

UNMANNED AERIAL VEHICLE MISSION SIMULATION IN FLIGHTGEAR

Konstantin Metodiev

Space Research and Technology Institute – Bulgarian Academy of Sciences
e-mail: komet@space.bas.bg

Keywords: *Unmanned Aerial Vehicle, FlightGear*

Abstract: *In the report hereby, a relatively easy-to-implement unmanned aerial vehicle simulator is considered. A 3D model of the aircraft was created in a CAD environment and subsequently exported into a format used by FlightGear simulator. Stability and handling quantities of the aircraft are obtained through CFD analysis and recorded in xml files that the simulator recognizes. A longitudinal flight control loop was investigated in addition. A test flight task was performed in the simulator and the obtained results were analyzed.*

ПОЛЕТНА ЗАДАЧА НА БЛА В СИМУЛАТОР FLIGHTGEAR

Константин Методиев

Институт за космически изследвания и технологии – Българска академия на науките
e-mail: komet@space.bas.bg

Ключови думи: *Безпилотен летателен апарат, FlightGear*

Резюме: *В настоящия доклад е разгледан сравнително лесен за реализация симулатор на безпилотен летателен апарат. Триизмерен модел на летателния апарат е създаден в CAD среда и впоследствие трансформиран във формат, използван от симулатор FlightGear. Характеристики на устойчивост и управляемост на самолета са получени чрез CFD анализ и записани в xml файлове, които симулаторът разпознава. Допълнително е изследван контур за автоматично управление на самолета в надлъжното движение. Изпълнена е тестова полетна задача в симулатора и са анализирани получените резултати.*

Introduction

Unmanned aerial vehicles (UAVs), also known as drones, are aircraft operated without a human pilot on board. They are controlled remotely by a human operator or can autonomously follow pre-programmed instructions. UAVs are utilized for various purposes, such as aerial surveillance, photography and videography, cargo delivery, scientific research, mapping, disaster response, and recreational activities, [1].

There are indeed simulators available for UAVs. Drone simulators provide a realistic and immersive environment to practice flight training, conduct mission planning, test new equipment configurations, and simulate various flight scenarios without the risk of damaging actual aircraft. These simulators can be beneficial for both recreational drone enthusiasts and professional operators.

Drone simulators commonly offer features such as:

- **Flight training:** Simulators help beginners learn the basics of drone flight controls, navigation, and maneuvering. They provide a safe environment for practicing takeoffs, landings, and various flight patterns.
- **Camera simulation:** Many simulators incorporate camera simulation, allowing users to practice aerial photography and videography skills. They often simulate different camera modes, settings, and stabilization controls.
- **Realistic environments:** Simulators offer a variety of virtual landscapes to mimic different flying conditions. This includes urban areas, rural landscapes, forests, mountains, and more.

Weather conditions like wind, rain, fog, or even emergencies such as engine failures can be simulated as well.

- Flight planning: Users can create and simulate missions, including flight paths, waypoints, and specific objectives. This is particularly useful for professional applications such as mapping, surveying, or search-and-rescue operations.
- Virtual models: Simulators have models of various drone types, allowing users to practice with specific drone models they intend to operate in reality. This helps simulate different flight characteristics, payloads, and limitations.

Some popular UAV simulators include DRL (Drone Racing League) Simulator, VelociDrone, Liftoff, FPV Freerider, and RealFlight, among others. These simulators have different features, physics engines, and available drone models. Some of them even support multiplayer races or competitions. Free and paid options are available, catering to both beginners and experienced pilots.

The project aim is to come up with methods and resources of doing a sophisticated yet affordable drone simulator yourself. The simulator is expected to serve multiple purposes, such as a teaching tool, to work out a solution to the mission planning problem, etc. The following paragraphs takes into consideration all design stages in succession.

