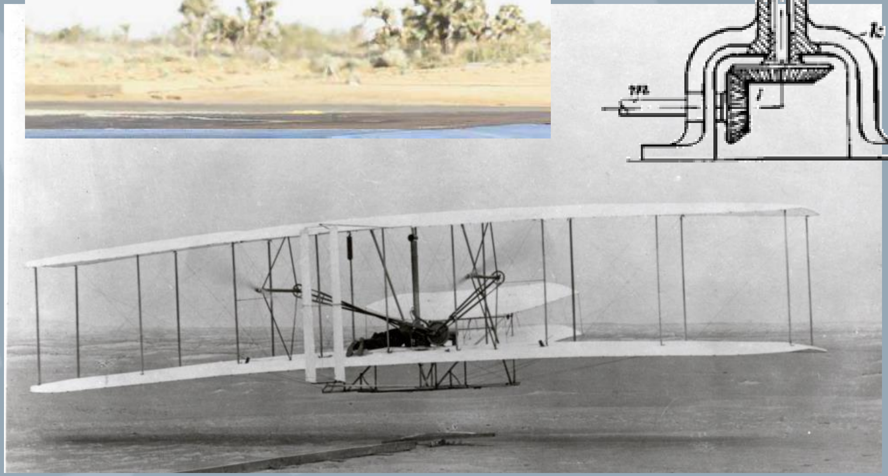
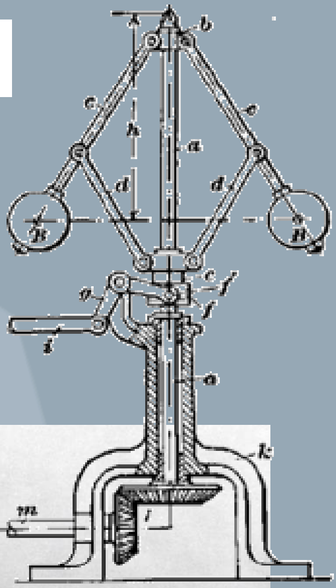
MASI

Master in Aircraft
Systems Integration

Introduction to Aircraft Control

FCL Design with Matlab





Alberto P. Martínez Borja
(AIRBUS- UC3M)

Outline

- Stability Augmentation Systems (SAS)
  - Phugoid Suppression
  - Short Period Stability Augmentation
  - Longitudinal Pole Placement
- Control Augmentation Systems (CAS)
  - Speed Controller
  - Altitude Controller

Phugoid Suppression

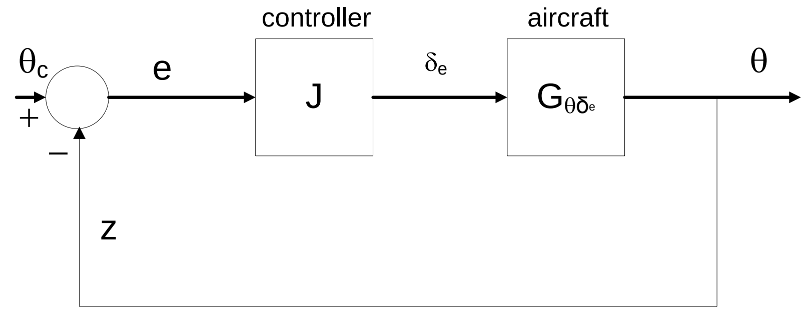
- Stability Augmentation Systems (SAS)
  - Phugoid Suppression
  - Short Period Stability Augmentation
  - Longitudinal Pole Placement

Phugoid Suppression

Phugoid Suppression: Pitch Attitude Controller

- The characteristic lightly damped, low-frequency oscillations in speed, pitch attitude, and altitude are not in evidence in actual flight; the pilot (human or automatic) effectively suppresses them, maintaining flight at more or less constant speed and height.
- In principle it can be achieved by using feedback signals derived from any one or a combination of pitch attitude θ, altitude h, speed v, and their derivatives.
- Since the Phugoid oscillation cannot occur if the pitch angle θ is not allowed to change (except when commanded to), a pitch-attitude-hold feature in the autopilot would be expected to suppress the Phugoid.
- Pitch attitude is readily available from either the real horizon (human pilot) or the vertical gyro (autopilot).

