# LQR controller apply to Reach Civilian Aircraft Model

R12921080 徐昀湞

## I. MOTIVATION

The application of LQR to aircraft altitude stabilization presents a promising avenue for enhancing aviation safety and performance. The lessons learned from the Air France 447 disaster underscore the urgency and importance of this research. By developing more robust, efficient, and fault-tolerant control systems, we can significantly reduce the risk of similar accidents and improve the overall safety of air travel.

## II. INTRODUCTION

In this final project, I trying to model an aircraft and design a LQR controller to stabilize the aircraft. The aircraft model will be develop from aerodynamic principle. This analysis will be construct in platform. In this paper. All parameter is from [2].

## III. AIRCRAFT NON-LINEAR DYNAMICS

### A. Equations of Motion (EOMs)

Equations of motion (EOMs) assume that bodies are rigid. According to Newton's second law, the total external forces acting on a body are equal to how its linear momentum changes over time. Euler added to this by considering the total moments acting on the body, which show how its angular momentum changes over time. These equations are given as following:

$$\sum F = \frac{d}{dt}(mV) = m\frac{dV}{dt} = m\begin{bmatrix}\dot{u}\\ \dot{v}\\ \dot{w}\end{bmatrix} \tag{1}$$

$$\sum M = \frac{d}{dt}(H) = \frac{d(I\omega)}{dt} = \begin{bmatrix}I_x & I_{xy} & I_{xz}\\ I_{yx} & I_y & I_{yz}\\ I_{zx} & I_{zy} & I_z\end{bmatrix}\begin{bmatrix}\dot{p}\\ \dot{q}\\ \dot{r}\end{bmatrix} \tag{2}$$

When an airplane rotates in a non-rotating reference frame, its moments and inertia change with time. The airplane's body frame is always rotating at a certain speed compared to the fixed frame. Calculating changes in this rotating frame can be challenging. To make it easier, we use axes fixed to the airplane's body to calculate these changes. Then transfer to inertial frame. The transformation equation is show in Equation (3 )

$$\dot{\overline{V}}|_i = \dot{\overline{V}}|_b + \overline{\omega} \times \dot{\overline{V}}|_b \tag{3}$$

Inserting the value of $\dot{\overline{V}}|_i$ from equation (3) into equation (1), we obtain:

$$\sum \overline{F} = m(\dot{\overline{V}}|_b + \bar{\omega} \times \dot{\overline{V}}|_b) \tag{4}$$

where as $\overline{\omega} \times \overline{V}|_b = \begin{bmatrix}\hat{i} & \hat{j} & \hat{k}\\ p & q & r\\ u & v & w\end{bmatrix} = (qw - rv)\hat{i} - (pw - ru)\hat{j} + (pv - qu)\hat{k}$ so we get

$$\begin{cases}F_x = m(\dot{u} + qw - rv)\\ F_y = m(\dot{v} + ru - pw)\\ F_z = m(\dot{w} + pv - qu)\end{cases} \tag{5}$$

Re-arrange equation(5), we obtain:

$$\dot{\overline{V}}|_b = \frac{1}{m}\sum \overline{F} - \overline{\omega} \times \overline{V}|_b \tag{6}$$

or rewrite the differential equation into matrix form.

$$\begin{bmatrix}\dot{u}\\ \dot{v}\\ \dot{w}\end{bmatrix} = \frac{1}{m}\sum \overline{F} - \overline{\omega} \times \begin{bmatrix}u\\ v\\ w\end{bmatrix} \tag{7}$$

Similar with equation (7), the summation of moments in an inertial frame is represented in equation(5). Again, using the Coriolis equation, we can convert the moments into body frame equation (8). The result is the rotational velocity equation(9)

$$\sum \overline{M} = \frac{d}{dt}|_b(I\overline{\omega}) + \omega \times I\overline{\omega} \tag{8}$$

$$\begin{cases}M_x = I_x\dot{p} - I_{xz}\dot{r} + (I_z - I_y)qr - I_{xz}pq\\ M_y = I_y\dot{q} - I_{xz}(p^2 - q^2) + (I_x - I_z)rp\\ M_z = I_xz\dot{p} - I_z\dot{r} + (I_y - I_x)pq + I_{xz}rq\end{cases} \tag{9}$$

$$\dot{\overline{\omega}}|_b = I^{-1}(\sum \overline{M} - \overline{\omega} \times I\overline{\omega}) \tag{10}$$

or we can rewrite equation into explicit first order form.

