Chapter 5

Trim Aircraft at a Desired Flight Condition & Find the Linearized Model at the Trim

Equilibriums and Linearized Models

----- **For details See Appendix C in**

Introduction to Control Systems Engineering,
H. G. Kwatny and B. C. Chang, 1st Edition, 2021,
Cognella Academic Publishing.

Most of the systems to be controlled involve nonlinear dynamics, but most of the available control system design tools are linear approaches. Fortunately, linear control system design tools can still work very well with most of the nonlinear systems if the range of operation is reasonably linear. To design a linear controller for a nonlinear system, a general practice is to first identify an equilibrium point of interest with the nonlinear system so that a linearized state-space model can be1obtained.

Consider the nonlinear state-space model equations,

$$\begin{aligned} \dot{x} &= f(x,u), \quad x \in R^n, u \in R^m \\ y &= h(x,u), \quad y \in R^p \end{aligned} \tag{C.1}$$

Often, we are interested in motions near a particular equilibrium point, x^*, u^*, y^*. In such circumstances, a linear approximation to the equations is a useful first step in analysis and design. To obtain a linear approximate model, the local perturbation variables $\bar{x}(t)$, $\bar{u}(t)$, $\bar{y}(t)$ are defined by the following relations,

$$x(t) = x^* + \bar{x}(t), \quad u(t) = u^* + \bar{u}(t), \quad y(t) = y^* + \bar{y}(t) \tag{C.2}$$

Since x^*, u^*, y^* are constant vectors, Eqs. (C.1) and (C.2) will lead to the following

$$\begin{aligned} \dot{x}(t) &= \dot{\bar{x}}(t) = f(x^* + \bar{x}(t), u^* + \bar{u}(t)) \\ y(t) &= y^* + \bar{y}(t) = h(x^* + \bar{x}(t), u^* + \bar{u}(t)) \end{aligned} \tag{C.3}$$

Now, according to Taylor series expansion for f, g, we have

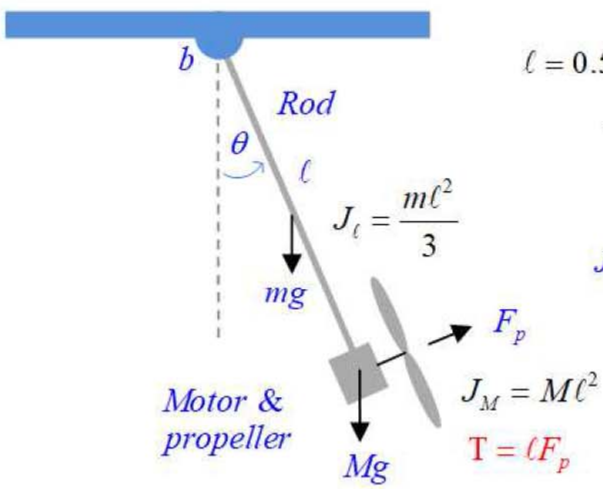
$$\begin{aligned} f(x^* + \bar{x}, u^* + \bar{u}) &= f(x^*, u^*) + \left.\frac{\partial f}{\partial x}\right|_{x^*,u^*} \bar{x} + \left.\frac{\partial f}{\partial u}\right|_{x^*,u^*} \bar{u} + \text{h.o.t.} \\ h(x^* + \bar{x}, u^* + \bar{u}) &= h(x^*, u^*) + \left.\frac{\partial h}{\partial x}\right|_{x^*,u^*} \bar{x} + \left.\frac{\partial h}{\partial u}\right|_{x^*,u^*} \bar{u} + \text{h.o.t.} \end{aligned} \tag{C.4}$$

Recall that $f(x^*, u^*) = 0$ and $h(x^*, u^*) = y^*$, so that upon dropping the higher order terms we have the following linearized state-space model,

$$\begin{aligned}\dot{\bar{x}} &= \left.\frac{\partial f}{\partial x}\right|_{x^*,u^*} \bar{x} + \left.\frac{\partial f}{\partial u}\right|_{x^*,u^*} \bar{u} := A\bar{x} + B\bar{u} \\ \bar{y} &= \left.\frac{\partial h}{\partial x}\right|_{x^*,u^*} \bar{x} + \left.\frac{\partial h}{\partial u}\right|_{x^*,u^*} \bar{u} := C\bar{x} + D\bar{u}\end{aligned} \tag{C.5}$$

These differentials are called ***Jacobian matrices***. Let us emphasize that the linear equations are only valid approximation in a sufficiently small neighborhood of the equilibrium point at which they are derived.

