



FlightGear application for flight simulation of a mini-UAV

Tomáš Vogeltanz and Roman Jašek

Citation: [AIP Conference Proceedings](#) **1648**, 550014 (2015); doi: 10.1063/1.4912769

View online: <http://dx.doi.org/10.1063/1.4912769>

View Table of Contents: <http://scitation.aip.org/content/aip/proceeding/aipcp/1648?ver=pdfcov>

Published by the [AIP Publishing](#)

Articles you may be interested in

[The effect of ground sensor and UAV flight path geometry on the tomographic reconstruction of a weakly sheared daytime convective planetary boundary layer](#)

J. Acoust. Soc. Am. **137**, 2387 (2015); 10.1121/1.4920687

[JSBSim library for flight dynamics modelling of a mini-UAV](#)

AIP Conf. Proc. **1648**, 550015 (2015); 10.1063/1.4912770

[Preface of the "Mini symposium on generalized inverses and their applications"](#)

AIP Conf. Proc. **1558**, 1495 (2013); 10.1063/1.4825803

[Gear Forging Simulation Using Cyclic Symmetry](#)

AIP Conf. Proc. **712**, 458 (2004); 10.1063/1.1766567

[Landing gear in-flight vibration dampener](#)

J. Acoust. Soc. Am. **73**, 1093 (1983); 10.1121/1.389093

FlightGear Application for Flight Simulation of a mini-UAV

Tomáš Vogeltanz and Roman Jašek

*Tomas Bata University in Zlín, Faculty of Applied Informatics
Department of Informatics and Artificial Intelligence
nám. T.G. Masaryka 5555, 760 01 Zlín, CZECH REPUBLIC*

Abstract. This paper presents using of FlightGear application for flight simulation of a mini-UAV (Unmanned Aerial Vehicle). The first part of the paper is about general information of UAVs and about the fundamentals of airplane flight mechanics. The main section summarizes the information about FlightGear application and describes a way of using the application to simulate a mini-UAV. The process of altitude change by using autopilot has been simulated. In the created simulation, three stage cascade controller was used for an altitude control. Heading was controlled by 2 stage cascade PID (Proportional-Integral-Derivative) controller. An automatic throttle system was applied for a velocity control. The setting of all controllers is included in the description of the simulation. Finally, the results of FlightGear simulation are discussed.

Keywords: aircraft; airplane; autopilot; FlightGear; modelling; PID; simulation; UAV

PACS: 07.05.Tp

INTRODUCTION

Mini-UAVs are an inexpensive alternative to manned aircraft for a lot of applications including safety, aerial reconnaissance, surveying, environmental monitoring, and agriculture. The research community does not clearly define the term mini-UAV (or small-UAV). [1] [2] [3] [4] Instead of creation new definition about weight and proportions of the mini-UAV, we can simply say, a UAV is mini (or small) if the strength of an average man suffices for lifting and moving the UAV without any trouble. For the remainder of this paper, the term mini-UAV is implied.

The flight-testing of an aircraft is a well-documented engineering procedure. However, every aircraft design, construction, implementation and test is unique and presents different challenges to engineers, pilots, and test team. The same can be said about UAVs, but the criteria for UAVs can be different from manned aircraft. [2] [5]

UAVs promise greater precision, but the auto-pilot system which keeps the vehicle in the air and in control is critical to the success of UAV systems. In many cases, testing newly developed autopilot systems in a virtual environment is the only way to guarantee absolute safety. Additionally, the model allows better repeatability in testing. [1] [2]

However, a software model developed from first principles has unknown accuracy and a major effort of any aerospace company is the creation of a high-fidelity six degree of freedom (6DOF) simulation for each of the vehicles which are developed. For the model which is supposed to be used for real simulation, it is necessary to include implementation, verification and validation into its development process. [1] [6]

FUNDAMENTALS OF AIRPLANE FLIGHT MECHANICS

The forces of lift, weight, drag, thrust, and side force act along the axes, forcing the aircraft to move in the axes direction. The three moments, yaw, roll, and pitch force the aircraft to turn around the axes. X-axis is the roll axis, pointing forwards along the symmetry axis. Y-axis is the pitch axis, pointing rightwards from aircraft. Z-axis is the yaw axis, pointing downwards. [1] [2] [3] [4]

The equations of motion are composed of translational (force) equations and rotational (moment) equations and are called the 6DOF equations of motion. The aircraft can move in three dimensions in space and can rotate about three axes. The origin of the coordinate system of the UAV is located at its center of gravity. Figure 1 shows the forces and moments acting on an aircraft. [1] [2] [3] [4] [6]

