



Chapter 5

Linear Design Models

Equations of Motion

$$\dot{p}_n = (\cos \theta \cos \psi)u + (\sin \phi \sin \theta \cos \psi - \cos \phi \sin \psi)v + (\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi)w$$

$$\dot{p}_e = (\cos \theta \sin \psi)u + (\sin \phi \sin \theta \sin \psi + \cos \phi \cos \psi)v + (\cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi)w$$

$$\dot{h} = u \sin \theta - v \sin \phi \cos \theta - w \cos \phi \cos \theta$$

$$\dot{u} = rv - qw - g \sin \theta + \frac{\rho V_a^2 S}{2m} \left[C_X(\alpha) + C_{X_q}(\alpha) \frac{cq}{2V_a} + C_{X_{\delta_e}}(\alpha) \delta_e \right] + \frac{T_p(\delta_t, V_a)}{m}$$

$$\dot{v} = pw - ru + g \cos \theta \sin \phi + \frac{\rho V_a^2 S}{2m} \left[C_{Y_0} + C_{Y_\beta} \beta + C_{Y_p} \frac{bp}{2V_a} + C_{Y_r} \frac{br}{2V_a} + C_{Y_{\delta_a}} \delta_a + C_{Y_{\delta_r}} \delta_r \right]$$

$$\dot{w} = qu - pv + g \cos \theta \cos \phi + \frac{\rho V_a^2 S}{2m} \left[C_Z(\alpha) + C_{Z_q}(\alpha) \frac{cq}{2V_a} + C_{Z_{\delta_e}}(\alpha) \delta_e \right]$$

$$\dot{\phi} = p + q \sin \phi \tan \theta + r \cos \phi \tan \theta$$

$$\dot{\theta} = q \cos \phi - r \sin \phi$$

$$\dot{\psi} = q \sin \phi \sec \theta + r \cos \phi \sec \theta$$

$$\dot{p} = \Gamma_1 pq - \Gamma_2 qr + \frac{1}{2} \rho V_a^2 S b \left[C_{p_0} + C_{p_\beta} \beta + C_{p_p} \frac{bp}{2V_a} + C_{p_r} \frac{br}{2V_a} + C_{p_{\delta_a}} \delta_a + C_{p_{\delta_r}} \delta_r \right] + \Gamma_3 Q_p(\delta_t, V_a)$$

$$\dot{q} = \Gamma_5 pr - \Gamma_6 (p^2 - r^2) + \frac{\rho V_a^2 Sc}{2J_y} \left[C_{m_0} + C_{m_\alpha} \alpha + C_{m_q} \frac{cq}{2V_a} + C_{m_{\delta_e}} \delta_e \right]$$

$$\dot{r} = \Gamma_7 pq - \Gamma_1 qr + \frac{1}{2} \rho V_a^2 S b \left[C_{r_0} + C_{r_\beta} \beta + C_{r_p} \frac{bp}{2V_a} + C_{r_r} \frac{br}{2V_a} + C_{r_{\delta_a}} \delta_a + C_{r_{\delta_r}} \delta_r \right] + \Gamma_4 Q_p(\delta_t, V_a)$$

Equations of Motion

$$C_{p_0} = \Gamma_3 C_{l_0} + \Gamma_4 C_{n_0}$$

$$C_{p_\beta} = \Gamma_3 C_{l_\beta} + \Gamma_4 C_{n_\beta}$$

$$C_{p_p} = \Gamma_3 C_{l_p} + \Gamma_4 C_{n_p}$$

$$C_{p_r} = \Gamma_3 C_{l_r} + \Gamma_4 C_{n_r}$$

$$C_{p_{\delta_a}} = \Gamma_3 C_{l_{\delta_a}} + \Gamma_4 C_{n_{\delta_a}}$$

$$C_{p_{\delta_r}} = \Gamma_3 C_{l_{\delta_r}} + \Gamma_4 C_{n_{\delta_r}}$$

$$C_{r_0} = \Gamma_4 C_{l_0} + \Gamma_8 C_{n_0}$$

$$C_{r_\beta} = \Gamma_4 C_{l_\beta} + \Gamma_8 C_{n_\beta}$$

$$C_{r_p} = \Gamma_4 C_{l_p} + \Gamma_8 C_{n_p}$$

$$C_{r_r} = \Gamma_4 C_{l_r} + \Gamma_8 C_{n_r}$$

$$C_{r_{\delta_a}} = \Gamma_4 C_{l_{\delta_a}} + \Gamma_8 C_{n_{\delta_a}}$$

$$C_{r_{\delta_r}} = \Gamma_4 C_{l_{\delta_r}} + \Gamma_8 C_{n_{\delta_r}}$$

$$C_X(\alpha) \triangleq -C_D(\alpha) \cos \alpha + C_L(\alpha) \sin \alpha$$

$$C_{X_q}(\alpha) \triangleq -C_{D_q} \cos \alpha + C_{L_q} \sin \alpha$$

$$C_{X_{\delta_e}}(\alpha) \triangleq -C_{D_{\delta_e}} \cos \alpha + C_{L_{\delta_e}} \sin \alpha$$

$$C_Z(\alpha) \triangleq -C_D(\alpha) \sin \alpha - C_L(\alpha) \cos \alpha$$

$$C_{Z_q}(\alpha) \triangleq -C_{D_q} \sin \alpha - C_{L_q} \cos \alpha$$

$$C_{Z_{\delta_e}}(\alpha) \triangleq -C_{D_{\delta_e}} \sin \alpha - C_{L_{\delta_e}} \cos \alpha$$

Lift and Drag Models

Nonlinear model with stall effects

$$C_L(\alpha) = (1 - \sigma(\alpha)) [C_{L_0} + C_{L_\alpha} \alpha] + \sigma(\alpha) [2 \operatorname{sign}(\alpha) \sin^2 \alpha \cos \alpha]$$

$$\sigma(\alpha) = \frac{1 + e^{-M(\alpha - \alpha_0)} + e^{M(\alpha + \alpha_0)}}{(1 + e^{-M(\alpha - \alpha_0)}) (1 + e^{M(\alpha + \alpha_0)})}$$

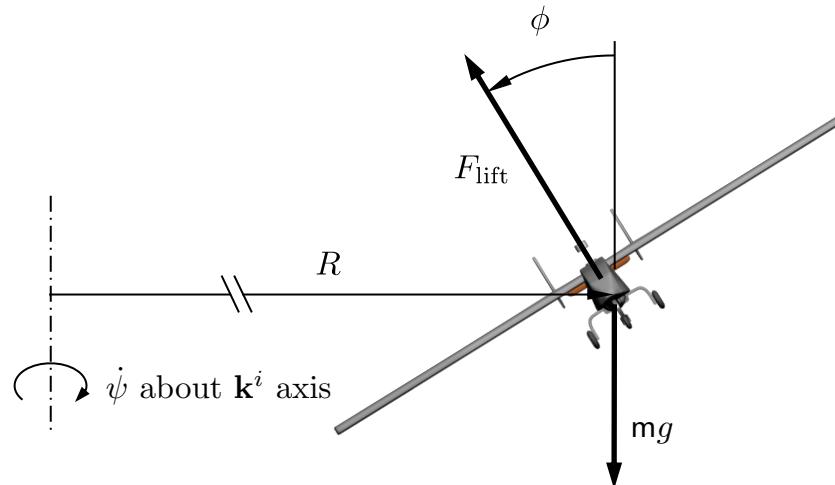
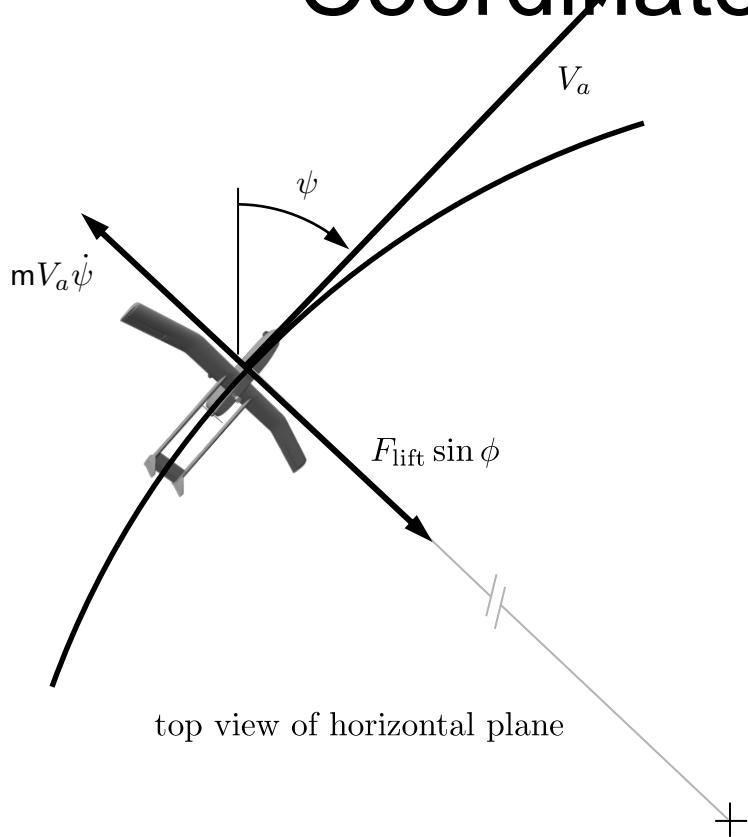
$$C_D(\alpha) = C_{D_p} + \frac{(C_{L_0} + C_{L_\alpha} \alpha)^2}{\pi e A R}$$

Linear model

$$C_L(\alpha) = C_{L_0} + C_{L_\alpha} \alpha$$

$$C_D(\alpha) = C_{D_0} + C_{D_\alpha} \alpha$$

Coordinated Turn – No Wind



view in direction of $-i^b$ axis
forces shown in j^b - k^b plane

$$F_{\text{lift}} \cos \phi = mg$$

$$\begin{aligned} F_{\text{lift}} \sin \phi &= m \frac{v^2}{R} \\ &= mv\omega \\ &= mV_a \dot{\psi} \end{aligned}$$

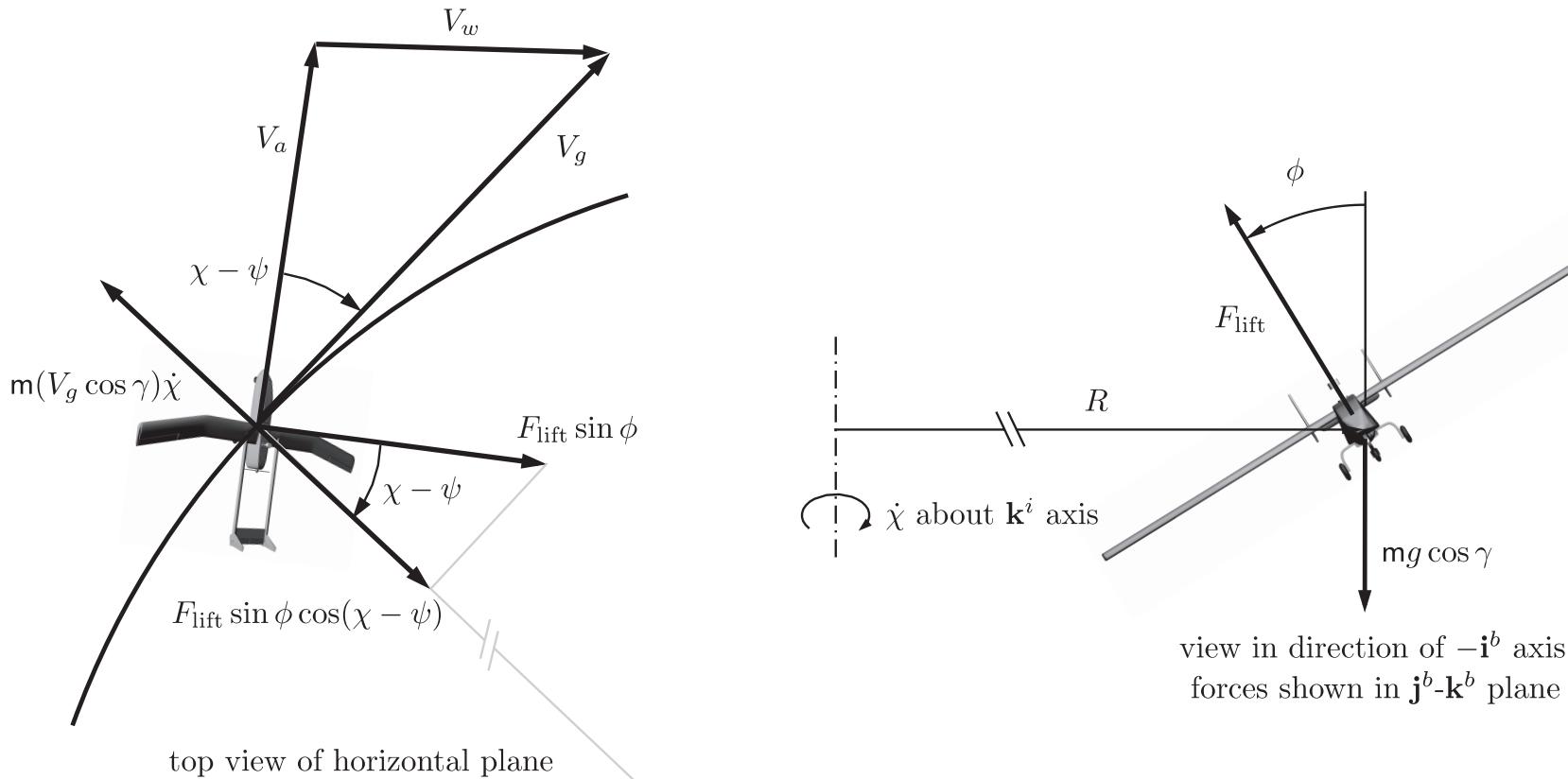
Dividing gives

$$\tan \phi = \frac{V_a \dot{\psi}}{g}$$

and

$$\dot{\psi} = \frac{g}{V_a} \tan \phi$$

Coordinated Turn



$$\begin{aligned}
 F_{lift} \sin \phi \cos(\chi - \psi) &= m \frac{v^2}{R} \\
 &= mv\omega \\
 &= m(V_g \cos \gamma) \dot{\chi}
 \end{aligned}$$

$$F_{lift} \cos \phi = mg \cos \gamma$$

Coordinated Turn

Dividing the two expressions gives

$$\dot{\chi} = \frac{g}{V_g} \tan \phi \cos(\chi - \psi)$$

which is the coordinated turn condition in wind

In the absence of wind, we have $V_a = V_g$ and $\psi = \chi$ which gives

$$\dot{\psi} = \frac{g}{V_a} \tan \phi$$

which is the expression commonly seen in the literature

The turning radius is given by

$$R = \frac{V_g \cos \gamma}{\dot{\chi}} = \frac{V_g^2 \cos \gamma}{g \tan \phi \cos(\chi - \psi)}$$