$$G_{\theta\delta_e} = \frac{-1.158\,s^2 - 0.3545\,s - 0.003873}{s^4 + 0.7505\,s^3 + 0.9355\,s^2 + 0.009463\,s + 0.004196}$$



$$\frac{\theta}{\theta_c} = \frac{G_{\theta\delta_e} J(s)}{1 + G_{\theta\delta_e} J(s)}$$

$$J(s) = k_p + \frac{k_i}{s} + k_d s$$

Short Period Stability Augmentation

- Stability Augmentation Systems (SAS)
  - Phugoid Suppression
  - Short Period Stability Augmentation
  - Longitudinal Pole Placement

Short Period Stability Augmentation

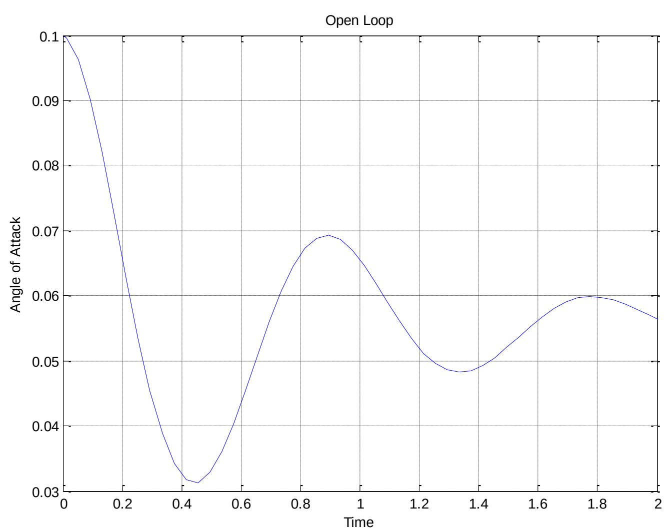
$$\Delta\dot{\bar{x}}_{SP} = \begin{bmatrix} \Delta\dot{q} \\ \Delta\dot{\alpha} \end{bmatrix} = \begin{bmatrix} M_q & M_\alpha \\ 1 & -\dfrac{L_\alpha}{V_N} \end{bmatrix} \begin{bmatrix} \Delta q \\ \Delta\alpha \end{bmatrix} + \begin{bmatrix} M_{\delta_e} \\ -\dfrac{L_{\delta_e}}{V_N} \end{bmatrix} \Delta\delta_e$$

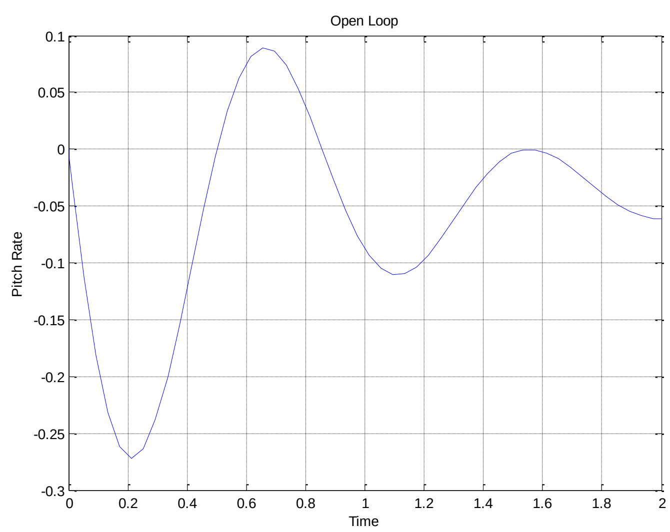
$$\Delta\dot{\bar{x}}_{SP} = \begin{bmatrix} \Delta\dot{q} \\ \Delta\dot{\alpha} \end{bmatrix} = \begin{bmatrix} -1.47961 & -49.4425 \\ 1 & -1.16668 \end{bmatrix} \begin{bmatrix} \Delta q \\ \Delta\alpha \end{bmatrix} + \begin{bmatrix} -22.4739 \\ -0.121741 \end{bmatrix} \Delta\delta_e$$