Materials and Methods

In Fig. 1, two renders made by Autodesk Inventor Studio are shown, [2, 3]. The model layout bears a resemblance to many other UAVs widely available. The model of power plant has been inspired by Chinese DLE-60 [4] which is a piston, two-stroke, air cooling engine with a diaphragm carburetor and a backward mounted pusher propeller. The suggested empennage (a twin boom inverted-V tail) configuration is preferable but not obligatory. Deliberately accelerating the airflow through a classic U-shaped tail for instance would act upon the control surfaces, thus affecting (even worse, jeopardizing) the longitudinal stability and control. It is up to designer to make a right decision. The payload consists of a near infrared (NIR) and a visible light camera mounted on a gimbal. An advantage of the NIR camera is its ability to produce brighter images in dark environments, [5]. The payload is installed right behind the nose landing gear (NLG) as far away from the engine as possible. At first sight, it may look like a poor choice because the NLG strut does impede the camera field of view, indeed. On the other hand, both engine heat and vibrations must not interfere with the payload. What is more, the fuel tank is always being mounted right beneath the wing least the remaining fuel should shift the mass center location beyond allowable limits. In addition, a first-person view camera is mounted atop the tail. The autopilot might be installed wherever it is possible provided distance to the airplane mass center is known and set in advance. A one-stage Fowler flap is mounted on the wing to increase airfoil mean aerodynamic chord and camber when deployed. The Fowler flap is said to create a substantial amount of lift at the cost of a complicated and heavy deploying mechanism, a four-bar linkage in this particular study case, [2, 3].

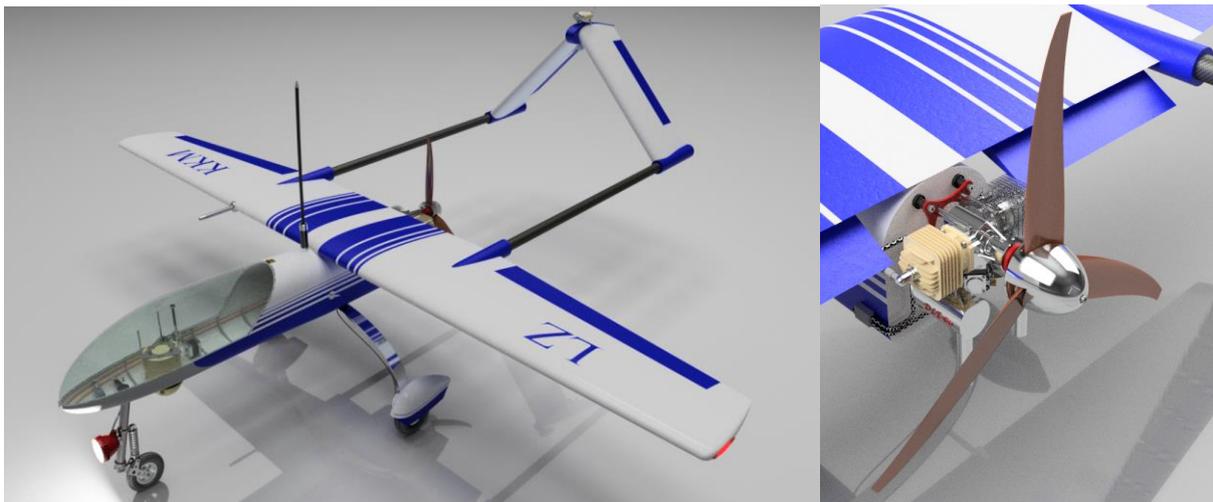


Fig. 1. “Flight of Fancy” project, [2] (left half) and engine DLE-60 with pusher propeller

A simplified longitudinal displacement autopilot has been used in the project to maintain constant pitch angle during steady longitudinal flight. Consider equations of aircraft longitudinal motion with stick fixed derived in textbook [6]. The reader might refer to eq. (1-59), (1-103), (1-128), and table 1-1. All addends are dimensionless.

$$\begin{aligned}
(1) \quad & \left(\frac{mU}{Sq} \dot{u} - C_{x,u} u \right) - \left(\frac{c}{2U} C_{x,\dot{\alpha}} \dot{\alpha} + C_{x,\alpha} \alpha \right) - \left(\frac{c}{2U} C_{x,q} \dot{\theta} + C_w \theta \right) = C_{Fx} \\
& -C_{z,u} u + \left(\frac{mU}{Sq} - \frac{c}{2U} C_{z,\dot{\alpha}} \right) \dot{\alpha} - C_{z,\alpha} \alpha - \left(\frac{mU}{Sq} + \frac{c}{2U} C_{z,q} \right) \dot{\theta} = C_{Fz} \\
& -C_{m,u} u + \frac{c}{2U} C_{m,\dot{\alpha}} \dot{\alpha} - C_{m,\alpha} \alpha + \left(\frac{I_y}{Sq c} \ddot{\theta} - \frac{c}{2U} C_{m,q} \dot{\theta} \right) = C_m
\end{aligned}$$