$$\begin{bmatrix}\dot{p}\\ \dot{q}\\ \dot{r}\end{bmatrix} = I^{-1}(\sum \overline{M} - \overline{\omega} \times I\overline{\omega}) \tag{11}$$

We are also interest about angular and translational position about respect to inertial axis. The angular position is derived in equation (12). By using transformation matrix. We can easily get vector between inertial frame and body frame. (Refer to figure(1))

$$\begin{bmatrix}\dot{\phi}\\ \dot{\theta}\\ \dot{\psi}\end{bmatrix} = \begin{bmatrix}1 & sin\phi tan\theta & cos\phi tan\theta\\ 0 & cos\phi & -sin\phi\\ 0 & sin\phi/cos\theta & cos\phi/cos\theta\end{bmatrix}\overline{\omega} \tag{12}$$

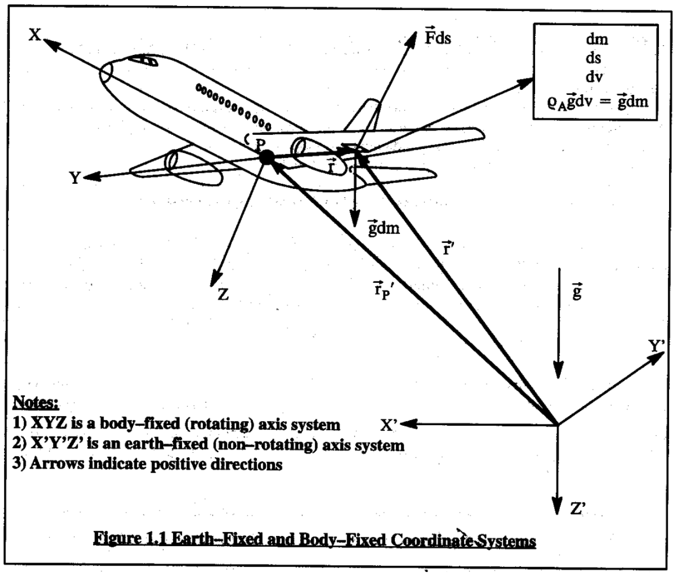


Fig. 1: Earth-Fixed and Body-Fixed Coordinate Systems

Similar with equation(12). Transfer the body velocity to inertial frame.

$$\overline{V}_i = R_{b/i}\overline{V}_b \tag{13}$$

Where $R_{b/i}$ is

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & C\phi & S\phi \\ 0 & -S\phi & C\psi \end{bmatrix} \begin{bmatrix} C\theta & 0 & -S\theta \\ 0 & 1 & 0 \\ S\theta & 0 & C\theta \end{bmatrix} \begin{bmatrix} C\psi & S\psi & 0 \\ -S\phi & C\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{14}$$

So we can get position vector.

$$\begin{bmatrix} \frac{dX}{dt} \\ \frac{dY}{dt} \\ \frac{dZ}{dt} \end{bmatrix} = R_{b/i} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{15}$$

Later on we can combine all equations into a state space model.