ω_{sp} = 7.15 [rad/s]
ξ_{sp} = 0.185

Open Loop Short Period eigenvalues: -1.32315 ±7.0298i

Effect of a 0.1 rad perturbation in Angle of Attack (α)





Longitudinal Pole Placement

- Stability Augmentation Systems (SAS)
  - Phugoid Suppression
  - Short Period Stability Augmentation
  - Longitudinal Pole Placement

Longitudinal Pole Placement

$$\Delta\dot{\vec{x}}_{SP} = \begin{bmatrix} \Delta\dot{q} \\ \Delta\dot{\alpha} \end{bmatrix} = \begin{bmatrix} M_q & M_\alpha \\ 1 & -\dfrac{L_\alpha}{V_N} \end{bmatrix} \begin{bmatrix} \Delta q \\ \Delta\alpha \end{bmatrix} + \begin{bmatrix} M_{\delta_e} \\ -\dfrac{L_{\delta_e}}{V_N} \end{bmatrix} \Delta\delta_e$$

$$\Delta\dot{\vec{x}}_{SP} = \begin{bmatrix} \Delta\dot{q} \\ \Delta\dot{\alpha} \end{bmatrix} = \begin{bmatrix} -1.47961 & -49.4425 \\ 1 & -1.16668 \end{bmatrix} \begin{bmatrix} \Delta q \\ \Delta\alpha \end{bmatrix} + \begin{bmatrix} -22.4739 \\ -0.121741 \end{bmatrix} \Delta\delta_e$$

```
Asp=[-1.47961 -49.4425;1 -1.16668]
Bsp=[-22.4739;-0.121741]
```

ω_{sp} = 7.15 [rad/s]
ξ_{sp} = 0.185

Desired poles with ω_{sp} = 3 [rad/s] and ξ_{sp} = 0.6 ⇒ $s = -1.8 \pm 2.4i$

```
Ksp=place(Asp,Bsp,[roots([1 2*0.6*3 3^2])])
```

k_q = -0.0528
k_α = 1.9085

Closed Loop system

```
Acl=Asp-Bsp*Ksp
```

$$\Delta\delta_e = \begin{bmatrix} -0.0264 & -2.3463 \end{bmatrix} \begin{bmatrix} \Delta q \\ \Delta\alpha \end{bmatrix}$$

Longitudinal Pole Placement

Alternative formulation - Open Loop eigenvalues

```
opeig=eig(Asp)
```

Desired poles with ω_{sp} = 3 [rad/s] and ξ_{sp} = 0.6 $\Rightarrow$ $s = -1.8 \pm 2.4i$
Closed Loop system

```
sys=tf(1,[1 -2*real(opeig(1)) real(opeig(1))^2+imag(opeig(1))^2])*
tf(1, [1 2*0.0033 0.0672^2+0.0033^2])
[A,B,C,D]=ssdata(sys)
```

Full model: Only Short Period poles are changed, maintaining Phugoid:

$$s = -1.8 \pm 2.4i,\ -0.0033 \pm 0.0672i$$

```
ev=eig(A)
Plist=[roots([1 2*.6*3 3^2]) ev([3 4],1)]
K1=place(A,B(:,1),Plist)
```

$$\Delta\delta_e = \begin{bmatrix} 0.9538 & -5.2689 & -0.0685 & -0.1908 \end{bmatrix} \begin{bmatrix} V \\ \gamma \\ q \\ \alpha \end{bmatrix}$$

$k_V = 0.9537$ $k_\gamma = -5.2703$
$k_q = -0.0685$ $k_\alpha = -0.1909$

Speed Controller

- Control Augmentation Systems (CAS)
  - Speed Controller
  - Altitude Controller