For level flight in the absence of wind we have the standard formula

$$R = \frac{V_a^2}{g \tan \phi}$$

Trim Conditions

Given the nonlinear system

$$\dot{x} = f(x, u)$$

The equilibria (x^*, u^*) are defined by

$$f(x^*, u^*) = 0$$

An aircraft in equilibrium is in a *trim condition*. In general, trim conditions may include states that are not constant. Therefore, trim conditions are given by

$$\dot{x}^* = f(x^*, u^*)$$

Calculating Trim

Objective is to compute trim states and inputs when aircraft simultaneously satisfies three conditions:

- Traveling at constant speed V_a^*
- Climbing at constant flight path angle γ^*
- In constant orbit of radius R^*

V_a^* , γ^* , and R^* , are inputs to the trim calculations

States: $x \triangleq (p_n, p_e, p_d, u, v, w, \phi, \theta, \psi, p, q, r)^\top$

Inputs: $u \triangleq (\delta_e, \delta_a, \delta_r, \delta_t)^\top$

Calculating Trim

For constant-climb orbit:

- speed of aircraft not changing → $\dot{u}^* = \dot{v}^* = \dot{w}^* = 0$
- roll and pitch angles constant → $\dot{\phi}^* = \dot{\theta}^* = \dot{p}^* = \dot{q}^* = 0$

Turn rate constant and given by

$$\dot{\psi}^* = \frac{V_a^*}{R^*} \cos \gamma^* \quad \rightarrow \quad \dot{r}^* = 0$$

Climb rate constant, and given by

$$\dot{h}^* = V_a^* \sin \gamma^*$$

Given parameters V_a^* , γ^* , and R^* , can specify \dot{x}^* as

$$\dot{x}^* = (p_n^* \ p_e^* \ \dot{h}^* \ \dot{u}^* \ \dot{v}^* \ \dot{w}^* \ \dot{\phi}^* \ \dot{\theta}^* \ \dot{\psi}^* \ \dot{p}^* \ \dot{q}^* \ \dot{r}^*)^\top$$

(continued...)

Calculating Trim

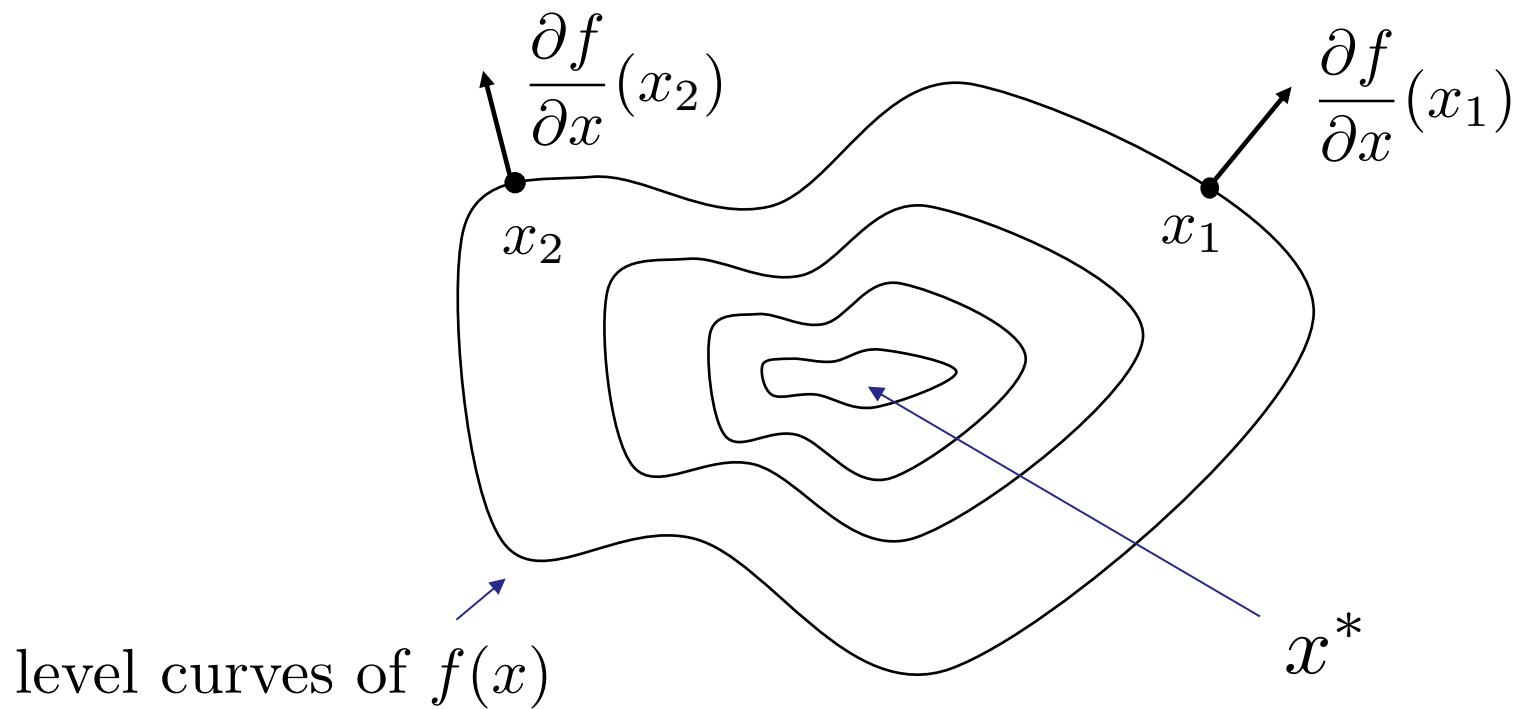
Given parameters V_a^* , γ^* , and R^* , can specify \dot{x}^* as

$$\dot{x}^* = \begin{pmatrix} \dot{p}_n^* \\ \dot{p}_e^* \\ \dot{h}^* \\ \dot{u}^* \\ \dot{v}^* \\ \dot{w}^* \\ \dot{\phi}^* \\ \dot{\theta}^* \\ \dot{\psi}^* \\ \dot{p}^* \\ \dot{q}^* \\ \dot{r}^* \end{pmatrix} = \begin{pmatrix} [\text{don't care}] \\ [\text{don't care}] \\ V_a^* \sin \gamma^* \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \frac{V_a^*}{R^*} \cos \gamma^* \\ 0 \\ 0 \\ 0 \end{pmatrix} = f(x^*, u^*)$$

Problem of finding x^* and u^* such that $\dot{x}^* = f(x^*, u^*)$, reduces to solving nonlinear algebraic systems of equations

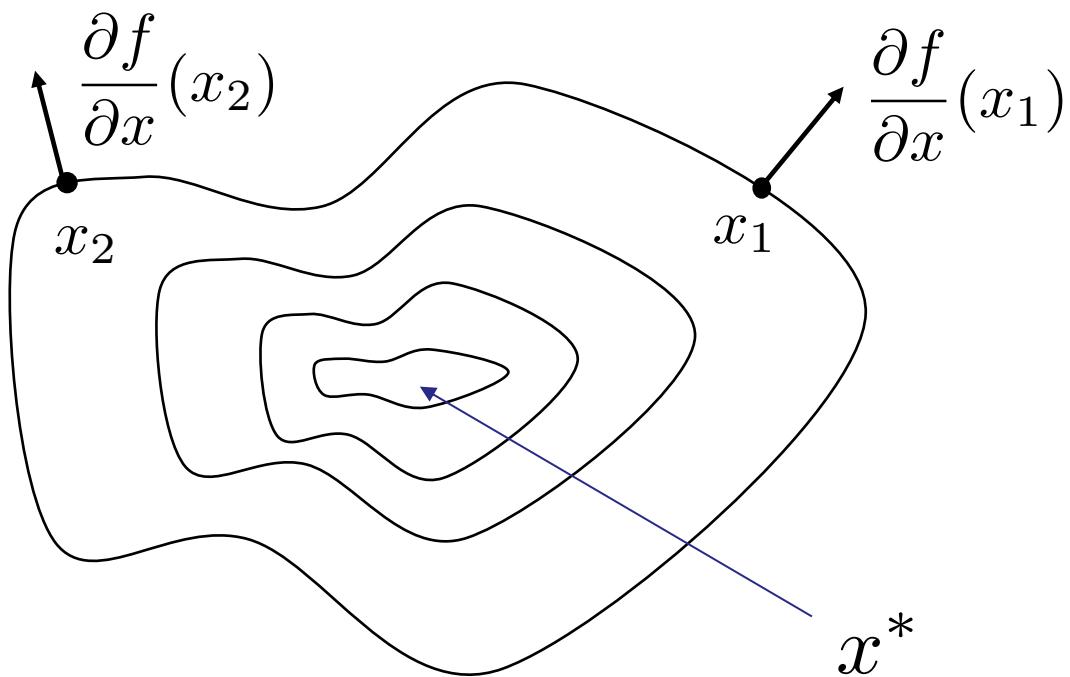
Can use Matlab trim command – see Appendix F

Unconstrained Optimization



$$x^* = \arg \min_{x \in \mathbb{R}^n} f(x)$$

Unconstrained Optimization



Gradient descent algorithm:

$$x^{[k+1]} = x^{[k]} - \alpha_k \frac{\partial f}{\partial x}(x^{[k]})$$

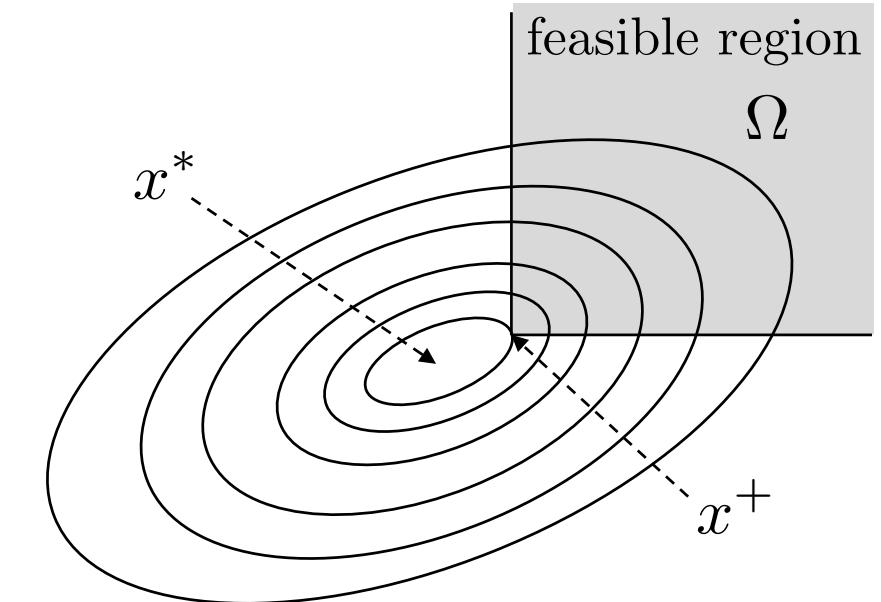
Constrained Optimization

Can generally pose as:

$$\min_{x \in \Omega} f(x)$$

Usually more manageable to pose as:

$$\begin{aligned} & \min_{x \in \mathbb{R}^n} f(x) \\ \text{s.t. } & h_1(x) = 0, \\ & \vdots, \\ & h_m(x) = 0, \\ & g_1(x) \leq 0, \\ & \vdots, \\ & g_p(x) \leq 0 \end{aligned}$$

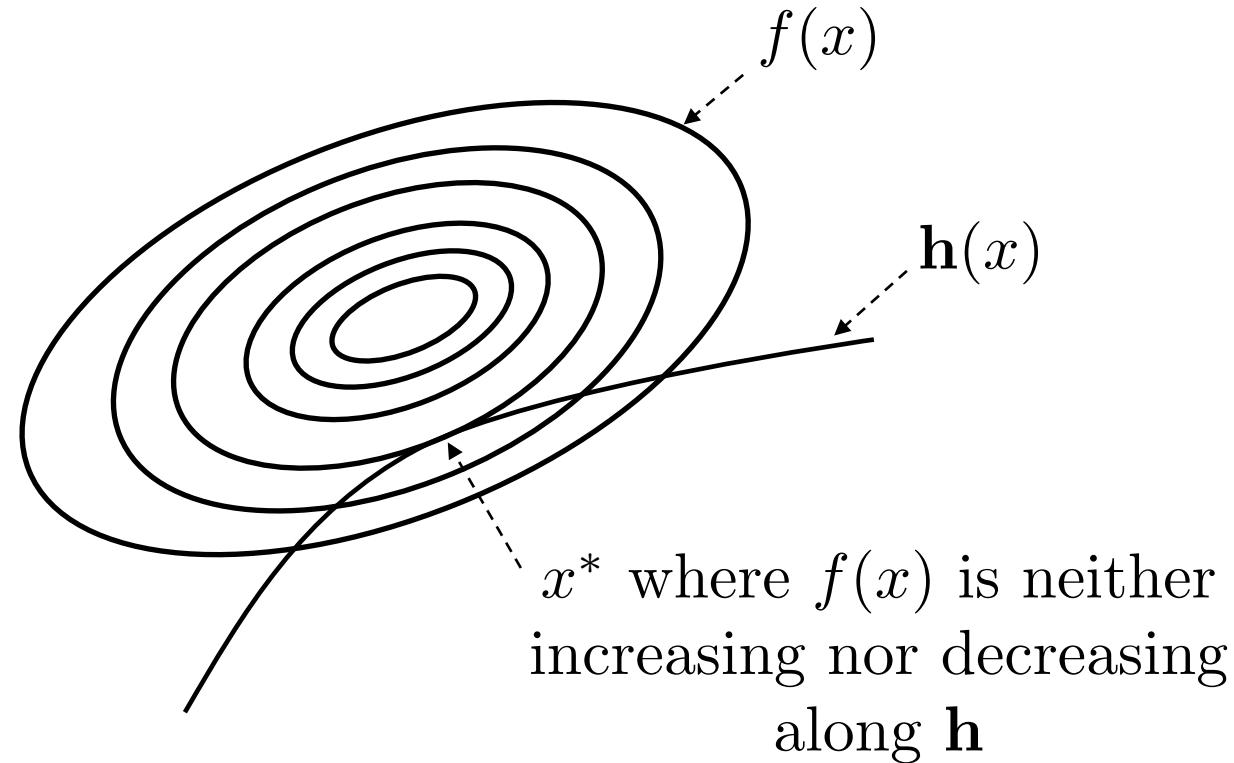


Note that the constrained optimum x^+ does not equal the unconstrained optimum x^* .