### B. External Force

In order to solve equation(7). Calculating external force applied to aircraft is required. External force is consist of gravitational force, aerodynamics force, and propulsive force acting on the aircraft. Notice that equation(16) is respect to body frame.

$$\sum \overline{F} = \overline{F}_{gavaity}|_b + \overline{F}_{engine}|_b + \overline{F}_{aero}|_b \tag{16}$$

*1) Gravitational Force:* In an inertial force, weight always acting downward force. Than we transfer gravitational force into body frame in order to reduce complexity of calculation.

$$\overline{F}_{gravity}|_i = \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \tag{17}$$

$$\overline{F}_{gravity}|_b = R_{i/v}|_i\overline{F}_{gravity}|_i = mg \begin{bmatrix} -S\theta \\ C\theta S\phi \\ C\theta S\theta \end{bmatrix} \tag{18}$$

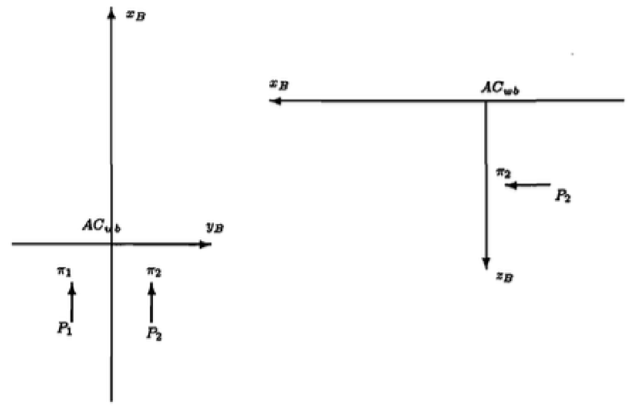


Fig. 2: Earth-Fixed and Body-Fixed Coordinate Systems

*2) Propulsive Force:* The RCAM is similar with Boeing 757-200, twin engines airplane. It is assume thrust produced by engines in line with x-axis in body frame.( refer to figure(2)) We can express thrust force as the following equation.

$$\overline{F}_{engine}|_b = \begin{bmatrix} F_{engin1} + F_{engin2} \\ 0 \\ 0 \end{bmatrix} \tag{19}$$

*3) Aerodynamics Force:* There are three forces acting along the axes of the body frame. These are normal, axial, and side forces. Generally, the forces are captured in the wind axes by definition and need to be transformed back to body axes using a transformation matrix.

$$\overline{F}_{aero}|_b = \begin{bmatrix} F_A \\ F_S \\ F_N \end{bmatrix} = \begin{bmatrix} C\beta & -S\beta & 0 \\ S\beta & C\beta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\alpha & 0 & S\alpha \\ 0 & 1 & 0 \\ S\alpha & 0 & C\alpha \end{bmatrix} \begin{bmatrix} -D \\ SF \\ -L \end{bmatrix} \tag{20}$$

Where $C\beta = \cos(\beta)$, $S\beta = \sin(\beta)$, $C\alpha = \cos(\alpha)$,$S\alpha = \sin(\alpha)$.

Now we know all parameters except $D$, $SF$,$L$.

### C. External Moments

External moments in a body frame are a combination of moments generated by gravitational force, aerodynamics force, and propulsive forces. Notice that equation (22) is respect to body frame.

$$\sum \overline{M} = \overline{M}_{gavaity} + \overline{M}_{engine} + \overline{M}_{areo} \tag{21}$$

As the weight is acting at the center of gravity (CG), gravitational force will not generate any moment about the CG. The propulsive and aerodynamics forces are multiplied with the moment arm (distance of application of force from CG in the respective axes) to obtain the moments. For propulsive forces, the center of the engine was taken as a reference, and for aerodynamics forces, the quarter chord location was taken as a reference for the calculation of moment arms.

$$\sum \overline{M} = \sum_{i=1}^{2} \begin{bmatrix} (x_{CG} - x_{Ti})F_{propulsive} \\ (y_{CG} - y_{Ti})F_{propulsive} \\ (z_{CG} - z_{Ti})F_{propulsive} \end{bmatrix} + \begin{bmatrix} \mathcal{L} \\ M \\ N \end{bmatrix} \tag{22}$$

### D. Aerodynamics Forces

Aerodynamic forces depend on various factors like speeds, rotations, control inputs, etc. These forces can be described using all motion variables, but typically only the most important ones are considered, while the others are ignored.