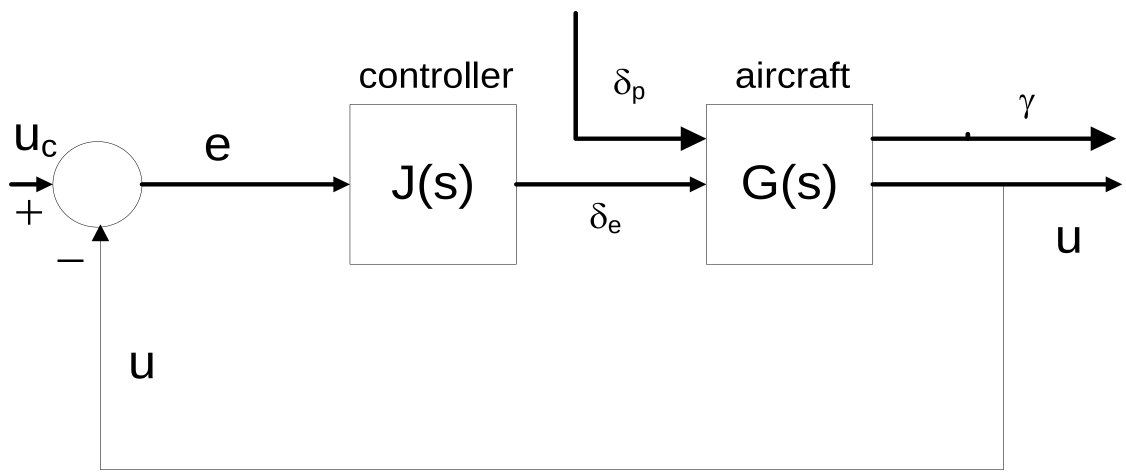
Speed Controller

Justification

- Phugoid presence is not only evident in transients to steady states but also during maneuvers.
- For instance, opening the throttle for changing from level to climbing flight leads to this situation and takes typically 10 minutes to damp.
- In reality, it is suppressed by the pilot.
- Alternative option based on speed error is available.
- Both elevator (δ_e) and throttle (δ_p) influence the speed:
  - Throttle in the short-term only
  - Elevator in the long term, for changing the steady state speed level.
- A sophisticated control based on both actuators could be envisaged.
- However, for suppressing Phugoid in maneuvers, only elevator is needed.

Speed Controller

- Commanded speed (u_c) with actual speed (u) as feedback.
- Output vector: Speed (u) and Flight Path Angle (γ)
- Speed does not change appreciably in the Short Period mode, only Phugoid is considered. **(Phugoid Approximation)**



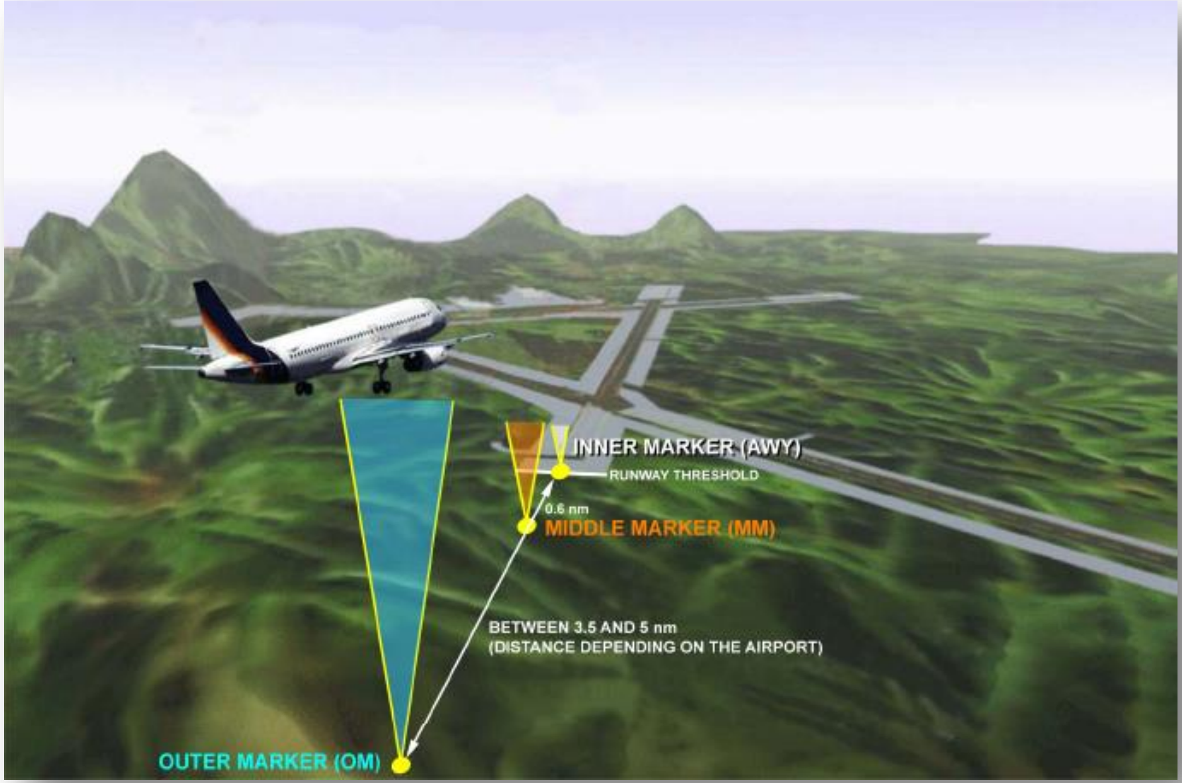
$$G(s) = \begin{bmatrix} G_{\gamma\delta_p} & G_{\gamma\delta_e} \\ G_{u\delta_p} & G_{u\delta_e} \end{bmatrix}$$