Example: A Nonlinear Lightly Damped Pendulum Positioning System



$\ell = 0.5\text{m}, \quad m = 0.1\text{kg}, \quad g = 9.81\text{m/s}^2$

$M = 0.1\text{kg}, \quad J_T = J_M + J_\ell$

friction coefficient $b = 4.5 \times 10^{-3}$ Nm/rad/s

$$J_T \ddot{\theta}(t) + b\dot{\theta}(t) + mg\frac{\ell}{2}\sin\theta(t) + Mg\ell\sin\theta(t) = \mathrm{T}(t)$$

$F_p(\omega)$, ω is the motor speed in rpm

$$0.0232\dot{\omega}(t) + \omega(t) = 47.22E(t),$$

$E(t)$ is the motor input voltage in V

Nonlinear State Equations and Equilibriums

$$\ddot{\theta}(t) + a_1\dot{\theta}(t) + a_0 \sin\theta(t) = b_0 \mathrm{T}(t)$$

where $a_1 = 0.135$, $a_0 = 22.073$, and $b_0 = 30$. Let the state variables be $x_1(t) = \theta(t)$ and $x_2(t) = \dot{\theta}(t)$, then the nonlinear state equation associated with Equation 10.3 can be written as

$$\dot{x} = \begin{bmatrix}\dot{x}_1 \\ \dot{x}_2\end{bmatrix} = \begin{bmatrix} x_2 \\ -a_0 \sin x_1 - a_1 x_2 + b_0 \mathrm{T}\end{bmatrix} = f(x, \mathrm{T}) = \begin{bmatrix} f_1(x_1, x_2, \mathrm{T}) \\ f_2(x_1, x_2, \mathrm{T})\end{bmatrix} \tag{10.4}$$

Assume the operating equilibrium is chosen to keep the angular displacement of the pendulum at $\theta(t) = \theta^* = 15^\circ = \pi/12$ rad. Then the equilibrium of the system can be found by solving the state equations with the derivative of the state variables set to zero. Now, we have

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = 0 \;\rightarrow\; \begin{array}{l} x_2 = 0 \\ -a_0 \sin 15^\circ - a_1 x_2 + b_0 \mathrm{T} = 0 \end{array} \;\rightarrow\; \begin{bmatrix} x_1^* \\ x_2^* \end{bmatrix} = \begin{bmatrix} 15^\circ \\ 0 \end{bmatrix}, \;\; \mathrm{T}^* = 0.19043\mathrm{Nm}$$

Linearized State-Space Model

Next, we will find a linearized state-space model for the $\theta^* = 15^\circ$ equilibrium. At this equilibrium $(x_1^*, x_2^*, \mathrm{T}^*) = (15^\circ, 0^\circ/\mathrm{s}, 0.19043\ \mathrm{Nm})$, we have the linearized state-space model

$$\dot{\bar{x}}(t) = A\bar{x}(t) + B\bar{\mathrm{T}}(t) \tag{10.5}$$

where the matrices A and B are computed via Jacobian matrices J_x and J_T, respectively, as follows:

$$A = J_x = \left[\frac{\partial f}{\partial x}\right]_{x^*,\mathrm{T}^*} = \begin{bmatrix} \partial f_1/\partial x_1 & \partial f_1/\partial x_2 \\ \partial f_2/\partial x_1 & \partial f_2/\partial x_2 \end{bmatrix}_{x^* = \begin{bmatrix} \pi/12 \\ 0 \end{bmatrix}, \mathrm{T}^*} = \begin{bmatrix} 0 & 1 \\ -a_0 \cos(\pi/12) & -a_1 \end{bmatrix}$$

and

$$B = J_\mathrm{T} = \left[\frac{\partial f}{\partial \mathrm{T}}\right]_{x^*,\mathrm{T}^*} = \begin{bmatrix} \partial f_1/\partial \mathrm{T} \\ \partial f_2/\partial \mathrm{T} \end{bmatrix}_{x^*,\mathrm{T}^*} = \begin{bmatrix} 0 \\ b_0 \end{bmatrix}$$

That is,

$$\dot{\bar{x}}(t) = \begin{bmatrix} \dot{\bar{x}}_1(t) \\ \dot{\bar{x}}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -21.32 & -0.135 \end{bmatrix} \begin{bmatrix} \bar{x}_1(t) \\ \bar{x}_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 30 \end{bmatrix} \bar{\mathrm{T}}(t) = A\bar{x}(t) + B\bar{\mathrm{T}}(\mathrm{t}) \tag{10.6}$$

Note that the relationship among the real values $x(t)$, $\mathrm{T(t)}$, the equilibrium values x^*, T^*, and the differential (perturbed) values $\bar{x}(t)$, $\bar{\mathrm{T}}$ are shown in the following:

$$x(t) = \bar{x}(t) + x^*, \quad \mathrm{T}(t) = \bar{\mathrm{T}}(t) + \mathrm{T}^* \tag{10.7}$$

For instance, $\bar{x}^T = \begin{bmatrix} 20^\circ & 0^\circ/\mathrm{s} \end{bmatrix}$ means that the real state vector is $x^T = \begin{bmatrix} 35^\circ & 0^\circ/\mathrm{s} \end{bmatrix}$, and a $\bar{\mathrm{T}} = 0$ Nm reveals that the real torque is $\mathrm{T} = \mathrm{T}^* = 0.19043$ Nm.