$$\begin{cases} L = C_L QS \\ D = C_D QS \\ SF = C_\gamma \\ \mathcal{L} = C_l QSb \\ M = C_m QSb \\ N = C_n QSb \end{cases} \tag{23}$$

$$\begin{cases} \alpha = tan^{-1}(\frac{w}{u}) = \frac{w}{u} \\ \beta = tan^{-1}(\frac{w}{V}) = \frac{w}{V} \\ V = \sqrt{u^2+v^2+w^2} \end{cases} \tag{24}$$

Where $Q = \frac{1}{2}^2$ ans S is main wing area. And $\alpha$ is angle of attack. And we suppose $\rho$ is 1.225kg/$m^3$. In appendix. There is a detail explain with MATLAB code.

### E. State Space Model

Represent equation(7), equation(11), equation(12) and equation(15) in to one state space matrix.

$$\dot{\overline{X}} = \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \\ \dot{p} \\ \dot{q} \\ \dot{r} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \\ dX/dt \\ dY/dt \\ dZ/dt \end{bmatrix} = \begin{bmatrix} \frac{1}{m}\sum \overline{F} - \overline{\omega} \times \overline{V}|b \\ I^(-1)(\sum \overline{M} - \overline{\omega} \times I\overline{\omega}) \\ H(\Phi)\overline{\omega} \\ R_{b/i}\overline{V}_b \end{bmatrix} \tag{25}$$

Decompose state space model into state variables matrix and control variables,

$$\dot{\overline{X}} = f(\overline{X}, \overline{U})\overline{X} + f(\overline{X}, \overline{U})\overline{U} \tag{26}$$

$$\overline{X} = \begin{bmatrix} u \\ v \\ w \\ p \\ q \\ r \\ \phi \\ \theta \\ \psi \\ X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_10 \\ x_11 \\ x_12 \end{bmatrix} ; \overline{U} = \begin{bmatrix} \delta_{areliron} \\ \delta_{elevator} \\ \delta_{rudder} \\ \delta_{throttle1} \\ \delta_{throttle1} \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{bmatrix} \tag{27}$$

## IV. Lineralize Non-linear Model

Since moments and areodynamics are non-linear part in this 6-DOF model. To design a linear controller like LQR. Linearlize the model is necessary process. In this case we assume the aircraft is cursing in 180kt/h.

| | |
|---|---|
| $u_0$ | 80 m/s |
| $v_0$ | 0 m/s |
| $w_0$ | 3.8 m/s |
| $p_0$ | 0 rad/s |
| $q_0$ | 0 rad/s |
| $r_0$ | 0 rad/s |
| $\phi_0$ | 0 deg |
| $\theta_0$ | 0.0604 deg |
| $\psi_0$ | 0 deg |

After we determining equilibrium point of this model. Then find Jacobin matrix of state space model. equation(25)

$$\dot{X} = \frac{\partial f(\overline{X}, \overline{U})}{\partial X}|_{X=X_0}\overline{X} + f(\overline{X_0}, \overline{U}) \tag{28}$$

**Notice that there are two if statements in RCAM model. It can't be linearlize due to if statements. Please comment those code when you are doing linearlize process.** In general state space model. We have got A matrix and B matrix already. Choose C as identity matrix and D is null matrix. So we can get comprehensive state space matrix.

$$\begin{gathered} \dot{x} = [\frac{\partial f(\overline{X}, \overline{U})}{\partial X}|_{X=X_0}\overline{X}]x + f(\overline{X_0}, 0)u \\ y = x \end{gathered} \tag{29}$$

## V. Control Design

A, B, C, D matrix are determined now. Then we aim to design a controller to let aircraft stable. We hope aircraft can fly in a straight line. So set Q, R as following:

$$Q = \begin{bmatrix} 0 &&&&&&&&& \\ & 0 &&&&&&&& \\ && 10 &&&&&&& \\ &&& 10 &&&&&& \\ &&&& 10 &&&&& \\ &&&&& 10 &&&& \\ &&&&&& 10 &&& \\ &&&&&&& 0 && \\ &&&&&&&& 0 & \\ &&&&&&&&& 0 \end{bmatrix} \tag{30}$$

$$R = \begin{bmatrix} 1 &&&& \\ & 1 &&& \\ && 1 && \\ &&& 100 & \\ &&&& 100 \end{bmatrix} \tag{31}$$

To let aircraft fly in straight line. We let static angles of aircraft and forward velocity free. The thrust force is determined by pilot since this controller doesn't perform airspeed management. So we set penalty value of thrust to 100.