In equations above $\mathbf{m} = \mathbf{const}$ is aircraft mass, α is angle of attack, θ is angle of pitch, \mathbf{c} is mean aerodynamic chord, \mathbf{S} is wing planform area, \mathbf{q} is dynamic pressure, \mathbf{u} is longitudinal velocity, \mathbf{U} is total velocity, \mathbf{C}_x , \mathbf{C}_z , and \mathbf{C}_m are aerodynamic coefficients (drag, lift, pitch moment), $\mathbf{C}_w = -\mathbf{mg}/(\mathbf{S}\mathbf{q})$ is gravity coefficient. In equations (1), lateral disturbance forces and moments haven't been taken into account whatsoever.

Stability derivatives taking part in eq. (1) have been estimated according to a recipe published in textbook [7], p.125, regarding an aircraft with conventional tail. The inverted V-tail stability derivative $\mathbf{C}_z, \alpha_{tail}$, taking part in \mathbf{C}_m, α computation, assumes value of thin plate due to a lack of reliable data.

After applying Laplace transformation with zero initial conditions and assuming no external inputs, i.e. $\mathbf{C}_{Fx} = \mathbf{C}_{Fz} = \mathbf{C}_m = \mathbf{0}$ the system equations might be written down in operator form as follows

$$\begin{aligned}
(2) \quad & \left(\frac{mU}{Sq} s - C_{x,u} \right) u - \left(\frac{c}{2U} C_{x,\dot{\alpha}} s + C_{x,\alpha} \right) \alpha - \left(\frac{c}{2U} C_{x,q} s + C_w \right) \theta = 0 \\
& -C_{z,u} u + \left[\left(\frac{mU}{Sq} - \frac{c}{2U} C_{z,\dot{\alpha}} \right) s - C_{z,\alpha} \right] \alpha - \left(\frac{mU}{Sq} + \frac{c}{2U} C_{z,q} \right) s \theta = 0 \\
& -C_{m,u} u + \left(\frac{c}{2U} C_{m,\dot{\alpha}} s - C_{m,\alpha} \right) \alpha + \left(\frac{I_y}{Sq c} s^2 - \frac{c}{2U} C_{m,q} s \right) \theta = 0
\end{aligned}$$

In order to simplify the equations further, short period mode of oscillatory motion is solely considered. Forces acting in \mathbf{X} direction does not contribute to the short period oscillation, therefore the equation containing forces in that direction might be omitted. According to a suggestion made in [6], derivatives $\mathbf{C}_z, \dot{\alpha}$, \mathbf{C}_z, u , \mathbf{C}_m, u might also be neglected for some reason. After introducing control derivatives \mathbf{C}_z, δ_e and \mathbf{C}_m, δ_e following system is obtained:

$$\begin{aligned}
(3) \quad & \left(\frac{mU}{Sq} s - C_{z,\alpha} \right) \alpha - \left(\frac{mU}{Sq} + \frac{c}{2U} C_{z,q} \right) s \theta = C_{z,\delta_e} \delta_e \\
& \left(\frac{c}{2U} C_{m,\dot{\alpha}} s - C_{m,\alpha} \right) \alpha + \left(\frac{I_y}{Sq c} s - \frac{c}{2U} C_{m,q} \right) s \theta = C_{m,\delta_e} \delta_e
\end{aligned}$$

Solution to system (1) might be worked out by substitution to obtain a transfer function concerning pitch angle θ response and elevator angle δ_e input. The transfer function is a part of the longitudinal autopilot block diagram. Mass \mathbf{m} , moment of inertia \mathbf{I}_y , mean aerodynamic chord \mathbf{c} have been measured in CAD model development environment. The longitudinal stability derivatives are computed after implementing a CFD analysis.

In Fig. 2, the SimWorks integrated development environment is shown, so are residual plots of the aerodynamic coefficients (right half). The console window at the bottom outputs the aerodynamic coefficients values obtained after completing last iteration. In the middle, the aircraft placed within the computational domain is visible. Trial version of SimWorks does not include a module for transient flow analysis which is the reason why the propeller rotational motion and following vortex shed haven't been taken into account. Given the inverted V-tail however, the numerical results are not expected to be distorted significantly as it was explained earlier in the paragraph. Hence, the results might be used to fill in the aircraft configuration xml file in FlightGear directory.