Analysis of the Open-Loop System

Recall that the eigenvalues of the A matrix, or the poles of the system are the roots of the following characteristic equation:

$$|sI - A| = \begin{vmatrix} s & -1 \\ 21.32 & s + 0.135 \end{vmatrix} = s^2 + 0.135s + 21.32 := s^2 + 2\varsigma\omega_n s + \omega_n^2 = 0 \tag{10.8}$$

The roots of this characteristic equation are

$$-\alpha \pm j\omega = -0.0675 \pm j4.6169$$

and the corresponding damping ratio and the natural frequency are

$$\varsigma = 0.0146 \quad \text{and} \quad \omega_n = 4.6174\ \text{rad/s}$$

respectively. We have learned from the previous chapters, especially Section 3.4.3, that the time-domain behavior of the system is closely related to the damping factor α, the frequency ω, the damping ratio ς, and the natural frequency ω_n, which are derived from the roots of the characteristic equation. Hence, it is possible to get a general idea of how the system will behave based on the information of the system poles, or, equivalently, the roots of the characteristic equation.

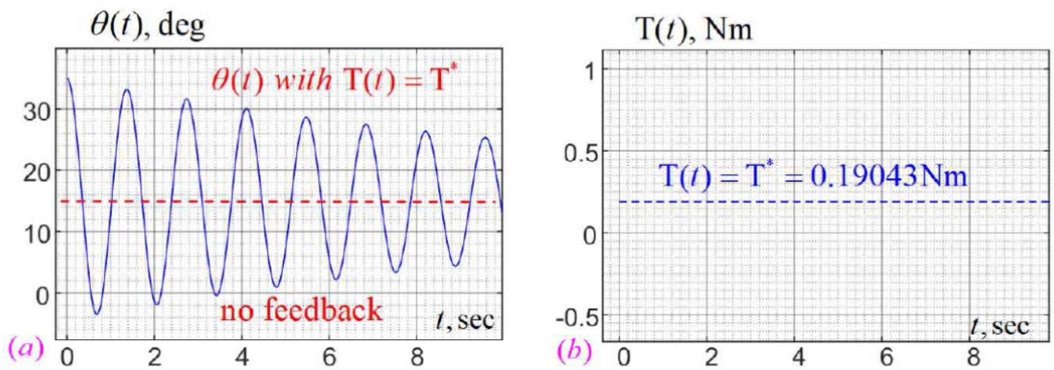


Fig. 10.2: An oscillatory time response of the nonlinear lightly damped pendulum system due to a perturbed initial condition.

Trim Aircraft at a Desired Flight Condition
& Find the Linearized Model at the Trim

directory:
C:\ …. \MATLAB\ F18_model_Teach\2024Summer_R2015b\2_F18full_DUtrim_R2015b

this folder includes:

0. aerodynamics_coefficients_rev2.mat
1. Trim_f18fullDU_a.m
2. f18full_DUtrim.mdl

The sequence to run the simulation is 1, which will call 2,

Objectives:
This program trims the aircraft at a desired flight condition, and find the linearized model at this trim.

Initialization:
Initialize to find Trim A, which is a straight level flight with 10 degree angle of attack.

```
%========================================================================
% Trimming initial conditions
%========================================================================

% State Initial Value
V       =  300;         % Airspeed , ft/s  Guess, not fixed
beta    =  0*d2r;       % Sideslip Angle, rad   Desired to be zero
alpha   =  10*d2r;      % Angle-of-attack, rad   Desired to be 10 deg

p       =  0*d2r;       % Roll rate, rad/s   Desired to be zero
q       =  0*d2r;       % Pitch rate, rad/s   Desired to be zero
r       =  0*d2r;       % Yaw rate, rad/s   Desired to be zero


phi     =  0*d2r;     % Roll Angle, rad   Desired to be zero
theta   =  10*d2r;    % Pitch Angle, rad   Desired to be 10 deg so that gamma can be zero
psi     =  0*d2r;     % Yaw Angle, rad   Guess, not fixed

pN      =  0;           % Position North,  ft   Not fixed
pE      =  0;           % Position East,  ft   Not fixed
h       =  25000;           % Altitude,  ft   Not fixed

% Stack Initial Condition for State
x_init = [V;beta;alpha;p;q;r;phi;theta;psi;pN;pE;h];
x = x_init;

% Initialize Input Value

d_STAB   = 0*d2r;    %Not fixed
d_AIL    = 0*d2r;    %Not fixed
d_RUD    = 0*d2r;    %Not fixed
T = 5000;    %Not fixed
```

```
u_init = [d_AIL; d_RUD; d_STAB; T];
```