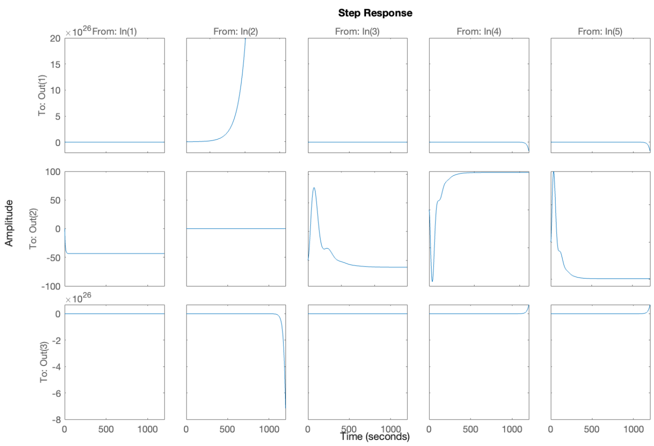


Fig. 3: Step respond without controller

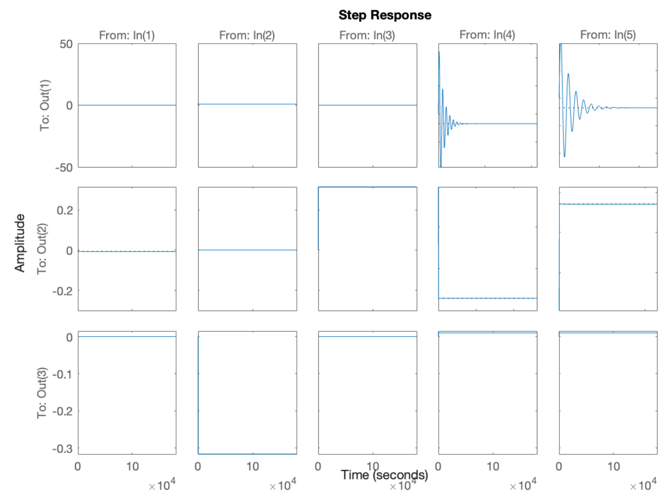


Fig. 4: Step respond with controller

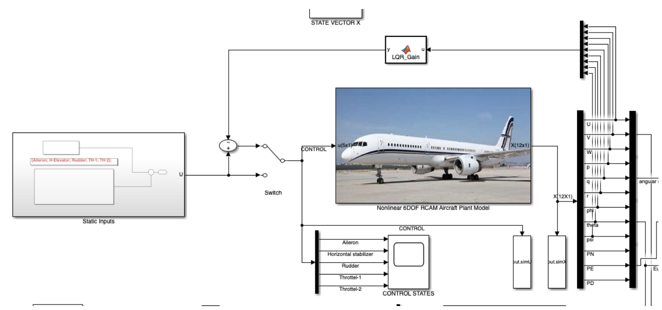


Fig. 5: Step respond with controller

### A. LQR control

The model in the performance index is used the feedback control. By defining performance index we can set matrix Q and matrix R to penalty the terms we want it goes to zero. So here is the stander form

$$\dot{x} = Ax + Bu \tag{32}$$

Than we define output matrix is

$$y = Cx + Du \tag{33}$$

Than we define performance index as following

$$V = \int_0^\infty \{x^T Q x + u^T R u\} dt \tag{34}$$

Because it is a infinite time equation. So we can solve Riccait equation easily. Get $P(t)$ by solving infinite time Riccati equation $A^T P + PA + Q - PBR^{-1}B^T P = 0$. In this paper, lqr function in MATLAB is utilized. Then we get feedback control is $u = -R^{-1}G^T P(t)x$.