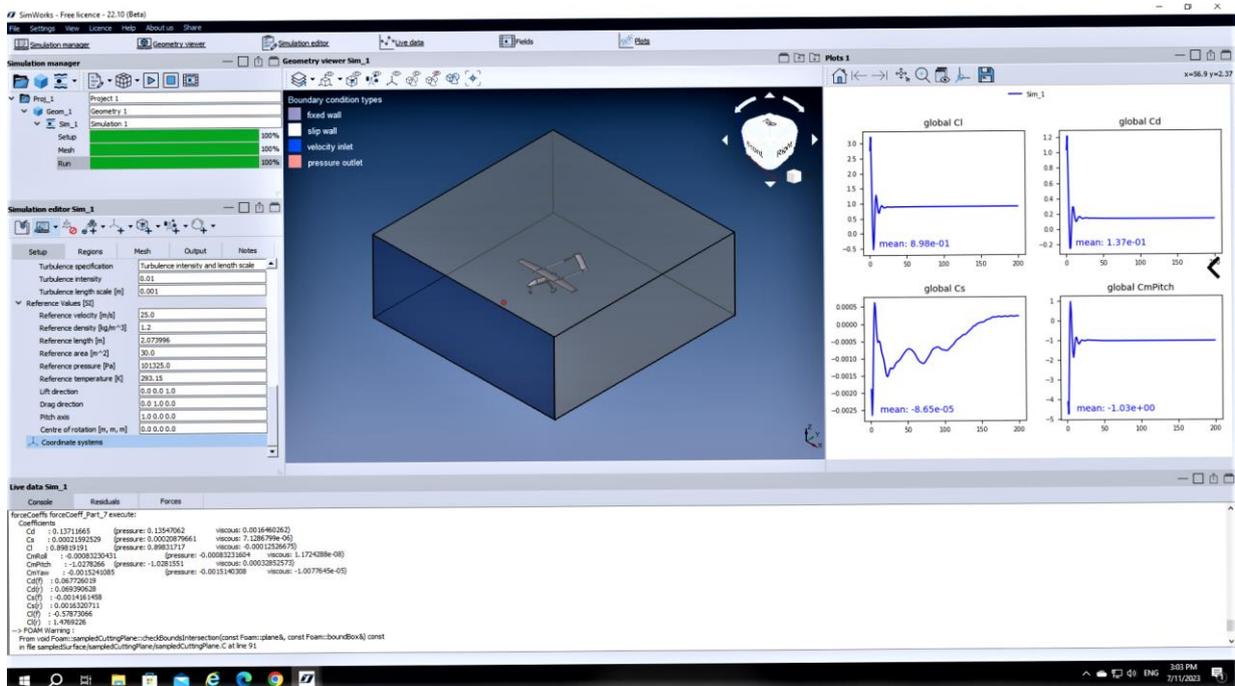


Fig. 2. SimWorks development environment with residual plots

Results

In Fig. 3, both drag C_D and lift C_L aerodynamic coefficients obtained by CFD analysis are shown. At zero angle of attack, the lift coefficient is virtually zero because the wing foil is nearly symmetrical. The lift curve slope $\partial C_L / \partial \alpha = 5.92 \text{ rad}^{-1}$. The minimum (parasite) drag coefficient $C_{D\min} = 0.062$. The zero-lift drag coefficient is slightly bigger, $C_{D0} \approx 0.0625$. The maximum lift-to-drag ratio of 7.1 occurs at angle of attack 6 deg approximately.