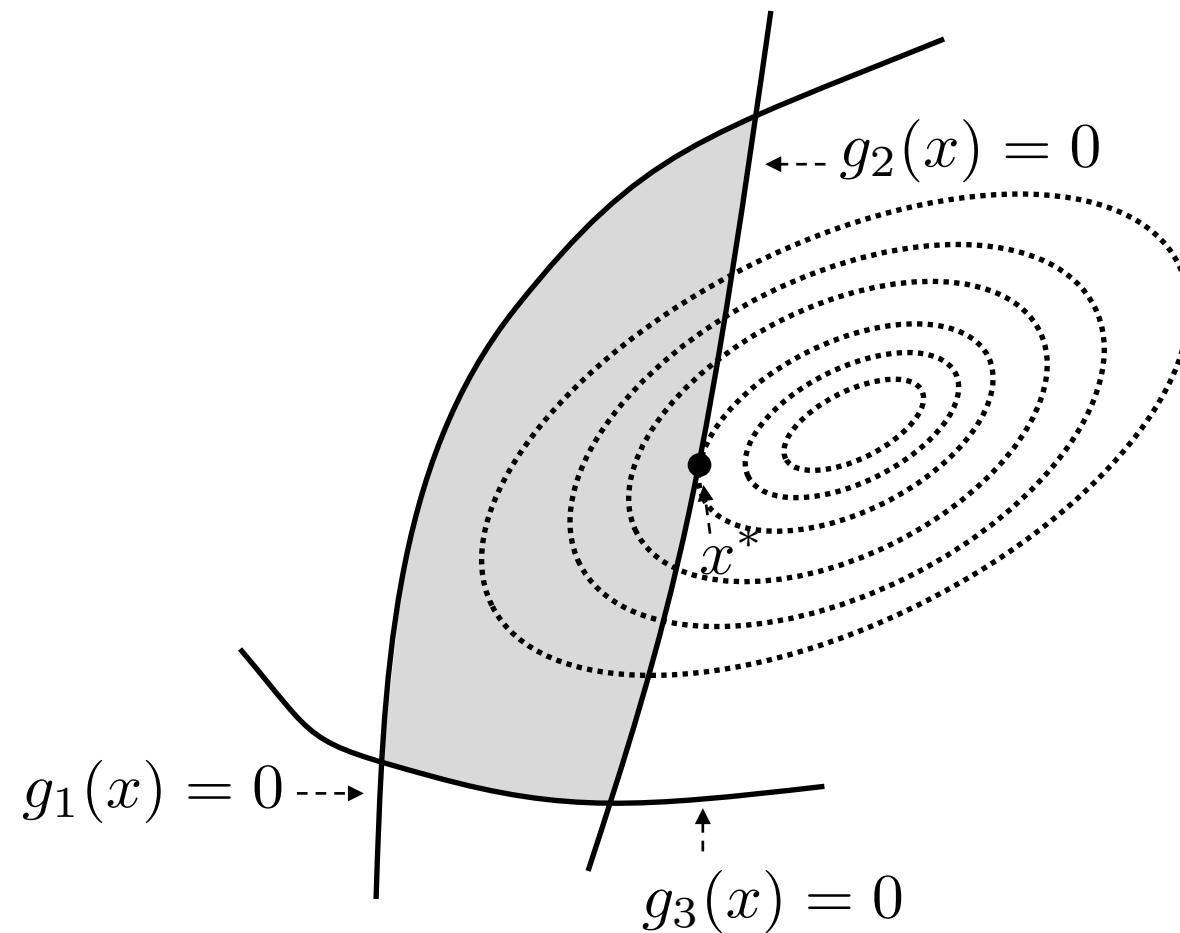
$h_i(x) = 0$ are called “equality constraints”

$g_i(x) \leq 0$ are called “inequality constraints”

Equality Constraints



Inequality Constraints



Optimization in Python

```
from scipy.optimize import minimize

minimize(fun, # function to be minimized
           x0, # initial guess for gradient descent
           args=(), # additional arguments for fun
           method=None, # optimization method
           jac=None, # function that defines Jacobian of f
           hess=None, # function that defines Hessian of f
           hessp=None, # Hessian in a specific direction
           bounds=None, # inequality bounds on x
           constraints=(), # definition of constraints
           tol=None, # tolerance for termination
           callback=None, # function called after each iteration
           options=None) # other algorithm options
```

Example: Optimization in Python

```
from scipy.optimize import minimize
import numpy as np

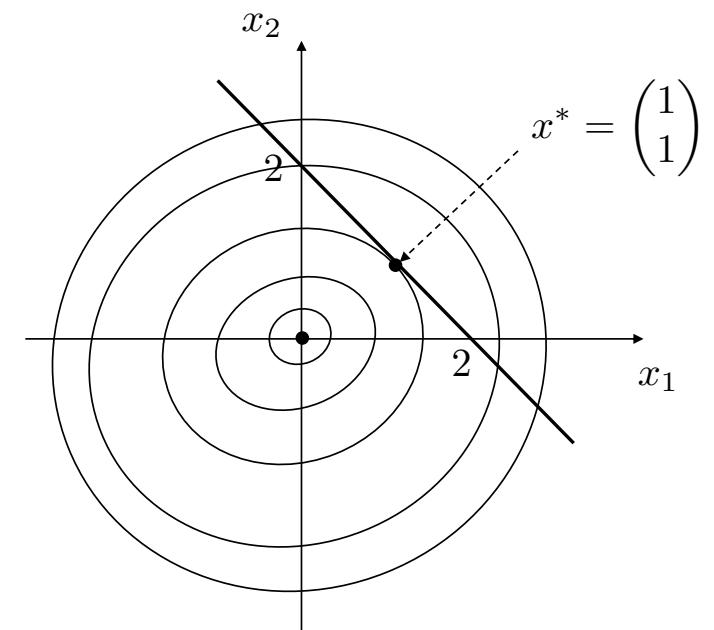
def objective(x):
    return(x[0]**2 + x[1]**2)

x0 = np.array([[5], [2]]) # initial condition
cons = ({'type': 'eq',
          'fun': lambda x: np.array([
              x[0]+x[1]-2, # x1+x2=2
          ]),
      })

res = minimize(objective, x0, method='SLSQP', constraints=cons)
print("xstar =", res.x)
```

$$\min x_1^2 + x_2^2$$

$$\text{s.t. } x_1 + x_2 = 2$$



Objective Function for Trim

```
def trim_objective_fun(x, mav, Va, gamma):
    state = x[0:13]
    delta = MsgDelta(elevator=x.item(13),
                     aileron=x.item(14),
                     rudder=x.item(15),
                     throttle=x.item(16))
    desired_trim_state_dot
        = np.array([[0., 0., -Va*np.sin(gamma), 0.,
                    0., 0., 0., 0., 0., 0., 0., 0.]]) .T
    mav._state = state
    mav._update_velocity_data()
    forces_moments = mav._forces_moments(delta)
    f = mav._derivatives(state, forces_moments)
    tmp = desired_trim_state_dot - f
    J = np.linalg.norm(tmp[2:13])**2.0
    return J
```

Constraints for Trim

```
cons = ({'type': 'eq',
      'fun': lambda x: np.array([
          x[3]**2 + x[4]**2 + x[5]**2 - Va**2, # magnitude of velocity
          # vector is Va
          x[4], # v=0, force side velocity to be zero
          x[6]**2 + x[7]**2 + x[8]**2 + x[9]**2 - 1., # quaternion is unit length
          x[7], # e1=0 - forcing e1=e3=0 ensures zero roll and zero yaw in trim
          x[9], # e3=0
          x[10], # p=0 - angular rates should all be zero
          x[11], # q=0
          x[12], # r=0
      ]),
      'jac': lambda x: np.array([
          [0., 0., 0., 2*x[3], 2*x[4], 2*x[5], 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
          [0., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
          [0., 0., 0., 0., 0., 0., 2*x[6], 2*x[7], 2*x[8], 2*x[9], 0., 0., 0., 0., 0., 0., 0.],
          [0., 0., 0., 0., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
          [0., 0., 0., 0., 0., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
          [0., 0., 0., 0., 0., 0., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 0., 0.],
          [0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 0.],
          [0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0.],
          [0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 1., 0., 0., 0., 0., 0.],
          [0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 1., 0., 0., 0., 0.]
      ])
  })
```

Lateral Transfer Functions - Roll

Objective: Find simple transfer function relationship between ϕ and δ_a

Start with: $\dot{\phi} = p + q \sin \phi \tan \theta + r \cos \phi \tan \theta$

Assuming that θ is small and defining the linearization error (can be considered as a control disturbance) to be $d_{\phi_1} \triangleq q \sin \phi \tan \theta + r \cos \phi \tan \theta$ gives

$$\dot{\phi} = p + d_{\phi_1}$$

Differentiating and substituting for $\dot{\phi}$ gives

$$\begin{aligned}\ddot{\phi} &= \dot{p} + \dot{d}_{\phi_1} \\ &= \Gamma_1 pq - \Gamma_2 qr + \frac{1}{2} \rho V_a^2 Sb \left[C_{p_0} + C_{p_\beta} \beta + C_{p_p} \frac{bp}{2V_a} + C_{p_r} \frac{br}{2V_a} + C_{p_{\delta_a}} \delta_a + C_{p_{\delta_r}} \delta_r \right] + \dot{d}_{\phi_1} \\ &= \Gamma_1 pq - \Gamma_2 qr + \frac{1}{2} \rho V_a^2 Sb \left[C_{p_0} + C_{p_\beta} \beta + C_{p_p} \frac{b}{2V_a} (\dot{\phi} - d_{\phi_1}) + C_{p_r} \frac{br}{2V_a} + C_{p_{\delta_a}} \delta_a + C_{p_{\delta_r}} \delta_r \right] + \dot{d}_{\phi_1} \\ &= \left(\frac{1}{2} \rho V_a^2 Sb C_{p_p} \frac{b}{2V_a} \right) \dot{\phi} + \left(\frac{1}{2} \rho V_a^2 Sb C_{p_{\delta_a}} \right) \delta_a + \left\{ \Gamma_1 pq - \Gamma_2 qr + \frac{1}{2} \rho V_a^2 Sb \left[C_{p_0} + C_{p_\beta} \beta - C_{p_p} \frac{b}{2V_a} (d_{\phi_1}) + C_{p_r} \right. \right. \\ &\quad \left. \left. \left. \frac{br}{2V_a} + C_{p_{\delta_a}} \delta_a + C_{p_{\delta_r}} \delta_r \right] \right\} + \dot{d}_{\phi_1} \\ &= -a_{\phi_1} \dot{\phi} + a_{\phi_2} \delta_a + d_{\phi_2}\end{aligned}$$