- Both elevator and throttle influence the speed, but that the short and long term effects of each of these controls are quite different. The throttle principally affects the speed only in the short term. For a change of steady-state speed, the elevator must be used.

Altitude Controller

❑ Control Augmentation Systems (CAS)

- Speed Controller
- Altitude Controller



Altitude Controller

- One of the most important problems in the control of flight path is that of following a prescribed line in space, as defined for example by a radio beacon, or when the airplane flies down the ILS glide slope.
- We illustrate an Altitude Controller that also incorporates control of speed.
- We make the system model more realistic by including first-order lag elements for the two controls: that for the elevator is mainly associated with its servo actuator (time constant τ_e=0.1s); and that for the throttle with the relatively long time lag inherent in the build up of thrust of a jet engine following a sudden movement of the throttle (time constant τ_p=3.5s)
- Another feature that is incorporated to add realism to the example is a thrust limiter.
- Because transport aircraft inherently respond slowly to changes in thrust, the gains chosen to give satisfactory response for very small perturbations in speed will lead to a demand for thrust outside the engine envelope for larger speed errors. We have therefore included a nonlinear feature that limits the thrust to the range $0 \leq T \leq 1.1To$.
- This contains the implicit assumptions (quite arbitrary) that the airplane, flying near its ceiling, has 10% additional thrust available, and that idling engines correspond to zero thrust.

Altitude Controller

Horizontal Steady Flight

- Equation: mV'=T-D
- Small perturbation equation around equilibrium points:

$$V = V_o + \Delta V$$

$$T = T_o + T_V \Delta V; \quad T_V = \frac{\partial T}{\partial V}$$

$$D = D_o + D_V \Delta V; \quad D_V = \frac{\partial D}{\partial V}$$

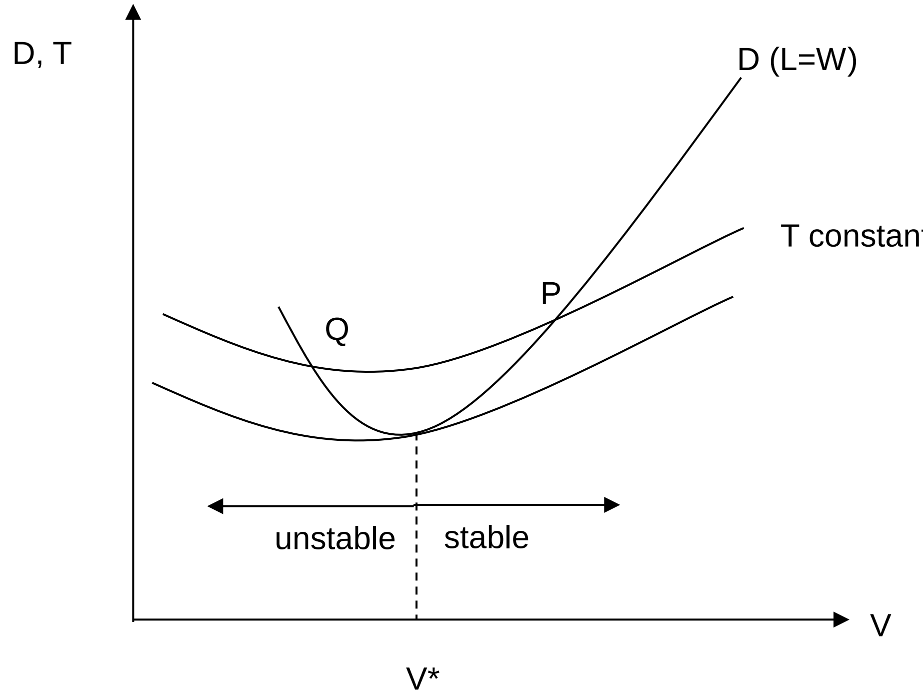
$$m\Delta\dot{V} = (T_V - D_V)\Delta V$$

$$\Delta V = a e^{\lambda t}$$

$$\lambda = \frac{(T_V - D_V)}{m}$$

$$D_V > T_V; \quad stable$$

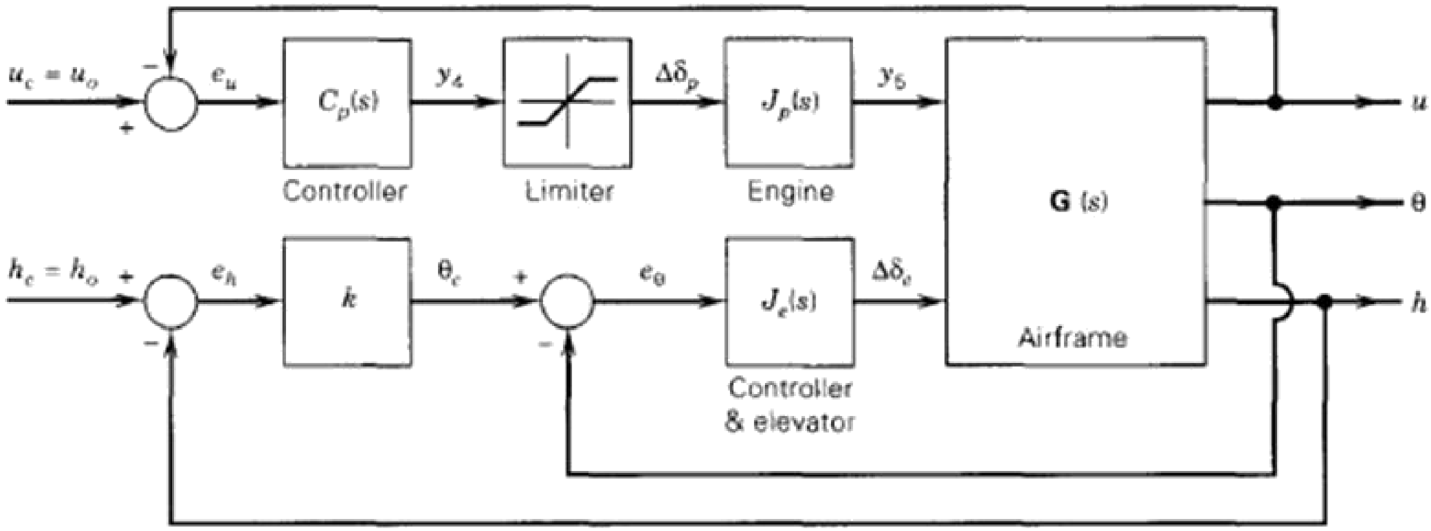
$$D_V < T_V; \quad unstable$$



- Same applies to other flight linear paths
- At low velocity, a control based on elevator alone to follow the path with stability is not enough