### B. Step Respond

By using ”step” command in Matlab to check step respond. Obviously, step respond of all output are stable with controller. All output converge to 0 (figure(4)).

And we can also calculate eigenvalue of matrix with feedback (equation (36)) and without feedback (equation (35)). We can know that system is more stabler after adding a feedback control.

$$\begin{bmatrix} 0.04655 \\ 0 \\ -0.07294 \\ -0.13963 \\ -0.27173 + 0.70532\,\mathrm{i} \\ -0.27173 - 0.70532\,\mathrm{i} \\ -0.8518 + 1.45763\,\mathrm{i} \\ -0.8518 - 1.45763\,\mathrm{i} \\ -1.27591 \end{bmatrix} \tag{35}$$

$$\begin{bmatrix} 0.0000 + 0.0000\,\mathrm{i} \\ -7.8238 + 6.2816\,\mathrm{i} \\ -7.8238 - 6.2816\,\mathrm{i} \\ -2.9208 + 0.0000\,\mathrm{i} \\ -0.1170 + 0.0000\,\mathrm{i} \\ -21.0505 + 14.1967\,\mathrm{i} \\ -21.0505 - 14.1967\,\mathrm{i} \\ -0.0190 + 0.165\,\mathrm{i} \\ -0.0190 - 0.165\,\mathrm{i} \end{bmatrix} \tag{36}$$

## VI. Simulation

After applying LQR controller in simulink model.(ref figure(5)), Please note that there is a switch taht can help user to engage LQR control and disengage.

### A. Preparing

Before we start simulation. Install FlightGear is required. In the simulink file. There is a blue block named ”Generate Run Script”. Please click that icon and set basic information. So it can generate initial condition of simulation. Notice than the RCAM model is similar with Boeing 757-200. But due to model limitation. This paper only demo model in Cessna 172. Of course you can install Boeing 757-200 model into your simulation environment.