Equations of Motion

$$\dot{p}_n = (\cos \theta \cos \psi)u + (\sin \phi \sin \theta \cos \psi - \cos \phi \sin \psi)v + (\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi)w$$

$$\dot{p}_e = (\cos \theta \sin \psi)u + (\sin \phi \sin \theta \sin \psi + \cos \phi \cos \psi)v + (\cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi)w$$

$$\dot{h} = u \sin \theta - v \sin \phi \cos \theta - w \cos \phi \cos \theta$$

$$\dot{u} = rv - qw - g \sin \theta + \frac{\rho V_a^2 S}{2m} \left[C_X(\alpha) + C_{X_q}(\alpha) \frac{cq}{2V_a} + C_{X_{\delta_e}}(\alpha) \delta_e \right] + \frac{\rho S_{\text{prop}} C_{\text{prop}}}{2m} \left[(k_{\text{motor}} \delta_t)^2 - V_a^2 \right]$$

$$\dot{v} = pw - ru + g \cos \theta \sin \phi + \frac{\rho V_a^2 S}{2m} \left[C_{Y_0} + C_{Y_\beta} \beta + C_{Y_p} \frac{bp}{2V_a} + C_{Y_r} \frac{br}{2V_a} + C_{Y_{\delta_a}} \delta_a + C_{Y_{\delta_r}} \delta_r \right]$$

$$\dot{w} = qu - pv + g \cos \theta \cos \phi + \frac{\rho V_a^2 S}{2m} \left[C_Z(\alpha) + C_{Z_q}(\alpha) \frac{cq}{2V_a} + C_{Z_{\delta_e}}(\alpha) \delta_e \right]$$

$$\dot{\phi} = p + q \sin \phi \tan \theta + r \cos \phi \tan \theta$$

$$\dot{\theta} = q \cos \phi - r \sin \phi$$

$$\dot{\psi} = q \sin \phi \sec \theta + r \cos \phi \sec \theta$$

$$\dot{p} = \Gamma_1 pq - \Gamma_2 qr + \frac{1}{2} \rho V_a^2 S b \left[C_{p_0} + C_{p_\beta} \beta + C_{p_p} \frac{bp}{2V_a} + C_{p_r} \frac{br}{2V_a} + C_{p_{\delta_a}} \delta_a + C_{p_{\delta_r}} \delta_r \right]$$

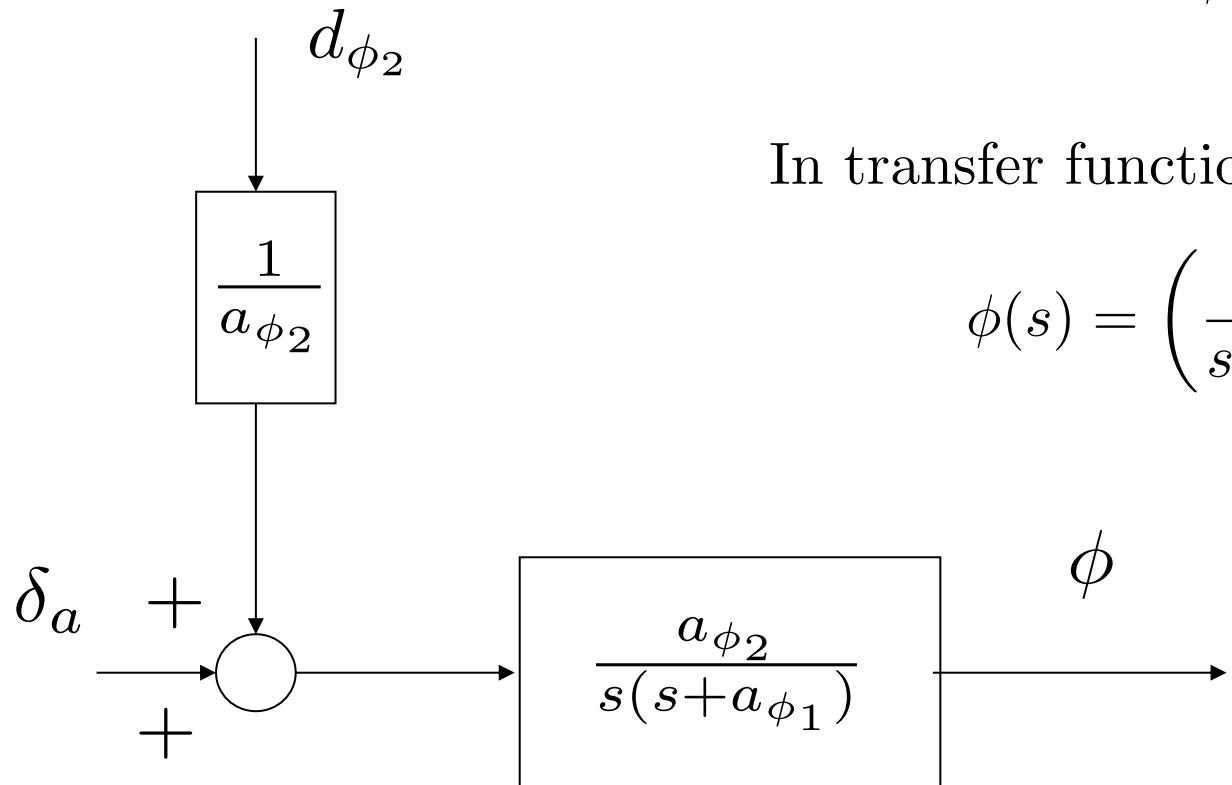
$$\dot{q} = \Gamma_5 pr - \Gamma_6 (p^2 - r^2) + \frac{\rho V_a^2 S c}{2J_y} \left[C_{m_0} + C_{m_\alpha} \alpha + C_{m_q} \frac{cq}{2V_a} + C_{m_{\delta_e}} \delta_e \right]$$

$$\dot{r} = \Gamma_7 pq - \Gamma_1 qr + \frac{1}{2} \rho V_a^2 S b \left[C_{r_0} + C_{r_\beta} \beta + C_{r_p} \frac{bp}{2V_a} + C_{r_r} \frac{br}{2V_a} + C_{r_{\delta_a}} \delta_a + C_{r_{\delta_r}} \delta_r \right]$$

Roll Dynamics

Linearization leads to second-order ordinary differential equation

$$\ddot{\phi} = -a_{\phi_1} \dot{\phi} + a_{\phi_2} \delta_a + d_{\phi_2}$$



In transfer function form we have

$$\phi(s) = \left(\frac{a_{\phi_2}}{s(s+a_{\phi_1})} \right) \left(\delta_a(s) + \frac{1}{a_{\phi_2}} d_{\phi_2}(s) \right)$$

Lateral Transfer Functions - Course

To derive transfer function from roll ϕ to course χ ,
start with no-wind coordinated turn condition

$$\dot{\chi} = \frac{g}{V_g} \tan \phi$$

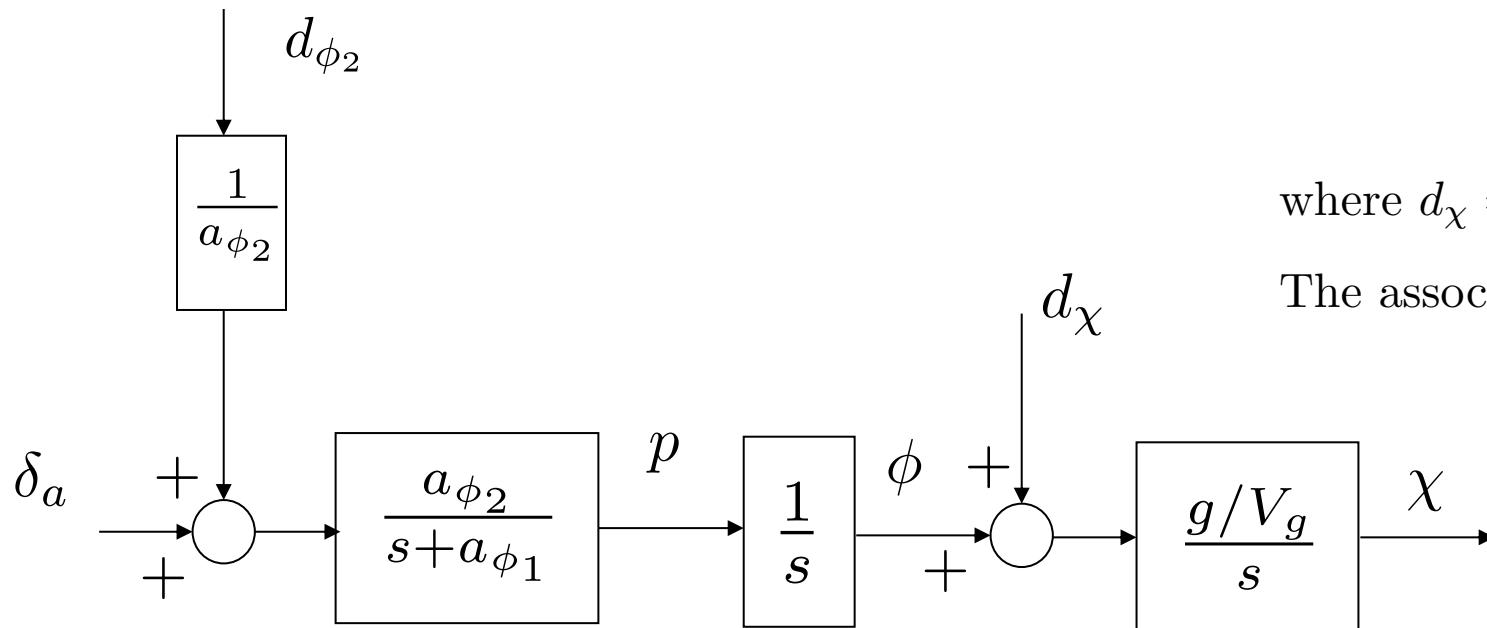
Add and subtract $\frac{g}{V_g}\phi$ to get

$$\begin{aligned}\dot{\chi} &= \frac{g}{V_g}\phi + \frac{g}{V_g}(\tan \phi - \phi) \\ &= \frac{g}{V_g}\phi + \frac{g}{V_g}d_\chi\end{aligned}$$

where $d_\chi = \tan \phi - \phi$ is an input disturbance

The associated transfer function is

$$\chi(s) = \frac{g/V_g}{s} (\phi(s) + d_\chi(s))$$



Lateral Transfer Functions - Sideslip

No-wind conditions: $v = V_a \sin \beta$

Constant airspeed: $\dot{v} = (V_a \cos \beta) \dot{\beta}$

Substituting for \dot{v} from the dynamics gives

$$(V_a \cos \beta) \dot{\beta} = pw - ru + g \cos \theta \sin \phi + \frac{\rho V_a^2 S}{2m} \left[C_{Y_0} + C_{Y_\beta} \beta + C_{Y_p} \frac{bp}{2V_a} + C_{Y_r} \frac{br}{2V_a} + C_{Y_{\delta_a}} \delta_a + C_{Y_{\delta_r}} \delta_r \right]$$

Solving for $\dot{\beta}$ and grouping terms gives

$$\dot{\beta} = -a_{\beta_1} \beta + a_{\beta_2} \delta_r + d_\beta$$

where

$$a_{\beta_1} = -\frac{\rho V_a S}{2m \cos \beta} C_{Y_\beta} \quad a_{\beta_2} = \frac{\rho V_a S}{2m \cos \beta} C_{Y_{\delta_r}}$$

$$d_\beta = \frac{1}{V_a \cos \beta} (pw - ru + g \cos \theta \sin \phi) + \frac{\rho V_a S}{2m \cos \beta} \left[C_{Y_0} + C_{Y_p} \frac{bp}{2V_a} + C_{Y_r} \frac{br}{2V_a} + C_{Y_{\delta_a}} \delta_a \right]$$

Equations of Motion

$$\dot{p}_n = (\cos \theta \cos \psi)u + (\sin \phi \sin \theta \cos \psi - \cos \phi \sin \psi)v + (\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi)w$$

$$\dot{p}_e = (\cos \theta \sin \psi)u + (\sin \phi \sin \theta \sin \psi + \cos \phi \cos \psi)v + (\cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi)w$$

$$\dot{h} = u \sin \theta - v \sin \phi \cos \theta - w \cos \phi \cos \theta$$

$$\dot{u} = rv - qw - g \sin \theta + \frac{\rho V_a^2 S}{2m} \left[C_X(\alpha) + C_{X_q}(\alpha) \frac{cq}{2V_a} + C_{X_{\delta_e}}(\alpha) \delta_e \right] + \frac{\rho S_{\text{prop}} C_{\text{prop}}}{2m} \left[(k_{\text{motor}} \delta_t)^2 - V_a^2 \right]$$

$$\dot{v} = pw - ru + g \cos \theta \sin \phi + \frac{\rho V_a^2 S}{2m} \left[C_{Y_0} + C_{Y_\beta} \beta + C_{Y_p} \frac{bp}{2V_a} + C_{Y_r} \frac{br}{2V_a} + C_{Y_{\delta_a}} \delta_a + C_{Y_{\delta_r}} \delta_r \right]$$

$$\dot{w} = qu - pv + g \cos \theta \cos \phi + \frac{\rho V_a^2 S}{2m} \left[C_Z(\alpha) + C_{Z_q}(\alpha) \frac{cq}{2V_a} + C_{Z_{\delta_e}}(\alpha) \delta_e \right]$$

$$\dot{\phi} = p + q \sin \phi \tan \theta + r \cos \phi \tan \theta$$

$$\dot{\theta} = q \cos \phi - r \sin \phi$$

$$\dot{\psi} = q \sin \phi \sec \theta + r \cos \phi \sec \theta$$

$$\dot{p} = \Gamma_1 pq - \Gamma_2 qr + \frac{1}{2} \rho V_a^2 S b \left[C_{p_0} + C_{p_\beta} \beta + C_{p_p} \frac{bp}{2V_a} + C_{p_r} \frac{br}{2V_a} + C_{p_{\delta_a}} \delta_a + C_{p_{\delta_r}} \delta_r \right]$$

$$\dot{q} = \Gamma_5 pr - \Gamma_6 (p^2 - r^2) + \frac{\rho V_a^2 S c}{2J_y} \left[C_{m_0} + C_{m_\alpha} \alpha + C_{m_q} \frac{cq}{2V_a} + C_{m_{\delta_e}} \delta_e \right]$$