Altitude Controller

- The system block diagram is shown below:

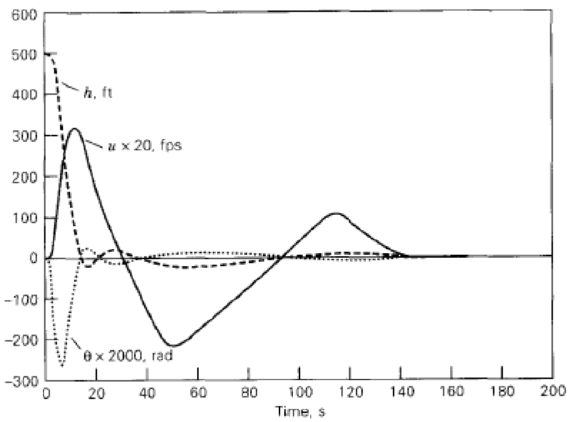


Altitude-hold controller.

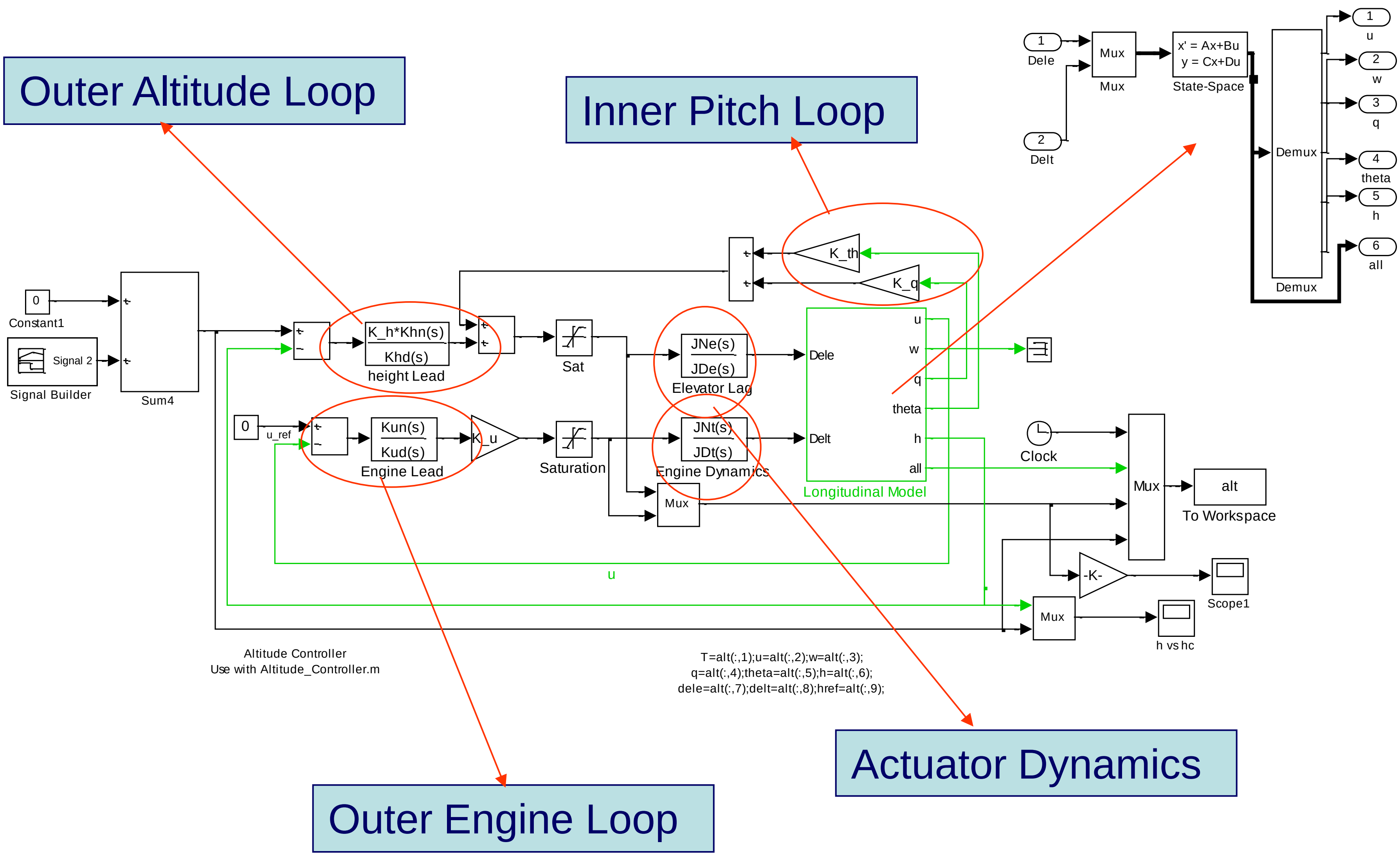
- The commanded speed and altitude are the reference values u_c, and h_c, so that the two corresponding error signals are $-\Delta u$ and $-\Delta h$.
- The inner loop for θ is that previously studied in the Speed Controller, with the $J_e(s)$ modified to account for the elevator servo actuator.