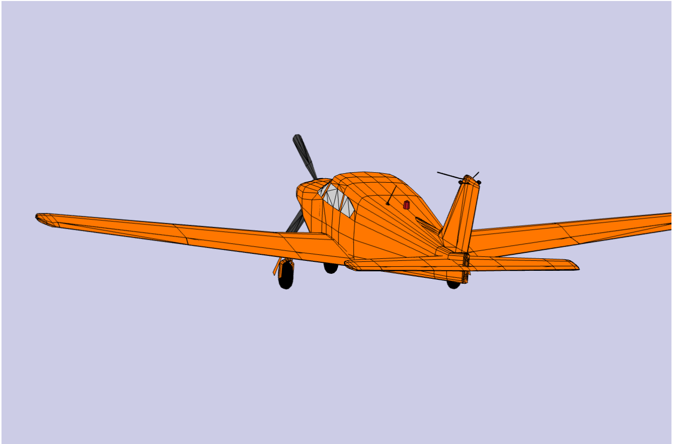

Fig. 6: MATLAB simulation with close loop

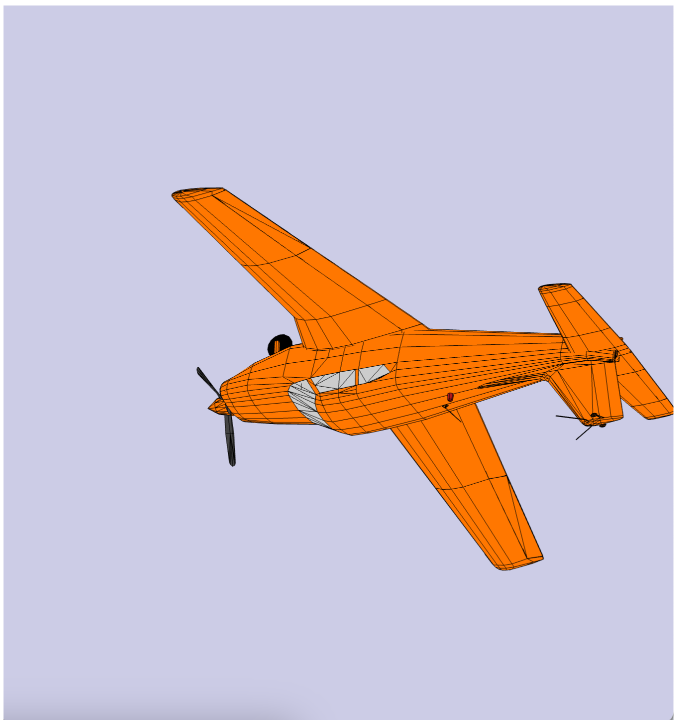

Fig. 7: MATLAB simulation without close loop

### *B. Simulation*

In this part we will use FlightGear to simulate behavior of aircraft. FlightGear provide udp protocol which can let MATLAB transmit data to. While full thrust apply on aircraft at t0 in open loop situation. But same situation apply to aircraft with LQR controller. Aircraft is stabler. We can know that in figure(6), figure(7), figure(8) and figure(9).

## VII. Conclusion

In this project, I managed to implement a controller which stabilizes the aircraft.

In the course of designing the controller, I learn how to derive close loop gain from performance index via Riccati equation. And I apply LQR technique on aircraft stabilize.

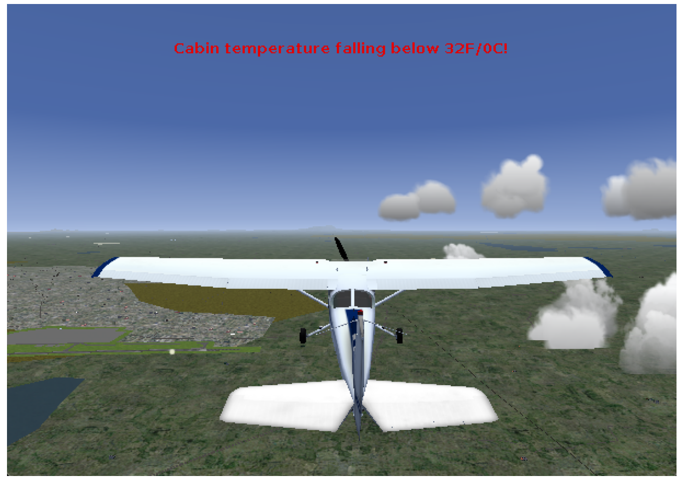


Fig. 8: FlightGear simulation with close loop

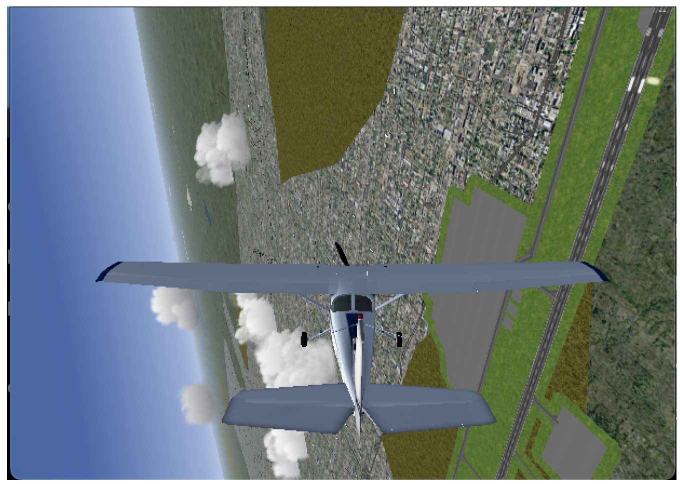

Fig. 9: FlightGear simulation without close loop

## VIII. References

## References

[1] John D. Anderson, Jr. (2017) Fundamentals of Aerodynamics.
[2] GARTEU. (1995) Robust Flight Control Design Challenge Problem Formulation and Manual: the Research Civil Aircraft Model (RCAM).
[3] M.V. Cook. (2007) Flight Dynamics Principles