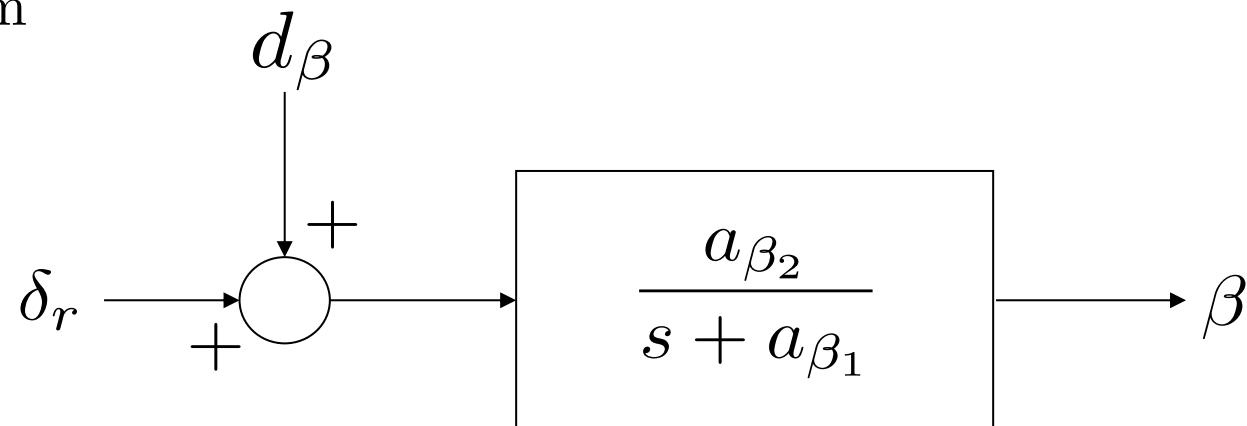
$$\dot{r} = \Gamma_7 pq - \Gamma_1 qr + \frac{1}{2} \rho V_a^2 S b \left[C_{r_0} + C_{r_\beta} \beta + C_{r_p} \frac{bp}{2V_a} + C_{r_r} \frac{br}{2V_a} + C_{r_{\delta_a}} \delta_a + C_{r_{\delta_r}} \delta_r \right]$$

Sideslip Dynamics

The differential equation

$$\dot{\beta} = -a_{\beta_1}\beta + a_{\beta_2}\delta_r + d_\beta$$

leads to the block diagram



$$a_{\beta_1} = -\frac{\rho V_a S}{2m} C_{Y_\beta}$$

$$a_{\beta_2} = \frac{\rho V_a S}{2m} C_{Y_{\delta_r}}$$

$$d_\beta = \frac{1}{V_a} (pw - ru + g \cos \theta \sin \phi) + \frac{\rho V_a S}{2m} \left[C_{Y_0} + C_{Y_p} \frac{bp}{2V_a} + C_{Y_r} \frac{br}{2V_a} + C_{Y_{\delta_a}} \delta_a \right]$$

Longitudinal Transfer Functions - Pitch

To derive the transfer function from elevator to pitch angle

$$\begin{aligned}\dot{\theta} &= q \cos \phi - r \sin \phi \\ &= q + q(\cos \phi - 1) - r \sin \phi \\ &\triangleq q + d_{\theta_1}\end{aligned}$$

Differentiating and substituting \dot{q} from the dynamics gives

$$\begin{aligned}\ddot{\theta} &= \Gamma_6(r^2 - p^2) + \Gamma_5 pr + \frac{\rho V_a^2 c S}{2J_y} \left[C_{m_0} + C_{m_\alpha} \alpha + C_{m_q} \frac{cq}{2V_a} + C_{m_{\delta_e}} \delta_e \right] + \dot{d}_{\theta_1} \\ &= \Gamma_6(r^2 - p^2) + \Gamma_5 pr + \frac{\rho V_a^2 c S}{2J_y} \left[C_{m_0} + C_{m_\alpha} (\theta - \gamma) + C_{m_q} \frac{c}{2V_a} (\dot{\theta} - d_{\theta_1}) + C_{m_{\delta_e}} \delta_e \right] + \dot{d}_{\theta_1} \\ &= \left(\frac{\rho V_a^2 c S}{2J_y} C_{m_q} \frac{c}{2V_a} \right) \dot{\theta} + \left(\frac{\rho V_a^2 c S}{2J_y} C_{m_\alpha} \right) \theta + \left(\frac{\rho V_a^2 c S}{2J_y} C_{m_{\delta_e}} \right) \delta_e + \left\{ \Gamma_6(r^2 - p^2) \right. \\ &\quad \left. + \Gamma_5 pr + \frac{\rho V_a^2 c S}{J_y} \left[C_{m_0} - C_{m_\alpha} \gamma - C_{m_q} \frac{c}{2V_a} d_{\theta_1} \right] + \dot{d}_{\theta_1} \right\} \\ &= -a_{\theta_1} \dot{\theta} - a_{\theta_2} \theta + a_{\theta_3} \delta_e + d_{\theta_2}\end{aligned}$$

Equations of Motion

$$\dot{p}_n = (\cos \theta \cos \psi)u + (\sin \phi \sin \theta \cos \psi - \cos \phi \sin \psi)v + (\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi)w$$

$$\dot{p}_e = (\cos \theta \sin \psi)u + (\sin \phi \sin \theta \sin \psi + \cos \phi \cos \psi)v + (\cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi)w$$

$$\dot{h} = u \sin \theta - v \sin \phi \cos \theta - w \cos \phi \cos \theta$$

$$\dot{u} = rv - qw - g \sin \theta + \frac{\rho V_a^2 S}{2m} \left[C_X(\alpha) + C_{X_q}(\alpha) \frac{cq}{2V_a} + C_{X_{\delta_e}}(\alpha) \delta_e \right] + \frac{\rho S_{\text{prop}} C_{\text{prop}}}{2m} \left[(k_{\text{motor}} \delta_t)^2 - V_a^2 \right]$$

$$\dot{v} = pw - ru + g \cos \theta \sin \phi + \frac{\rho V_a^2 S}{2m} \left[C_{Y_0} + C_{Y_\beta} \beta + C_{Y_p} \frac{bp}{2V_a} + C_{Y_r} \frac{br}{2V_a} + C_{Y_{\delta_a}} \delta_a + C_{Y_{\delta_r}} \delta_r \right]$$

$$\dot{w} = qu - pv + g \cos \theta \cos \phi + \frac{\rho V_a^2 S}{2m} \left[C_Z(\alpha) + C_{Z_q}(\alpha) \frac{cq}{2V_a} + C_{Z_{\delta_e}}(\alpha) \delta_e \right]$$

$$\dot{\phi} = p + q \sin \phi \tan \theta + r \cos \phi \tan \theta$$

$$\dot{\theta} = q \cos \phi - r \sin \phi$$

$$\dot{\psi} = q \sin \phi \sec \theta + r \cos \phi \sec \theta$$

$$\dot{p} = \Gamma_1 pq - \Gamma_2 qr + \frac{1}{2} \rho V_a^2 S b \left[C_{p_0} + C_{p_\beta} \beta + C_{p_p} \frac{bp}{2V_a} + C_{p_r} \frac{br}{2V_a} + C_{p_{\delta_a}} \delta_a + C_{p_{\delta_r}} \delta_r \right]$$

$$\dot{q} = \Gamma_5 pr - \Gamma_6 (p^2 - r^2) + \frac{\rho V_a^2 S c}{2J_y} \left[C_{m_0} + C_{m_\alpha} \alpha + C_{m_q} \frac{cq}{2V_a} + C_{m_{\delta_e}} \delta_e \right]$$

$$\dot{r} = \Gamma_7 pq - \Gamma_1 qr + \frac{1}{2} \rho V_a^2 S b \left[C_{r_0} + C_{r_\beta} \beta + C_{r_p} \frac{bp}{2V_a} + C_{r_r} \frac{br}{2V_a} + C_{r_{\delta_a}} \delta_a + C_{r_{\delta_r}} \delta_r \right]$$

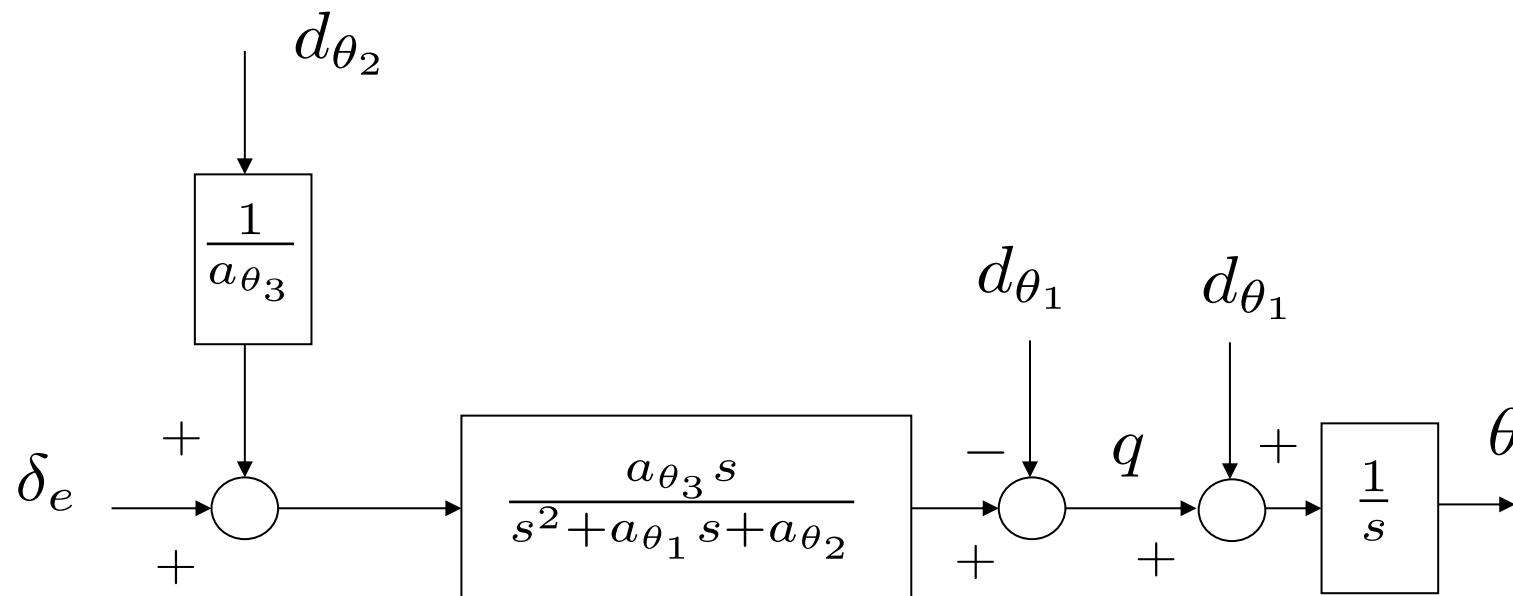
Longitudinal Transfer Functions - Pitch

The differential equation model is

$$\ddot{\theta} = -a_{\theta_1}\dot{\theta} - a_{\theta_2}\theta + a_{\theta_3}\delta_e + d_{\theta_2}$$

with associated transfer function

$$\theta(s) = \left(\frac{a_{\theta_3}}{s^2 + a_{\theta_1}s + a_{\theta_2}} \right) \left(\delta_e(s) + \frac{1}{a_{\theta_3}}d_{\theta_2}(s) \right)$$



Longitudinal TF - Altitude from Pitch

To derive the transfer function from pitch to altitude
(assuming constant airspeed)

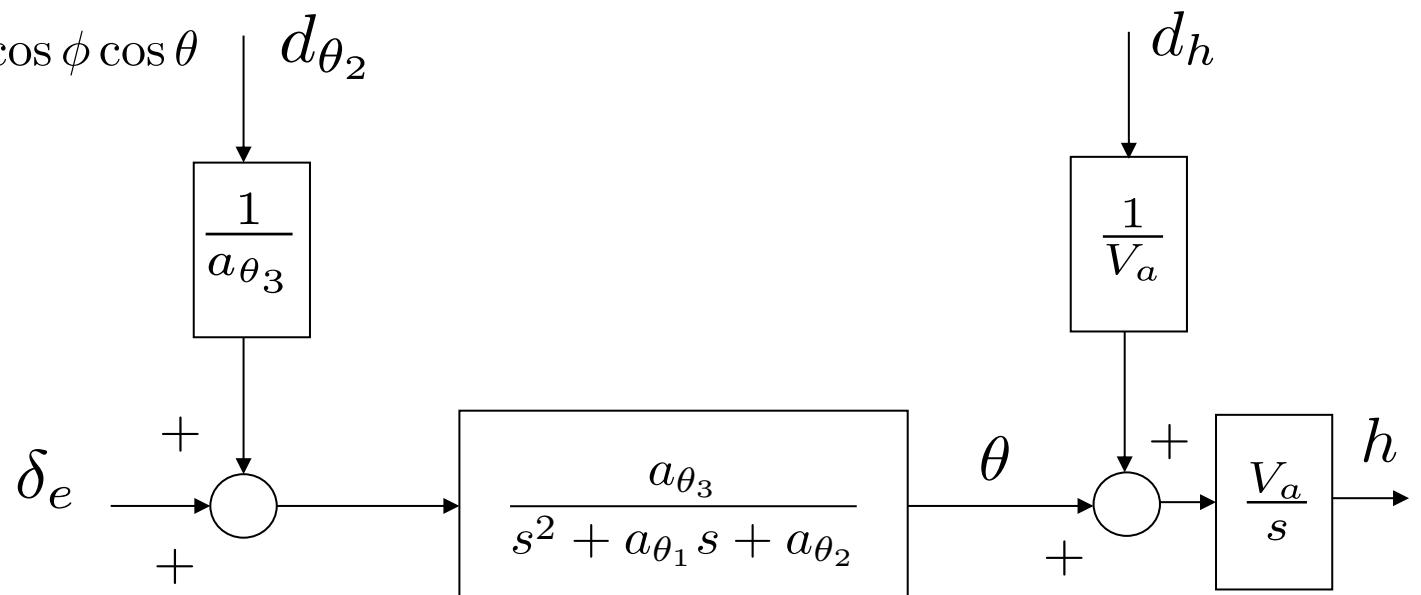
$$\begin{aligned}\dot{h} &= u \sin \theta - v \sin \phi \cos \theta - w \cos \phi \cos \theta \\ &= V_a \theta + (u \sin \theta - V_a \theta) - v \sin \phi \cos \theta - w \cos \phi \cos \theta \\ &= V_a \theta + d_h\end{aligned}$$

where

$$d_h \triangleq (u \sin \theta - V_a \theta) - v \sin \phi \cos \theta - w \cos \phi \cos \theta$$