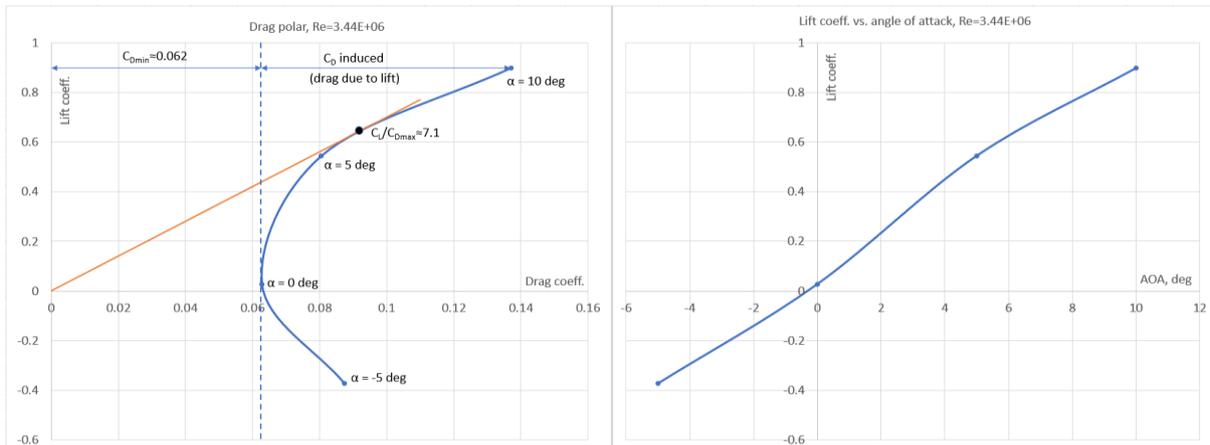


Fig. 3. Aerodynamics coefficients plots: drag polar (left) and lift coefficient vs. angle of attack

In Fig. 4, xCos block diagram of a simple longitudinal autopilot with pitch feedback is shown. Two transfer functions are connected in series: elevator servo and aircraft longitudinal dynamics according to eq. (3). The system is known to yield a steady state error.

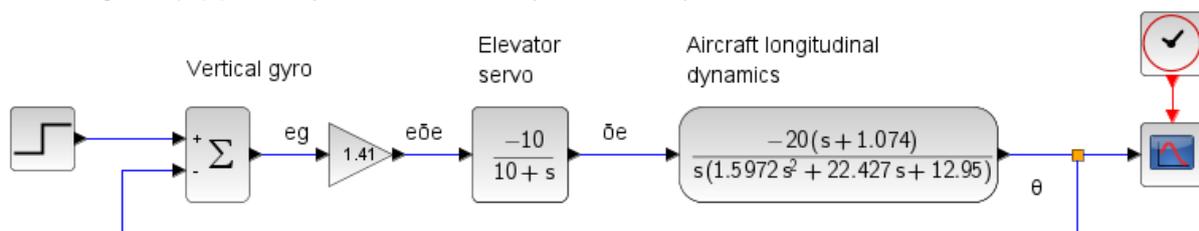


Fig. 4. xCos block diagram of a longitudinal autopilot with pitch feedback

After introducing a pitch rate feedback, Fig. 5, the steady state error is expected to diminish.

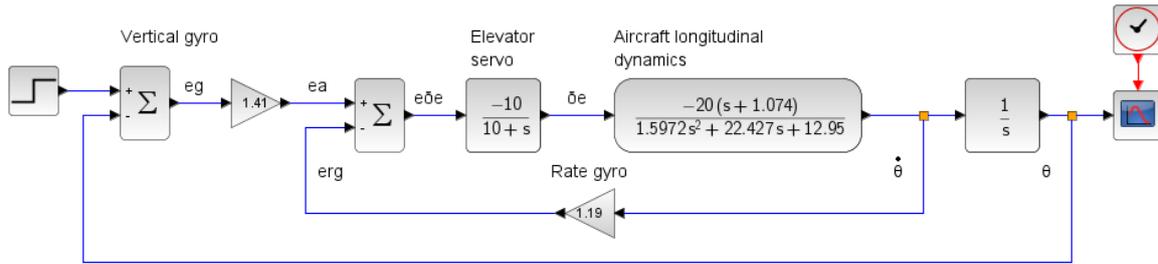


Fig. 5. xCos block diagram of a longitudinal autopilot with pitch rate feedback

In Fig 6, both autopilot responses are shown after applying a unit step function to the elevator:

$$(1) \quad \theta(t) = \begin{cases} 10 \text{ deg}, & t \geq 2 \text{ sec} \\ 0 \text{ deg}, & t \leq 2 \text{ sec} \end{cases}$$