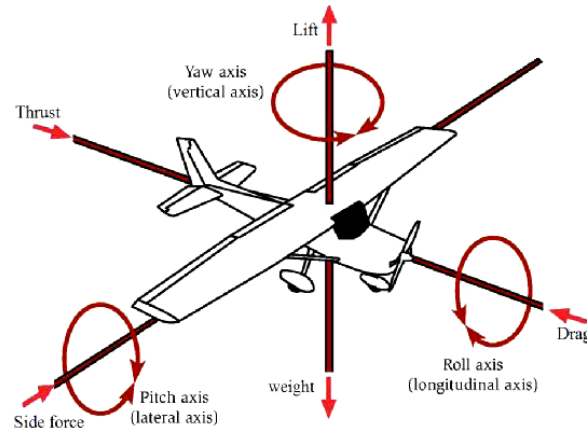


FIGURE 1. The forces and moments acting on an aircraft

FLIGHTGEAR FLIGHT SIMULATOR

FlightGear is an open-source flight simulator, written in the C++ programming language, to model and simulate an aircraft. Data visualization is another aspect considered while building the flight dynamics model. The common 3D format used in FlightGear has “ac” extension. FlightGear can produce a 3D graphic animation in real time and is connected to Flight Dynamics Model (FDM). [1] [3] [9]

FDM is the physics/math model which defines the movement of an aircraft under the forces and moments applied to it using the various control mechanisms and from the forces of nature. FDM includes development of a physical, inertial, and aerodynamic model representing the UAV. [1] [2] [3]

It is possible to choose between two primary FDMs: JSBSim and YASim. Actually, there is a third FDM named UIUC and based on LaRCsim originally written by the NASA but UIUC is no longer supported by default FlightGear builds. It is possible to add new dynamics models or even interface to external flight dynamics models. [9] [10]

JSBSim is open source software for a 6DOF nonlinear flight dynamics modelling. JSBSim is generally considered as a very accurate FDM and allows the modelling of any aircraft, missile, or rotorcraft, and models the rotational earth effects on the equation of motion. Particular aircraft flight control systems, propulsion, aerodynamics, landing gears, and autopilot are defined in eXtensible Markup Language (XML) format files. [1] [3] [7] [8]

YASim is an integrated part of FlightGear and uses a different approach than JSBSim by simulating the effect of the airflow on the different parts of an aircraft. The advantage of this approach is that it is possible to perform the simulation based on geometry and mass information combined with more commonly available performance numbers for an aircraft. This allows for quickly constructing a plausibly behaving aircraft that matches published performance numbers without requiring all the traditional aerodynamic test data. [9]

Except of FDM configuration files (e.g. aircraft, engine and thruster configuration files), other files are required for use with FlightGear flight simulator which include the electric system file, autopilot file, and 3D graphical model specification file. The final file required is a file to tie the previous files together in order to perform complete simulation using FlightGear. [3]

The electrical system file specifies the battery characteristics, and the lights and other parameters. [3]

To fly the modelled UAV autonomously, a tuning process have to be made for the built-in generic PID (proportional, integral, and derivative) autopilot of FlightGear which has the ability to hold aircraft velocity, vertical aircraft speed, altitude, pitch angle, angle of attack, bank angle, and true heading. [3]

FlightGear implements a PID algorithm in a flexible way which makes it reusable with similar aircraft. Any number of PID controllers can be defined in the autopilot configuration file. Cascading controllers can be implemented by specifying multiple PID controllers in which the output of the current stage is used as the input to the next stage. [3]

A flight path which contains a number of waypoints chosen over a selected area can be constructed. In order to use the chosen waypoints with FlightGear navigation system, a unique ID can be assigned to each waypoint, and the FlightGear database can be altered to include the new waypoints with their IDs. [3]

The fixed waypoints are determined by latitude and longitude. When a waypoint is entered in the aircraft route during the simulation time, FlightGear checks the database to see if it is a valid fixed point or not. This data is stored in the compressed file called fix.dat which can be found in the directory FG_ROOT\FlightGear\data\Nav aids. [3]

FlightGear Autopilot Simulation

Rascal 110 UAV [11] was used for the simulation. An autopilot for altitude control was done by using three stage cascade controller. The first controller was PI and the others were PID. A heading control was carried out by using 2 stage cascade PID controller. An automatic system for a velocity control was created by using one PID controller. The velocity change was compensated by throttle.

The parameter setting of all controllers is shown in Table 1. The Parameters: $\alpha = 0.1$, $\beta = 1$, $\gamma = 0$ were set to all PID controllers. The meaning of the parameters can be seen in Table 2.

TABLE 1. The parameter setting of the controllers

Altitude Control		Heading Control		Velocity Control	
Parameter	Value	Parameter	Value	Parameter	Value
K_{p1}	0.3	K_{pid1}	-0.1	K_{pid1}	0.1
K_{i1}	0	T_{i1}	10	T_{i1}	1
K_{pid2}	0.5	T_{d1}	10^{-5}	T_{d1}	10^{-3}
T_{i2}	1	K_{pid2}	0.05		
T_{d2}	10^{-5}	T_{i2}	0.5		
K_{pid3}	-0.05	T_{d2}	0.01		
T_{i3}	0.5				
T_{d3}	10^{-3}				

TABLE 2. The meaning of the parameters

Parameter	Meaning
K_p	Gain of the proportional component
K_i	Gain of the integral component
K_{pid}	The overall gain for the proportional, integral and derivative part
T_i	Integral time
T_d	Derivative time
α	Scaling factor for T_d
β	Reference weighing factor for the proportional component
γ	Reference weighing factor for the derivate component

In FlightGear application, lifting of the UAV from altitude of 152.4 meters (500 feet) to altitude of 304.8 meters (1000 feet) was simulated. The airspeed was set to 124.2 km/h at the beginning of the simulation, in other words, it was set to almost top speed of the UAV. The velocity control tried to hold the top speed. Heading control was adjusted to keep the heading at 0. Altitude control was set to change the altitude to 304.8 meters and hold it.