The associated transfer function is

$$h(s) = \frac{V_a}{s} \left(\theta + \frac{1}{V_a} d_h \right)$$



Equations of Motion

$$\dot{p}_n = (\cos \theta \cos \psi)u + (\sin \phi \sin \theta \cos \psi - \cos \phi \sin \psi)v + (\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi)w$$

$$\dot{p}_e = (\cos \theta \sin \psi)u + (\sin \phi \sin \theta \sin \psi + \cos \phi \cos \psi)v + (\cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi)w$$

$$\dot{h} = u \sin \theta - v \sin \phi \cos \theta - w \cos \phi \cos \theta$$

$$\dot{u} = rv - qw - g \sin \theta + \frac{\rho V_a^2 S}{2m} \left[C_X(\alpha) + C_{X_q}(\alpha) \frac{cq}{2V_a} + C_{X_{\delta_e}}(\alpha) \delta_e \right] + \frac{\rho S_{\text{prop}} C_{\text{prop}}}{2m} \left[(k_{\text{motor}} \delta_t)^2 - V_a^2 \right]$$

$$\dot{v} = pw - ru + g \cos \theta \sin \phi + \frac{\rho V_a^2 S}{2m} \left[C_{Y_0} + C_{Y_\beta} \beta + C_{Y_p} \frac{bp}{2V_a} + C_{Y_r} \frac{br}{2V_a} + C_{Y_{\delta_a}} \delta_a + C_{Y_{\delta_r}} \delta_r \right]$$

$$\dot{w} = qu - pv + g \cos \theta \cos \phi + \frac{\rho V_a^2 S}{2m} \left[C_Z(\alpha) + C_{Z_q}(\alpha) \frac{cq}{2V_a} + C_{Z_{\delta_e}}(\alpha) \delta_e \right]$$

$$\dot{\phi} = p + q \sin \phi \tan \theta + r \cos \phi \tan \theta$$

$$\dot{\theta} = q \cos \phi - r \sin \phi$$

$$\dot{\psi} = q \sin \phi \sec \theta + r \cos \phi \sec \theta$$

$$\dot{p} = \Gamma_1 pq - \Gamma_2 qr + \frac{1}{2} \rho V_a^2 S b \left[C_{p_0} + C_{p_\beta} \beta + C_{p_p} \frac{bp}{2V_a} + C_{p_r} \frac{br}{2V_a} + C_{p_{\delta_a}} \delta_a + C_{p_{\delta_r}} \delta_r \right]$$

$$\dot{q} = \Gamma_5 pr - \Gamma_6 (p^2 - r^2) + \frac{\rho V_a^2 S c}{2J_y} \left[C_{m_0} + C_{m_\alpha} \alpha + C_{m_q} \frac{cq}{2V_a} + C_{m_{\delta_e}} \delta_e \right]$$

$$\dot{r} = \Gamma_7 pq - \Gamma_1 qr + \frac{1}{2} \rho V_a^2 S b \left[C_{r_0} + C_{r_\beta} \beta + C_{r_p} \frac{bp}{2V_a} + C_{r_r} \frac{br}{2V_a} + C_{r_{\delta_a}} \delta_a + C_{r_{\delta_r}} \delta_r \right]$$

Longitudinal Transfer Functions - Airspeed

Both pitch angle and throttle strongly affect airspeed. We wish to derive the relationship.

If there is no wind, then

$$V_a = \sqrt{u^2 + v^2 + w^2}$$

Differentiating gives

$$\dot{V}_a = \dot{u} \cos \alpha \cos \beta + \dot{v} \sin \beta + \dot{w} \sin \alpha \cos \beta = \dot{u} \cos \alpha + \dot{w} \sin \alpha + d_{V_1}$$

where

$$d_{V_1} = -\dot{u}(1 - \cos \beta) \cos \alpha - \dot{w}(1 - \cos \beta) \sin \alpha + \dot{v} \sin \beta$$

Substituting from the dynamics gives

$$\begin{aligned}\dot{V}_a &= rV_a \cos \alpha \sin \beta - pV_a \sin \alpha \sin \beta - g \cos \alpha \sin \theta + g \sin \alpha \cos \theta \cos \phi \\ &\quad + \frac{\rho V_a^2 S}{2m} \left[-C_D(\alpha) - C_{D_\alpha} \alpha - C_{D_q} \frac{cq}{2V_a} - C_{D_{\delta_e}} \delta_e \right] + \frac{1}{m} T_p(\delta_t, V_a) + d_{V_1} \\ &= (rV_a \cos \alpha - pV_a \sin \alpha) \sin \beta - g \sin(\theta - \alpha) - g \sin \alpha \cos \theta (1 - \cos \phi) \\ &\quad + \frac{\rho V_a^2 S}{2m} \left[-C_{D_0} - C_{D_\alpha} \alpha - C_{D_q} \frac{cq}{2V_a} - C_{D_{\delta_e}} \delta_e \right] + \frac{1}{m} T_p(\delta_t, V_a) + d_{V_1} \\ &= -g \sin \gamma + \frac{\rho V_a^2 S}{2m} \left[-C_{D_0} - C_{D_\alpha} \alpha - C_{D_q} \frac{cq}{2V_a} - C_{D_{\delta_e}} \delta_e \right] + \frac{1}{m} T_p(\delta_t, V_a) + d_{V_2}\end{aligned}$$

Longitudinal Transfer Functions - Airspeed

$$\dot{V}_a = -g \sin \gamma + \frac{\rho V_a^2 S}{2m} \left[-C_{D_0} - C_{D_\alpha} \alpha - C_{D_q} \frac{cq}{2V_a} - C_{D_{\delta_e}} \delta_e \right] + \frac{1}{m} T_p(\delta_t, V_a) + d_{V_2}$$

Since this equation is nonlinear in V_a , we will linearize about trim: α^* , V_a^* , θ^* , δ_e^* , δ_t^* , and let

$$\bar{V}_a = V_a - V_a^*, \quad \bar{\theta} = \theta - \theta^*, \quad \bar{\delta}_e = \delta_e - \delta_e^*, \quad \bar{\delta}_t = \delta_t - \delta_t^*$$

to get

$$\begin{aligned} \dot{\bar{V}}_a &= -g \cos(\theta^* - \alpha^*) \bar{\theta} + \left\{ \frac{\rho V_a^* S}{m} \left[-C_{D_0} - C_{D_\alpha} \alpha^* - C_{D_{\delta_e}} \delta_e^* \right] + \frac{1}{m} \frac{\partial T_p}{\partial V_a}(\delta_t^*, V_a^*) \right\} \bar{V}_a + \frac{1}{m} \frac{\partial T_p}{\partial \delta_t}(\delta_t^*, V_a^*) \bar{\delta}_t + d_V \\ &= -a_{V_1} \bar{V}_a + a_{V_2} \bar{\delta}_t - a_{V_3} \bar{\theta} + d_V \end{aligned}$$

where

$$\begin{aligned} a_{V_1} &= \frac{\rho V_a^* S}{m} \left[C_{D_0} + C_{D_\alpha} \alpha^* + C_{D_{\delta_e}} \delta_e^* \right] - \frac{1}{m} \frac{\partial T_p}{\partial V_a}(\delta_t^*, V_a^*) \\ a_{V_2} &= \frac{1}{m} \frac{\partial T_p}{\partial \delta_t}(\delta_t^*, V_a^*) \\ a_{V_3} &= g \cos(\theta^* - \alpha^*) \end{aligned}$$

Longitudinal Transfer Functions - Airspeed

The associated transfer function is

$$\bar{V}_a(s) = \frac{1}{s + a_{V_1}} (a_{V_2} \bar{\delta}_t(s) - a_{V_3} \bar{\theta}(s) + d_V(s))$$

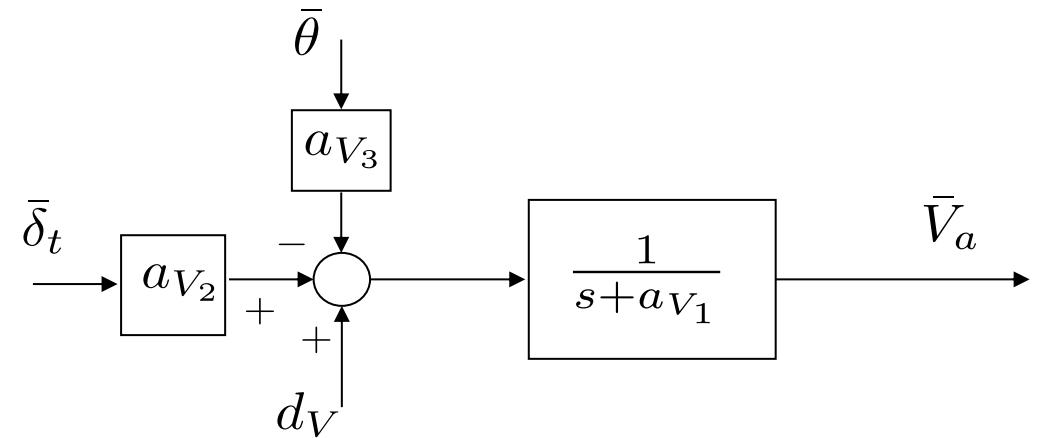
For throttle we have

$$\bar{V}_a(s) = \frac{a_{V_2}}{s + a_{V_1}} (\bar{\delta}_t(s) + d_V(s))$$

Key inputs are pitch angle
and throttle command

and for pitch angle we have

$$\bar{V}_a(s) = \frac{-a_{V_3}}{s + a_{V_1}} (\bar{\theta}(s) + d_V(s))$$



Linear State-space Models

Nonlinear state equations: $\dot{x} = f(x, u)$

Trim condition: $\dot{x}^* = f(x^*, u^*)$

Deviation from trim: $\bar{x} = x - x^*$

Rewriting the state equation in terms of deviation from trim gives

$$\begin{aligned}\dot{\bar{x}} &= \dot{x} - \dot{x}^* \\ &= f(x, u) - f(x^*, u^*) \\ &= f(x + x^* - x^*, u + u^* - u^*) - f(x^*, u^*) \\ &= f(x^* + \bar{x}, u^* + \bar{u}) - f(x^*, u^*)\end{aligned}$$

Using a Taylor series expansion around trim to linearize gives

$$\begin{aligned}\dot{\bar{x}} &= f(x^*, u^*) + \frac{\partial f(x^*, u^*)}{\partial x} \bar{x} + \frac{\partial f(x^*, u^*)}{\partial u} \bar{u} + H.O.T - f(x^*, u^*) \\ &\approx \frac{\partial f(x^*, u^*)}{\partial x} \bar{x} + \frac{\partial f(x^*, u^*)}{\partial u} \bar{u} \\ &\stackrel{\triangle}{=} A\bar{x} + B\bar{u}\end{aligned}$$

Lateral State-space Equations

The lateral states and inputs are defined as

$$\dot{x}_{\text{lat}} \triangleq (v, p, r, \phi, \psi)^\top \quad u_{\text{lat}} \triangleq (\delta_a, \delta_r)^\top$$

The nonlinear equations of motion are

$$\begin{aligned}\dot{v} &= pw - ru + g \cos \theta \sin \phi + \frac{\rho \sqrt{u^2 + v^2 + w^2} S}{2m} \frac{b}{2} [C_{Y_p} p + C_{Y_r} r] \\ &\quad + \frac{\rho(u^2 + v^2 + w^2)S}{2m} [C_{Y_0} + C_{Y_\beta} \tan^{-1} \left(\frac{v}{\sqrt{u^2 + w^2}} \right) + C_{Y_{\delta_a}} \delta_a + C_{Y_{\delta_r}} \delta_r] \\ \dot{p} &= \Gamma_1 pq - \Gamma_2 qr + \frac{\rho \sqrt{u^2 + v^2 + w^2} S}{2} \frac{b^2}{2} [C_{p_p} p + C_{p_r} r] + \Gamma_3 Q_p(\delta_t, V_a) \\ &\quad + \frac{1}{2} \rho(u^2 + v^2 + w^2) S b [C_{p_0} + C_{p_\beta} \tan^{-1} \left(\frac{v}{\sqrt{u^2 + w^2}} \right) + C_{p_{\delta_a}} \delta_a + C_{p_{\delta_r}} \delta_r] \\ \dot{r} &= \Gamma_7 pq - \Gamma_1 qr + \frac{\rho \sqrt{u^2 + v^2 + w^2} S}{2} \frac{b^2}{2} [C_{r_p} p + C_{r_r} r] + \Gamma_4 Q_p(\delta_t, V_a) \\ &\quad + \frac{1}{2} \rho(u^2 + v^2 + w^2) S b [C_{r_0} + C_{r_\beta} \tan^{-1} \left(\frac{v}{\sqrt{u^2 + w^2}} \right) + C_{r_{\delta_a}} \delta_a + C_{r_{\delta_r}} \delta_r] \\ \dot{\phi} &= p + q \sin \phi \tan \theta + r \cos \phi \tan \theta \\ \dot{\psi} &= q \sin \phi \sec \theta + r \cos \phi \sec \theta\end{aligned}$$

where we have used $\beta = \tan^{-1} \left(\frac{v}{\sqrt{u^2 + w^2}} \right)$ and $V_a = \sqrt{u^2 + v^2 + w^2}$