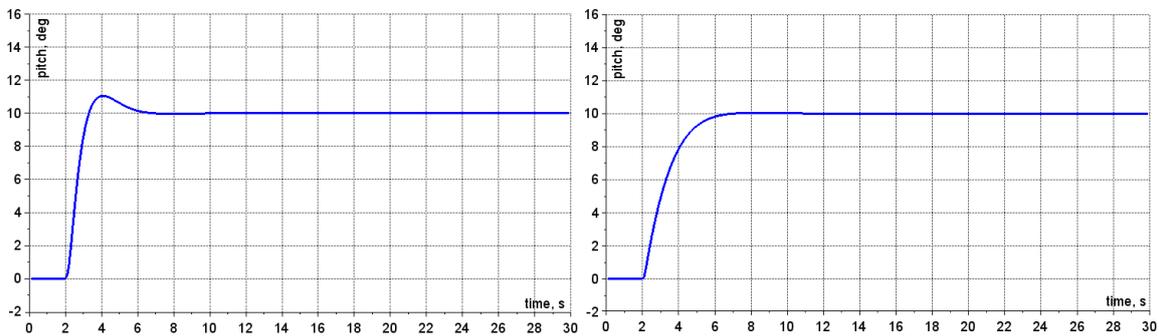


Fig. 6. Step responses with pitch (left) and pitch rate feedback

In Fig. 7, root loci of block diagrams are shown as described in figure caption.

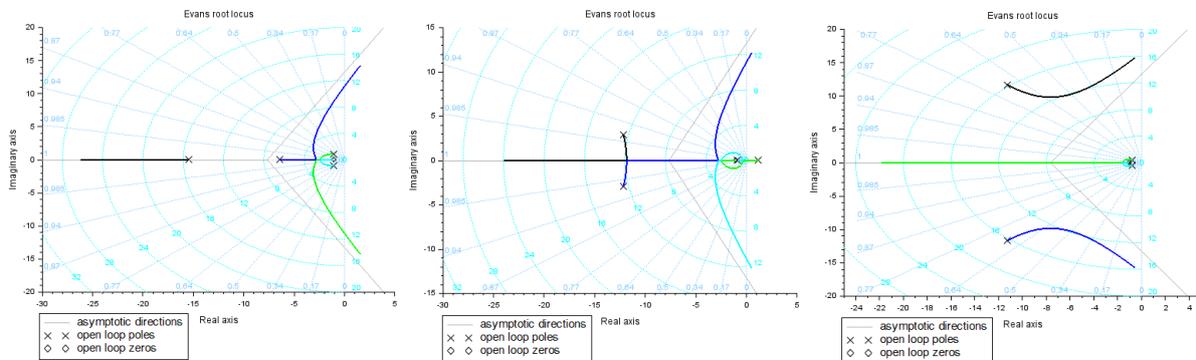


Fig. 7. Root loci: a) pitch feedback, b) inner, c) outer loops of pitch rate feedback

The lift force coefficient of tail surface with respect to a change in angle of attack assumes value of $C_{z,\alpha_{tail}} = 6.28 \text{ rad}^{-1}$. Following values of stability derivatives are obtained: $C_{z,\alpha} = -5.9825$; $C_{z,q} = -13.284$; $C_{z,\delta e} = -0.9153$; $C_{m,\alpha} = -0.3764$; $C_{m,q} = -53.021$; $C_{m,\delta e} = -3.6532$; $C_{m,\dot{\alpha}} = -26.643$

In Fig. 8, a screenshot depicting final approach, payload view, and FlightGear user interface Phi, [8] is shown. The airplane follows an airport traffic pattern, stored in an XML file, with rectangular shape. Needless to say, the file should be loaded by *AutopilotRoute ManagerLoad ...* dialog prior to mission start. The map is to be opened further by *EquipmentMap (Opens in browser)* dialog. In Phi environment, the user can draw the route on top of the map by activating check boxes of drop-down menu in the upper right corner. In order to establish a connection with Internet browser, the FlightGear executable must be invoked by option `--httpd=8080`. The payload azimuth and elevation angle can be adjusted by holding the left mouse button pressed and dragging the mouse over the window. The payload field of view can also be adjusted by means of the middle mouse button. The payload parameters can be tuned independently thanks to a composite viewer which allows multiple independent views of the scene, [9]. The composite viewer is only supported by nightly build of FlightGear v. 2020.4