Figure 2 and Figure 3 show the process of velocity and of altitude change. The UAV begins to climb immediately after sending the command. The UAV goes up by 152.4 meters during the 61.5 seconds. Then, the UAV is completely stabilized after a while. During the lifting, the velocity decreases to 113.26 km/h. The UAV achieves almost top speed (123.1 km/h) after 17.67 seconds.

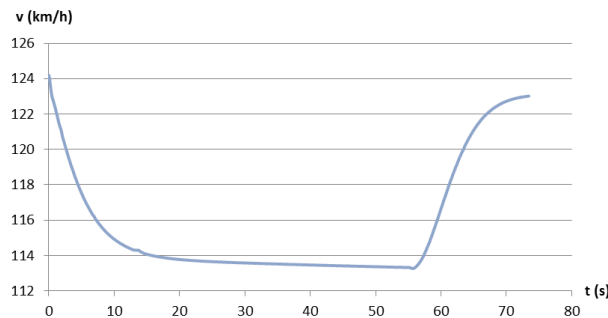


FIGURE 2. The process of velocity change of the UAV

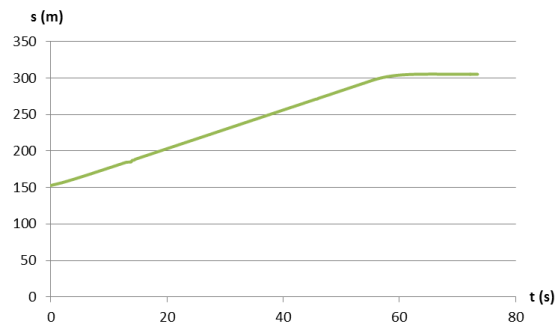


FIGURE 3. The process of altitude change of the UAV

CONCLUSION

This paper has presented using of FlightGear for flight simulation of a mini-UAV. The combination of JSBSim Flight Dynamic Model and FlightGear Flight Simulator provide an excellent base for building the simulation environment.

Essential configuration files have to be constructed to model the UAV in JSBSim and simulate it with FlightGear. These files include main UAV configuration, engine configuration, propeller configuration, electric system configuration, 3D model configuration, and an autonomous flight autopilot configuration file. All these files are tied together in the top level configuration file.

The main section of this work has shown the option how to simulate flight and how to test an autopilot system. The autopilot system of the simulated UAV was made by using PI and PID controllers. The setting of the PI and PID controllers has been described. The process of altitude change by using the autopilot has been simulated.

It has been found that the accurate stabilization of UAV can be done simply by using PI and PID controllers. It has been proved that when UAV soars, its velocity decreases fast at first and then slower.

ACKNOWLEDGMENTS

This work was supported by the Internal Grant Agency at Tomas Bata University in Zlín, project no. IGA/FAI/2014/006.

REFERENCES

1. X.Q. Chen, Y.Q. Chen and J.G. Chase, *Mobile Robots - State of the Art in Land, Sea, Air, and Collaborative Missions*. Croatia: In-Teh, May 2009, pp.177-201.
2. N. M. Jodeh, *Development of Autonomous Unmanned Aerial Vehicle*. Ohio, USA: Wright-Patterson Air Force Base, March 2006, 185 p.
3. T. Abdunabi, *Modelling and Autonomous Flight Simulation of a Small Unmanned Aerial Vehicle*. Sheffield, UK: The University of Sheffield, August 2006, 61 p.
4. Z. Ye, P. Bhattacharya, H. Mohamadian, H. Majleseini, and Y. Ye, "Equational Dynamic Modeling and Adaptive Control of UAV," in Proceedings of the 2006 IEEE/SMC International Conference on System of Systems Engineering, Los Angeles, CA, USA, 2006, April, pp. 339-343.
5. R. Austin, *Unmanned Air Systems: UAVs Design, Development and Deployment*. Wiltshire, UK: Wiley, 2010, 332 p.
6. D. G. Hull, *Fundamentals of Airplane Flight Mechanics*. Springer, 2007, 298 p.
7. JSBSim contributors. *JSBSim Open Source Flight Dynamics Model*. [Online]. Available: <http://jsbsim.sourceforge.net/>
8. JSBSim contributors. (2005, December 31). *Aeromatic*. [Online]. Available: <http://jsbsim.sourceforge.net/aeromatic2.html>
9. FlightGear contributors. *FlightGear Flight Simulator: Features*. [Online]. Available: <http://www.flightgear.org/about/features/>
10. FlightGear contributors. (2014, May 12). *FlightGear Wiki - UIUC*. [Online]. Available: <http://wiki.flightgear.org/UIUC>
11. J. Goppert. (2013, January 13). *Rascal*. [Online]. Available: <https://github.com/PX4/HIL/tree/master/aircraft/Rascal>