Jacobian Matrices

$$A_{\text{lat}} = \frac{\partial f_{\text{lat}}}{\partial x_{\text{lat}}} = \begin{pmatrix} \frac{\partial \dot{v}}{\partial v} & \frac{\partial \dot{v}}{\partial p} & \frac{\partial \dot{v}}{\partial r} & \frac{\partial \dot{v}}{\partial \phi} & \frac{\partial \dot{v}}{\partial \psi} \\ \frac{\partial \dot{p}}{\partial v} & \frac{\partial \dot{p}}{\partial p} & \frac{\partial \dot{p}}{\partial r} & \frac{\partial \dot{p}}{\partial \phi} & \frac{\partial \dot{p}}{\partial \psi} \\ \frac{\partial \dot{r}}{\partial v} & \frac{\partial \dot{r}}{\partial p} & \frac{\partial \dot{r}}{\partial r} & \frac{\partial \dot{r}}{\partial \phi} & \frac{\partial \dot{r}}{\partial \psi} \\ \frac{\partial \dot{\phi}}{\partial v} & \frac{\partial \dot{\phi}}{\partial p} & \frac{\partial \dot{\phi}}{\partial r} & \frac{\partial \dot{\phi}}{\partial \phi} & \frac{\partial \dot{\phi}}{\partial \psi} \\ \frac{\partial \dot{\psi}}{\partial v} & \frac{\partial \dot{\psi}}{\partial p} & \frac{\partial \dot{\psi}}{\partial r} & \frac{\partial \dot{\psi}}{\partial \phi} & \frac{\partial \dot{\psi}}{\partial \psi} \end{pmatrix}$$

$$B_{\text{lat}} = \frac{\partial f_{\text{lat}}}{\partial u_{\text{lat}}} = \begin{pmatrix} \frac{\partial \dot{v}}{\partial \delta_a} & \frac{\partial \dot{v}}{\partial \delta_r} \\ \frac{\partial \dot{p}}{\partial \delta_a} & \frac{\partial \dot{p}}{\partial \delta_r} \\ \frac{\partial \dot{r}}{\partial \delta_a} & \frac{\partial \dot{r}}{\partial \delta_r} \\ \frac{\partial \dot{\phi}}{\partial \delta_a} & \frac{\partial \dot{\phi}}{\partial \delta_r} \\ \frac{\partial \dot{\psi}}{\partial \delta_a} & \frac{\partial \dot{\psi}}{\partial \delta_r} \end{pmatrix}$$

Take partial derivatives and evaluate them at trim state and trim input

Lateral State-space Equations

$$\begin{pmatrix} \dot{\bar{v}} \\ \dot{\bar{p}} \\ \dot{\bar{r}} \\ \dot{\bar{\phi}} \\ \dot{\bar{\psi}} \end{pmatrix} = \begin{pmatrix} Y_v & Y_p & Y_r & A_{14} & 0 \\ L_v & L_p & L_r & 0 & 0 \\ N_v & N_p & N_r & 0 & 0 \\ 0 & 1 & A_{43} & A_{44} & 0 \\ 0 & 0 & A_{53} & A_{54} & 0 \end{pmatrix} \begin{pmatrix} \bar{v} \\ \bar{p} \\ \bar{r} \\ \bar{\phi} \\ \bar{\psi} \end{pmatrix} + \begin{pmatrix} Y_{\delta_a} & Y_{\delta_r} \\ L_{\delta_a} & L_{\delta_r} \\ N_{\delta_a} & N_{\delta_r} \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \bar{\delta}_a \\ \bar{\delta}_r \end{pmatrix}$$

Lateral Coefficient	Formula
Y_v	$\frac{\rho S b v^*}{4m V_a^*} [C_{Y_p} p^* + C_{Y_r} r^*] + \frac{\rho S v^*}{m} [C_{Y_0} + C_{Y_\beta} \beta^* + C_{Y_{\delta_a}} \delta_a^* + C_{Y_{\delta_r}} \delta_r^*] + \frac{\rho S C_{Y_\beta}}{2m} \sqrt{u^{*2} + w^{*2}}$
Y_p	$\omega \frac{\rho V_a^{*2} S}{2m} C_{Y_{\delta_a}}$
Y_r	$\omega \frac{\rho V_a^{*2} S}{2m} C_{Y_{\delta_r}}$
Y_{δ_a}	
Y_{δ_r}	
L_v	$\frac{\rho S b^2 v^*}{4V_a^*} [C_{p_p} p^* + C_{p_r} r^*] + \rho S b v^* [C_{p_0} + C_{p_\beta} \beta^* + C_{p_{\delta_a}} \delta_a^* + C_{p_{\delta_r}} \delta_r^*] + \frac{\rho S b C_{p_\beta}}{2} \sqrt{u^{*2} + w^{*2}}$
L_p	$\Gamma_1 q^* + \frac{\rho V_a^* S b^2}{4} C_{p_p}$
L_r	$-\Gamma_2 q^* + \frac{\rho V_a^* S b^2}{4} C_{p_r}$
L_{δ_a}	$\frac{\rho V_a^{*2} S b}{2} C_{p_{\delta_a}}$
L_{δ_r}	$\frac{\rho V_a^{*2} S b}{2} C_{p_{\delta_r}}$
N_v	$\frac{\rho S b^2 v^*}{4V_a^*} [C_{r_p} p^* + C_{r_r} r^*] + \rho S b v^* [C_{r_0} + C_{r_\beta} \beta^* + C_{r_{\delta_a}} \delta_a^* + C_{r_{\delta_r}} \delta_r^*] + \frac{\rho S b C_{r_\beta}}{2} \sqrt{u^{*2} + w^{*2}}$
N_p	$\Gamma_7 q^* + \frac{\rho V_a^* S b^2}{4} C_{r_p}$
N_r	$-\Gamma_1 q^* + \frac{\rho V_a^* S b^2}{4} C_{r_r}$
N_{δ_a}	$\frac{\rho V_a^{*2} S b}{2} C_{r_{\delta_a}}$
N_{δ_r}	$\frac{\rho V_a^{*2} S b}{2} C_{r_{\delta_r}}$

dimensional stability derivatives

Lateral State-space Equations

$$\begin{pmatrix} \dot{\bar{v}} \\ \dot{\bar{p}} \\ \dot{\bar{r}} \\ \dot{\bar{\phi}} \\ \dot{\bar{\psi}} \end{pmatrix} = \begin{pmatrix} Y_v & Y_p & Y_r & A_{14} & 0 \\ L_v & L_p & L_r & 0 & 0 \\ N_v & N_p & N_r & 0 & 0 \\ 0 & 1 & A_{43} & A_{44} & 0 \\ 0 & 0 & A_{53} & A_{54} & 0 \end{pmatrix} \begin{pmatrix} \bar{v} \\ \bar{p} \\ \bar{r} \\ \bar{\phi} \\ \bar{\psi} \end{pmatrix} + \begin{pmatrix} Y_{\delta_a} & Y_{\delta_r} \\ L_{\delta_a} & L_{\delta_r} \\ N_{\delta_a} & N_{\delta_r} \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \bar{\delta}_a \\ \bar{\delta}_r \end{pmatrix}$$

Lateral Coefficient	Formula
A_{14}	$g \cos \theta^* \cos \phi^*$
A_{43}	$\cos \phi^* \tan \theta^*$
A_{44}	$q^* \cos \phi^* \tan \theta^* - r^* \sin \phi^* \tan \theta^*$
A_{53}	$\cos \phi^* \sec \theta^*$
A_{54}	$p^* \cos \phi^* \sec \theta^* - r^* \sin \phi^* \sec \theta^*$

Alternative Form - Lateral

The lateral dynamics are often expressed using β instead of v . Recall that

$$v = V_a \sin \beta$$

Therefore

$$\bar{v} = V_a^* \cos \beta^* \bar{\beta}$$

Differentiating gives

$$\dot{\bar{\beta}} = \frac{1}{V_a^* \cos \beta^*} \dot{\bar{v}}$$

Utilizing side-slip angle
instead of lateral velocity

The associated state space equations are

$$\begin{pmatrix} \dot{\bar{\beta}} \\ \dot{\bar{p}} \\ \dot{\bar{r}} \\ \dot{\bar{\phi}} \\ \dot{\bar{\psi}} \end{pmatrix} = \begin{pmatrix} Y_v & \frac{Y_p}{V_a^* \cos \beta^*} & \frac{Y_r}{V_a^* \cos \beta^*} & \frac{g \cos \theta^* \cos \phi^*}{V_a^* \cos \beta^*} & 0 \\ L_v V_a^* \cos \beta^* & L_p & L_r & 0 & 0 \\ N_v V_a^* \cos \beta^* & N_p & N_r & 0 & 0 \\ 0 & 1 & \cos \phi^* \tan \theta^* & q^* \cos \phi^* \tan \theta^* & 0 \\ 0 & 0 & \cos \phi^* \sec \theta^* & p^* \cos \phi^* \sec \theta^* & 0 \\ & & & -r^* \sin \phi^* \sec \theta^* & 0 \end{pmatrix} \begin{pmatrix} \bar{\beta} \\ \bar{p} \\ \bar{r} \\ \bar{\phi} \\ \bar{\psi} \end{pmatrix} + \begin{pmatrix} \frac{Y_{\delta_a}}{V_a^* \cos \beta^*} & \frac{Y_{\delta_r}}{V_a^* \cos \beta^*} \\ L_{\delta_a} & L_{\delta_r} \\ N_{\delta_a} & N_{\delta_r} \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \bar{\delta}_a \\ \bar{\delta}_r \end{pmatrix}$$

Longitudinal State-space Equations

The longitudinal states and inputs are defined as

$$\dot{x}_{\text{lon}} \triangleq (u, w, q, \theta, h)^\top \quad u_{\text{lon}} \triangleq (\delta_e, \delta_t)^\top$$

The nonlinear equations of motion are

$$\begin{aligned} \dot{u} &= -qw - g \sin \theta + \frac{\rho(u^2 + w^2)S}{2m} \left[C_{X_0} + C_{X_\alpha} \tan^{-1} \left(\frac{w}{u} \right) + C_{X_{\delta_e}} \delta_e \right] + \frac{\rho\sqrt{u^2 + w^2}S}{4m} C_{X_q} cq \\ &\quad + \frac{\rho S_{\text{prop}}}{2m} C_{\text{prop}} \left[(k\delta_t)^2 - (u^2 + w^2) \right] \\ \dot{w} &= qu + g \cos \theta + \frac{\rho(u^2 + w^2)S}{2m} \left[C_{Z_0} + C_{Z_\alpha} \tan^{-1} \left(\frac{w}{u} \right) + C_{Z_{\delta_e}} \delta_e \right] + \frac{\rho\sqrt{u^2 + w^2}S}{4m} C_{Z_q} cq \\ \dot{q} &= \frac{1}{2J_y} \rho(u^2 + w^2)cS \left[C_{m_0} + C_{m_\alpha} \tan^{-1} \left(\frac{w}{u} \right) + C_{m_{\delta_e}} \delta_e \right] + \frac{1}{4J_y} \rho\sqrt{u^2 + w^2} S C_{m_q} c^2 q \\ \dot{\theta} &= q \\ \dot{h} &= u \sin \theta - w \cos \theta \end{aligned}$$

where we have used $\alpha = \tan^{-1} \left(\frac{w}{u} \right)$ and $V_a = \sqrt{u^2 + w^2}$

Jacobian Matrices

$$A_{\text{lon}} = \frac{\partial f_{\text{lon}}}{\partial x_{\text{lon}}} = \begin{pmatrix} \frac{\partial \dot{u}}{\partial u} & \frac{\partial \dot{u}}{\partial w} & \frac{\partial \dot{u}}{\partial q} & \frac{\partial \dot{u}}{\partial \theta} & \frac{\partial \dot{u}}{\partial h} \\ \frac{\partial \dot{w}}{\partial u} & \frac{\partial \dot{w}}{\partial w} & \frac{\partial \dot{w}}{\partial q} & \frac{\partial \dot{w}}{\partial \theta} & \frac{\partial \dot{w}}{\partial h} \\ \frac{\partial \dot{q}}{\partial u} & \frac{\partial \dot{q}}{\partial w} & \frac{\partial \dot{q}}{\partial q} & \frac{\partial \dot{q}}{\partial \theta} & \frac{\partial \dot{q}}{\partial h} \\ \frac{\partial \dot{\theta}}{\partial u} & \frac{\partial \dot{\theta}}{\partial w} & \frac{\partial \dot{\theta}}{\partial q} & \frac{\partial \dot{\theta}}{\partial \theta} & \frac{\partial \dot{\theta}}{\partial h} \\ \frac{\partial \dot{h}}{\partial u} & \frac{\partial \dot{h}}{\partial w} & \frac{\partial \dot{h}}{\partial q} & \frac{\partial \dot{h}}{\partial \theta} & \frac{\partial \dot{h}}{\partial h} \end{pmatrix}$$

$$B_{\text{lon}} = \frac{\partial f_{\text{lon}}}{\partial u_{\text{lon}}} = \begin{pmatrix} \frac{\partial \dot{u}}{\partial \delta_e} & \frac{\partial \dot{u}}{\partial \delta_t} \\ \frac{\partial \dot{w}}{\partial \delta_e} & \frac{\partial \dot{w}}{\partial \delta_t} \\ \frac{\partial \dot{q}}{\partial \delta_e} & \frac{\partial \dot{q}}{\partial \delta_t} \\ \frac{\partial \dot{\theta}}{\partial \delta_e} & \frac{\partial \dot{\theta}}{\partial \delta_t} \\ \frac{\partial \dot{h}}{\partial \delta_e} & \frac{\partial \dot{h}}{\partial \delta_t} \end{pmatrix}$$

