectories of leader, follower and actual flight test.

UAV. An actual flight test is conducted for the purpose of examining the actual flight performance in such an extreme condition. The actual result is shown in Fig. 16. During the fasting turning process, the UAV helicopter has in fact become unstable after the coordinate point (21, 20) in Fig. 16. We note that the trajectory shown after this critical point is flight motions generated by a pilot using the manual control function. Although the experiment is unsuccessful in this test, a serious accident is avoided with a protection scheme we install on the UAV, because of the precaution given by the hardware-in-the-loop simulation system.

4. Conclusions

We have presented in this paper the complete design procedure of a hardware-in-the-loop simulation system for our small-scale UAV helicopters. Simulation results are obtained for three flight tests, namely, the basic motion flight test, the full-envelope flight test and the multiple UAV formation test, and compared with those of the actual flight tests. The results have clearly indicated that the proposed hardware-in-the-loop simulation system is capable of accurately and efficiently predicting the real flight situations including potential dangers and accidents. The system is certainly an effective and useful tool for us in our UAV research.

References

[1] Bortoff SA. The University of Toronto RC helicopter: a test bed for nonlinear control. In: Proceedings of the 1999 IEEE conference on control applications, Hawaii, USA, 1999. p. 333–8.

[2] Cai G, Lin F, Chen BM, Lee TH. Systematic design and construction of UAV helicopters. Mechatronics 2008;18:545–58.

[3] Cai G, Chen BM, Lee TH, Lum KY. Comprehensive nonlinear modeling of an unmanned-aerial-vehicle helicopter. In: Proceedings of the 2008 AIAA guidance, navigation and control conference, Honolulu, Hawaii, USA, AIAA2008-7414, 2008.

[4] Chen BM, Lee TH, Peng K, Venkataramanan V. Composite nonlinear feedback control for linear systems with input saturation: theory and an application. IEEE Trans Automat Contr 2003;48(3):427–39.

[5] Dittrich JS, Johnson EN. Multi-sensor navigation system for an autonomous helicopter. In: Proceedings of the 21st digital avionics systems conference, vol. 2, 2002. p. 8C1/1–19.

[6] Dong M, Chen BM, Cai G, Peng K. Development of a real-time onboard and ground station software system for a UAV helicopter. J Aerosp Comput, Inform, Commun 2007;4:933–55.

[7] Gavrilets V, Shterenberg A, Dahleh MA, Feron E. Avionics system for a small unmanned helicopter performing aggressive maneuvers. In: Proceedings of the 19th digital avionics systems conferences, Philadelphia, USA, 2000. p. 1E2/1–7.

[8] Johnson EN, DeBitetto PA. Modeling and simulation for small autonomous helicopter development. In: Proceedings of the AIAA modeling and simulation technologies conference, New Orleans, USA, 1997.

[9] Peng K, Dong M, Chen Ben M, Cai G, Lum KY, Lee TH. Design and implementation of an autonomous flight control law for a UAV helicopter. In: Proceedings of the 26th Chinese control conference, Zhangjiajie, Hunan, China, vol. 6, 2007. p. 662–7.

[10] QNX Momentics (v6.3 SP2) Documentation, QNX Software Systems Corporation.

[11] Sanvido MAA, Schaufelberger W. Design of a framework for hardware-in-the-loop simulation and its application to a model helicopter. In: Proceedings of international EUROSIM congress, No. 4, Delft, Netherlands, 2001.

[12] Saripalli S, Montgomery JF, Sukhatme GS. Visually-guided landing of an unmanned aerial vehicle. IEEE Trans Robot Automat 2003;19:371–81.

[13] Sprague K, Gavrilets V, Dugail D, Mettler B, Feron E, Martinos I. Design and applications of an avionics system for a miniature acrobatic helicopter. In: Proceedings of the 20th digital avionics systems conference, vol. 1, 2001. p. 3C5/1–10.