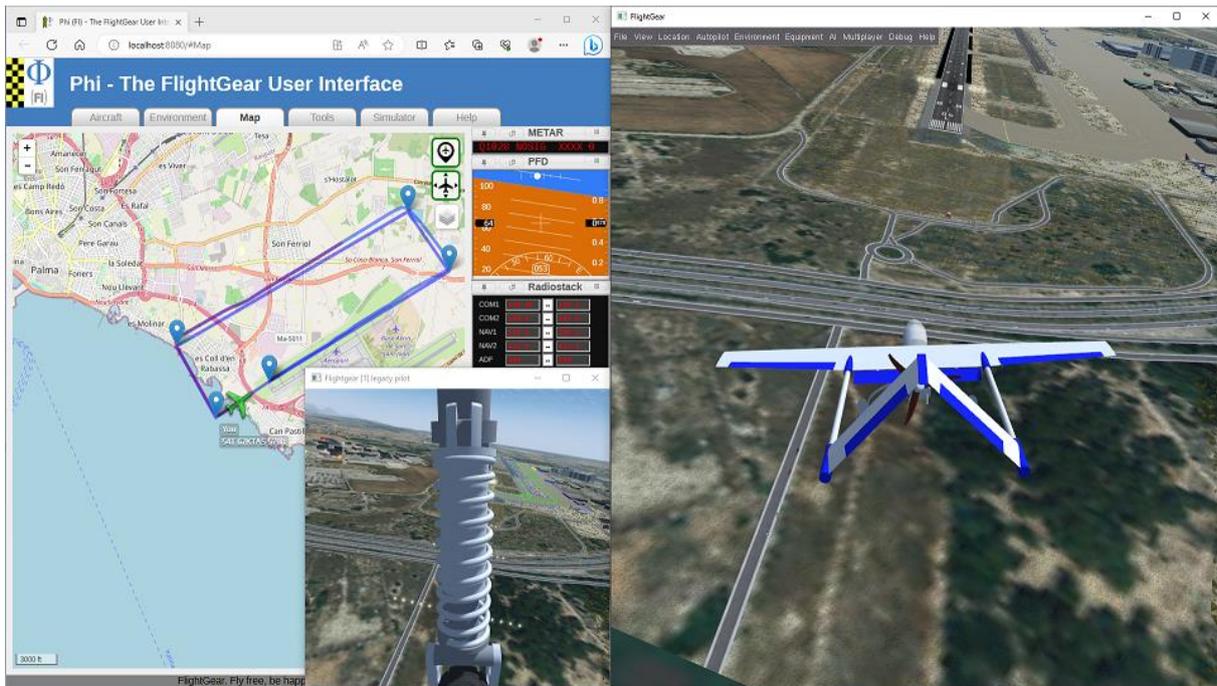


Fig. 8. Final approach, LEPA 06L (right), payload view (middle), and Phi user interface

Conclusion

The proposed simulator concept has somewhat been inspired by SimLat C-Star, [10] unmanned aerial training system. It is expected to reproduce C-Star functions and abilities as close as possible. A main advantage of suggested solution is low, next to zero cost. It takes some CAD and programming skills to implement the idea, though. Given perseverance with employed software (see disclaimer), a variety of mission types and UAV models could be implemented. Unlike most commercial flight simulators, FlightGear gives the developer ways and means to put various project designs into effect thanks to “a powerful scripting language and an open architecture,” [11].

Disclaimer

During project development, following software bundles have been used:

- Trial versions of Autodesk Inventor [12] and Invis AC3D [13] to create a 3D CAD model
- Free copies of SciLab, [14] and GNU Octave, [15] to design the longitudinal control loop
- Trial version of SimWorks, [16] an OpenFoam GUI to compute basic aerodynamic quantities

In memoriam

Let’s remember our colleague Plamen Kostov who passed suddenly away about a month ago.

References:

1. <https://talkai.info/chat/>
2. <https://grabcad.com/library/flight-of-fancy-1>
3. <https://grabcad.com/library/high-lift-mechanism-1>
4. <https://www.dlengine.com/en/rcengine/dle60/>
5. <https://www.e-consystems.com/blog/camera/technology/key-parameters-you-should-know-before-buying-near-infrared-nir-camera/>
6. Blakelock, J., Automatic control of Aircraft and Missiles, 2nd ed., John Wiley & Sons, 1991, ISBN 0-471-50651-6, p.p. 25, 36, 46
7. Nelson, R., Flight Stability and Automatic Control, 2nd ed., McGraw-Hill, 1998, ISBN 0-07-115838-3, p. 125
8. <https://wiki.flightgear.org/Phi>
9. https://wiki.flightgear.org/CompositeViewer_support#CompositeViewer
10. <https://www.simlat.com/products>
11. <https://sourceforge.net/projects/flightgear/>
12. <https://www.autodesk.com/products/inventor/overview>
13. <https://www.invis.com/>
14. <https://www.scilab.org/>
15. <https://octave.org/>
16. <https://www.idealsimulations.com/simworks-free-cfd-software/>