Take partial derivatives and evaluate them at trim state and trim input

Longitudinal State-space Equations

$$\begin{pmatrix} \dot{\bar{u}} \\ \dot{\bar{w}} \\ \dot{\bar{q}} \\ \dot{\bar{\theta}} \\ \dot{\bar{h}} \end{pmatrix} = \begin{pmatrix} X_u & X_w & X_q & -g \cos \theta^* & 0 \\ Z_u & Z_w & Z_q & -g \sin \theta^* & 0 \\ M_u & M_w & M_q & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ \sin \theta^* & -\cos \theta^* & 0 & u^* \cos \theta^* + w^* \sin \theta^* & 0 \end{pmatrix} \begin{pmatrix} \bar{u} \\ \bar{w} \\ \bar{q} \\ \bar{\theta} \\ \bar{h} \end{pmatrix} + \begin{pmatrix} X_{\delta_e} & X_{\delta_t} \\ Z_{\delta_e} & 0 \\ M_{\delta_e} & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \bar{\delta}_e \\ \bar{\delta}_t \end{pmatrix}$$

Longitudinal	Formula
X_u	$\frac{u^* \rho S}{m} [C_{X_0} + C_{X_\alpha} \alpha^* + C_{X_{\delta_e}} \delta_e^*] - \frac{\rho S w^* C_{X_\alpha}}{2m} + \frac{\rho S c C_{X_q} u^* q^*}{4m V_a^*} - \frac{\rho S_{prop} C_{prop} u^*}{m}$
X_w	$-q^* + \frac{w^* \rho S}{m} [C_{X_0} + C_{X_\alpha} \alpha^* + C_{X_{\delta_e}} \delta_e^*] + \frac{\rho S c C_{X_q} w^* q^*}{4m V_a^*} + \frac{\rho S C_{X_\alpha} u^*}{2m} - \frac{\rho S_{prop} C_{prop} w^*}{m}$
X_q	$-w^* + \frac{\rho V_a^* S C_{X_q} c}{4m}$
X_{δ_e}	$\frac{\rho V_a^{*2} S C_{X_{\delta_e}}}{2m}$
X_{δ_t}	Z_u dimensional stability derivatives
Z_u	$\frac{z_\alpha w^*}{2m} + \frac{u^* \rho S C_{Z_q} c q^*}{4m V_a^*}$
Z_w	$\frac{w^* \rho S}{m} [C_{Z_0} + C_{Z_\alpha} \alpha^* + C_{Z_{\delta_e}} \delta_e^*] + \frac{\rho S C_{Z_\alpha} u^*}{2m} + \frac{\rho w^* S c C_{Z_q} q^*}{4m V_a^*}$
Z_q	$u^* + \frac{\rho V_a^* S C_{Z_q} c}{4m}$
Z_{δ_e}	$\frac{\rho V_a^{*2} S C_{Z_{\delta_e}}}{2m}$
M_u	$\frac{u^* \rho S c}{J_y} [C_{m_0} + C_{m_\alpha} \alpha^* + C_{m_{\delta_e}} \delta_e^*] - \frac{\rho S c C_{m_\alpha} w^*}{2J_y} + \frac{\rho S c^2 C_{m_q} q^* u^*}{4J_y V_a^*}$
M_w	$\frac{w^* \rho S c}{J_y} [C_{m_0} + C_{m_\alpha} \alpha^* + C_{m_{\delta_e}} \delta_e^*] + \frac{\rho S c C_{m_\alpha} u^*}{2J_y} + \frac{\rho S c^2 C_{m_q} q^* w^*}{4J_y V_a^*}$
M_q	$\frac{\rho V_a^* S c^2 C_{m_q}}{4J_y}$
M_{δ_e}	$\frac{\rho V_a^{*2} S c C_{m_{\delta_e}}}{2J_y}$

Alternative Form – Longitudinal

The longitudinal dynamics are often expressed using α instead of w . Recall that

$$w = V_a \sin \alpha \cos \beta.$$

At trim, when $\beta = 0$ we have

$$\bar{w} = V_a^* \cos \alpha^* \bar{\alpha}.$$

Utilizing angle of attack
instead of downward velocity

Differentiating gives

$$\dot{\bar{\alpha}} = \frac{1}{V_a^* \cos \alpha^*} \dot{\bar{w}}.$$

The associated state space equations are

$$\begin{pmatrix} \dot{\bar{u}} \\ \dot{\bar{\alpha}} \\ \dot{\bar{q}} \\ \dot{\bar{\theta}} \\ \dot{\bar{h}} \end{pmatrix} = \begin{pmatrix} X_u & X_w V_a^* \cos \alpha^* & X_q & -g \cos \theta^* & 0 \\ \frac{Z_u}{V_a^* \cos \alpha^*} & Z_w & \frac{Z_q}{V_a^* \cos \alpha^*} & \frac{-g \sin \theta^*}{V_a^* \cos \alpha^*} & 0 \\ M_u & M_w V_a^* \cos \alpha^* & M_q & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ \sin \theta^* & -V_a^* \cos \theta^* \cos \alpha^* & 0 & u^* \cos \theta^* + w^* \sin \theta^* & 0 \end{pmatrix} \begin{pmatrix} \bar{u} \\ \bar{\alpha} \\ \bar{q} \\ \bar{\theta} \\ \bar{h} \end{pmatrix} + \begin{pmatrix} X_{\delta_e} & X_{\delta_t} \\ \frac{Z_{\delta_e}}{V_a^* \cos \alpha^*} & 0 \\ M_{\delta_e} & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \bar{\delta}_e \\ \bar{\delta}_t \end{pmatrix}$$

Numerical Computation of State Space Equations

Let $w = f(z)$ where $f : \mathbb{R}^n \mapsto \mathbb{R}^m$, i.e.,

$$\begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_m \end{pmatrix} = \begin{pmatrix} f_1(x_1, x_2, \dots, x_n) \\ f_2(x_1, x_2, \dots, x_n) \\ \vdots \\ f_m(x_1, x_2, \dots, x_n) \end{pmatrix}.$$

The Jacobian of f with respect to x is defined as

$$\frac{\partial f}{\partial x} = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & & & \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \cdots & \frac{\partial f_m}{\partial x_n} \end{pmatrix}.$$

Numerical Computation of State Space Equations

Let

$$\frac{\partial f}{\partial x} = \begin{pmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} & \cdots & \frac{\partial f}{\partial x_n} \end{pmatrix}, \quad \text{where} \quad \frac{\partial f}{\partial x_i} = \begin{pmatrix} \frac{\partial f_1}{\partial x_i} \\ \frac{\partial f_2}{\partial x_i} \\ \vdots \\ \frac{\partial f_n}{\partial x_i} \end{pmatrix}.$$

If ϵ is a small fixed number, then

$$\frac{\partial f}{\partial x_i}(x) \approx \frac{f(x + \epsilon e_i) - f(x)}{\epsilon},$$

which can be computed numerically, with two calls of $f(\cdot)$.

Note that computing $\frac{\partial f}{\partial x}(x)$ will require $n + 1$ functions calls to $f(\cdot)$.

Numerical Computation of State Space Equations

```
import numpy as np
def df_dx(x, z):
    # take partial of f(x, z) with respect to x
    eps = 0.001 # deviation
    A = np.zeros((m, n)) # Jacobian of f wrt x
    f_at_x = f(x, z)
    for i in range(0, n):
        x_eps = np.copy(x)
        x_eps[i][0] += eps # add eps to ith state
        f_at_x_eps = f(x_eps, z)
        df_dx_i = (f_at_x_eps - f_at_x) / eps
        A[:, i] = df_dx_i[:, 0]
    return A
```

Numerical Computation of State Space Equations

Define:

The Euler state: $x_e = (p_n, p_e, p_d, u, v, w, \phi, \theta, \psi, p, q, r)^\top \in \mathbb{R}^{12}$

The Quaternion state: $x_q = (p_n, p_e, p_d, u, v, w, e_0, e_x, e_y, e_z, p, q, r)^\top \in \mathbb{R}^{13}$

Let

$$\mathbf{p} = (p_n, p_e, p_d)^\top$$

$$\mathbf{v} = (u, v, w)^\top$$

$$\boldsymbol{\vartheta} = (\phi, \theta, \psi)^\top$$

$$\mathbf{q} = (e_0, e_1, e_2, e_3)^\top$$

$$\boldsymbol{\omega} = (p, q, r)^\top,$$

Then

$$x_e = (\mathbf{p}^\top, \mathbf{v}^\top, \boldsymbol{\vartheta}^\top, \boldsymbol{\omega}^\top)^\top$$

$$x_q = (\mathbf{p}^\top, \mathbf{v}^\top, \mathbf{q}^\top, \boldsymbol{\omega}^\top)^\top.$$

Numerical Computation of State Space Equations

Let $\vartheta = \Theta(\mathbf{q})$ convert quaternions to Euler angles, and

$\mathbf{q} = Q(\vartheta)$ converts Euler angles to quaternions.

Define

$$T_e \begin{pmatrix} \mathbf{p} \\ \mathbf{v} \\ \mathbf{q} \\ \boldsymbol{\omega} \end{pmatrix} \triangleq \begin{pmatrix} \mathbf{p} \\ \mathbf{v} \\ \Theta(\mathbf{q}) \\ \boldsymbol{\omega} \end{pmatrix}$$
$$T_q \begin{pmatrix} \mathbf{p} \\ \mathbf{v} \\ \vartheta \\ \boldsymbol{\omega} \end{pmatrix} \triangleq \begin{pmatrix} \mathbf{p} \\ \mathbf{v} \\ Q(\vartheta) \\ \boldsymbol{\omega} \end{pmatrix},$$

where

$$\frac{\partial T_e}{\partial x_q}(x_q) = \begin{pmatrix} I_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 4} & 0_{3 \times 3} \\ 0_{3 \times 3} & I_{3 \times 3} & 0_{3 \times 4} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & \frac{\partial \Theta}{\partial \mathbf{q}}(x_q) & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 4} & I_{3 \times 3} \end{pmatrix}.$$

Numerical Computation of State Space Equations

The Python dynamics file returns:

$$\dot{x}_q = f_q(x_q, u).$$

In terms of Euler states we have:

$$\begin{aligned}\dot{x}_e &= \frac{d}{dt}T_e(x_q) = \frac{\partial T_e}{\partial x_q}(x_q)\dot{x}_q = \frac{\partial T_e}{\partial x_q}(x_q)f_q(x_q, u) \\ &= \frac{\partial T_e}{\partial x_q}(T_q(x_e))f_q(T_q(x_e), u).\end{aligned}$$

If $\tilde{x}_e = x_e - x_e^*$, $\tilde{u} = u - u^*$, the state space equations are

$$\dot{\tilde{x}}_e = A\tilde{x}_e + B\tilde{u},$$

where

$$\begin{aligned}A &= \frac{\partial}{\partial x_e} \left(\frac{\partial T_e}{\partial x_q}(T_q(x_e))f_q(T_q(x_e), u) \right) \\ B &= \frac{\partial}{\partial u} \left(\frac{\partial T_e}{\partial x_q}(T_q(x_e))f_q(T_q(x_e), u) \right).\end{aligned}$$

Longitudinal State Space Equations

The longitudinal states and inputs are

$$x_{lon} = \begin{pmatrix} u \\ w \\ q \\ \theta \\ h \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} p_n \\ p_e \\ p_d \\ u \\ v \\ w \\ \phi \\ \theta \\ \psi \\ p \\ q \\ r \end{pmatrix} \triangleq E_1 x_e$$

$$u_{lon} = \begin{pmatrix} \delta_e \\ \delta_t \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \delta_t \\ \delta_e \\ \delta_a \\ \delta_r \end{pmatrix} \triangleq E_2 u.$$

Therefore the longitudinal state space equations are

$$\dot{x}_{lon} = A_{lon} x_{lon} + B_{lon} u_{lon}$$

where

$$A_{lon} = E_1 A E_1^\top$$

$$B_{lon} = E_1 B E_2^\top.$$

Lateral State Space Equations

The lateral states and inputs are

$$x_{lat} = \begin{pmatrix} v \\ p \\ r \\ \phi \\ \psi \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} p_n \\ p_e \\ p_d \\ u \\ v \\ w \\ \phi \\ \theta \\ \psi \\ p \\ q \\ r \end{pmatrix} \triangleq E_3 x_e$$

$$u_{lat} = \begin{pmatrix} \delta_a \\ \delta_r \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \delta_t \\ \delta_e \\ \delta_a \\ \delta_r \end{pmatrix} \triangleq E_4 u,$$

Therefore the lateral state space equations are

$$\dot{x}_{lat} = A_{lat} x_{lat} + B_{lat} u_{lat}$$

where

$$A_{lat} = E_3 A E_3^\top$$

$$B_{lat} = E_3 B E_4^\top.$$

Reduced Order Modes

Longitudinal Modes

- Short-period Mode
 - Short-period mode is the fast mode seen in pitch rate q and pitch angle
- Phugoid Mode
 - Phugoid mode is slow mode seen in pitch angle and u
- Induced by impulse on elevator

Lateral Modes

- Roll Mode
 - First order fast mode between aileron and roll rate
- Spiral-divergence Mode
 - First order (really) slow mode between aileron and yaw/course
- Dutch-roll Mode
 - Second order mode: coupling between roll, yaw, and side slip. Like a duck wagging its tail
- Induced by doublet on aileron or rudder

Simulation Project

- Find trim states and inputs. For the aerosonde model use $V_a = 25$ m/s, $\gamma = 0$, and $R = \infty$ m.
- Find the transfer functions derived in this chapter.
- Find the state space models derived in this chapter.
- Compute the eigenvalues of A_{lon} and A_{lat} and associate them with the phugoid mode, the short period mode, the roll mode, the dutch roll mode, and the spiral mode.