Altitude Controller

- If there is an initial error in h, say the altitude is too low, then in order to correct it, the airplane's flight path must be deflected upward. This requires an increase in angle of attack α to produce an increase in lift.
- Since short-term changes in θ are effectively changes in α, then much the same result is obtained by using θ as the commanded variable.
- Thus, in summary, the system commands a pitch angle that is proportional to height error and the inner loop uses the elevator to make the pitch angle follow the command.
- While all this is going on the speed will be changing because of both gravity and drag changes. The quickest and most straightforward way of controlling the speed is with the throttle, and the third loop accomplishes that.



Altitude Controller



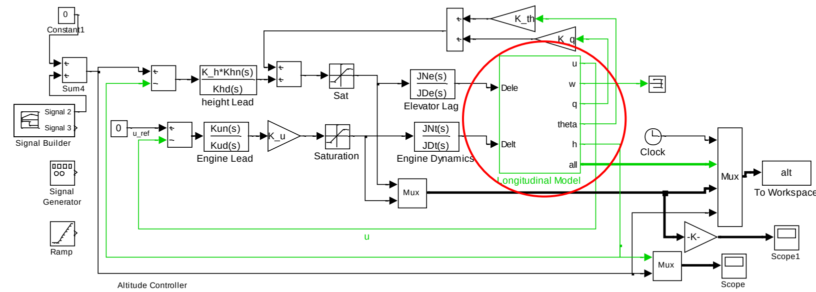
Altitude Controller

Initialization of the system

Xu=-1.982e3
Xw=4.025e3
Zu=-2.595e4
Zw=-9.030e4
Zq=-4.524e5
Zwd=1.909e3
Mu=1.593e4
Mw=-1.563e5
Mq=-1.521e7
Mwd=-1.702e4

g=9.81
theta0=0
S=511
cbar=8.324
U0=235.9
Iyy=.449e8
m=2.83176e6/g
rho=0.3045
Xdp=.3*m*g
Zdp=0
Mdp=0

Xde=-3.818e-6*(1/2*rho*U0^2*S)
Zde=-0.3648*(1/2*rho*U0^2*S)
Mde=-1.444*(1/2*rho*U0^2*S*cbar)




```
A =

   -0.0069    0.0139         0   -9.8100         0
   -0.0905   -0.3149  235.8928         0         0
    0.0004   -0.0034   -0.4282         0         0
         0         0    1.0000         0         0
         0   -1.0000         0  235.9000         0

B =

   -0.0001    2.9430
   -5.5079         0
   -1.1569         0
         0         0
         0         0

C =

     1     0     0     0     0
     0     1     0     0     0
     0     0     1     0     0
     0     0     0     1     0
     0     0     0     0     1
     0     0     0     0     0
     0     0     0     0     0

D =

     0     0
     0     0
     0     0
     0     0
     0     0
     1     0
     0     1
```