Trim Simulations

Run

```
% Trim_f18fullDU_a.m
% Objectives:
% This file trims the aircraft at a desired flight
% condition, find the linearized model at this trim,
% and design an LQR stabilizing controller for the trim.

% Unit Conversion : Degree <--> Radian
d2r = pi/180;
r2d = 1/d2r;

h0=25000;
%==========================================================================
% F/A-18 data
%
% Aircraft Physical Paramters
% Reference: S. B. Buttrill and P. D. Arbuckle  and K. D. Hoffler
%            Simulation model of a twin-tail, high performance airplane
%            NASA 1992, NASA TM-107601

S = 400;                % ft^2
b =  37.4;                % ft
c =  11.52;             % ft
rho = 1.0660e-003;      % slugs/ft^3  --- 25C / 25000 ft
Ixx = 23000;            % slugs*ft^2
Iyy = 151293;           % slugs*ft^2
Izz = 169945;           % slugs*ft^2
Ixz = - 2971;           % slugs*ft^2
m = 1034.5;             % slugs
g = 32.2;               % ft/s^2

%==========================================================================
% Statename , Inputname and Outputname
%
% Baseline and Revised has different feedback channels
% So, their output equations are different

statenames = {'V (ft/s)','Beta (rad)','Alpha (rad)','Roll Rate (rad/s)',...
    'Pitch Rate (rad/s)','Yaw Rate (rad/s)','Phi (rad)','Theta (rad)',...
    'Yaw (rad)','pN (ft)','pE (ft)','h (ft)'}'
inputnames = { 'Aileron (rad)','Rudder (rad)','Stabilator (rad)', 'T (lbf)'}'
%
%==========================================================================
% load aerodynamics coefficents
% the aerodynamic data is resulted from the Minnesotadata rev2.
load aerodynamics_coefficients_rev2

%==========================================================================
% Trimming initial conditions
%==========================================================================
```

```
%--------------------------------------------
% Initialization
% Initialize to find TrimA condition

% State Initial Value
V       =  300;        % Airspeed , ft/s  Guess, not fixed
beta    =  0*d2r;      % Sideslip Angle, rad   Desired to be zero
alpha   =  10*d2r;     % Angle-of-attack, rad   Desired to be 10 deg

p       =  0*d2r;      % Roll rate, rad/s   Desired to be zero
q       =   0*d2r;     % Pitch rate, rad/s   Desired to be zero
r       =  0*d2r;      % Yaw rate, rad/s   Desired to be zero


phi     =  0*d2r;     % Roll Angle, rad   Desired to be zero
theta   =  10*d2r;    % Pitch Angle, rad   Desired to be 10 deg so that gamma can be zero
psi     =  0*d2r;     % Yaw Angle, rad   Guess, not fixed

pN      =  0;          % Position North,  ft   Not fixed
pE      =  0;          % Position East,  ft   Not fixed
h       =  25000;      % Altitude,  ft   Not fixed

% Stack Initial Condition for State
x_init = [V;beta;alpha;p;q;r;phi;theta;psi;pN;pE;h];
x = x_init;

% Initialize Input Value

d_STAB   = 0*d2r;    %Not fixed
d_AIL    = 0*d2r;    %Not fixed
d_RUD    = 0*d2r;    %Not fixed
T = 5000;    %Not fixed

u_init = [d_AIL; d_RUD; d_STAB; T];
%u=u_init
disp('initial values')
x_init(1)
x_init(2:9)*r2d
x_init(10:11)
x_init(12)
u_init(1:3)*r2d
u_init(4)

%==========================================================================
% Operating Point Specificaton Setup

open('f18full_DUtrim')
opys = operspec('f18full_DUtrim');

opys.States(1).Known  =  0;    %Not fixed
opys.States(2).Known  =  1;    %Desired to be the assigned value
opys.States(3).Known  =  1;
%opys.States(2).Known  =  0;
%opys.States(3).Known  =  0;

opys.States(4).Known  =  1;    %Desired to be the assigned value
opys.States(5).Known  =  1;    %Desired to be the assigned value
opys.States(6).Known  =  1;    %Desired to be the assigned value
```

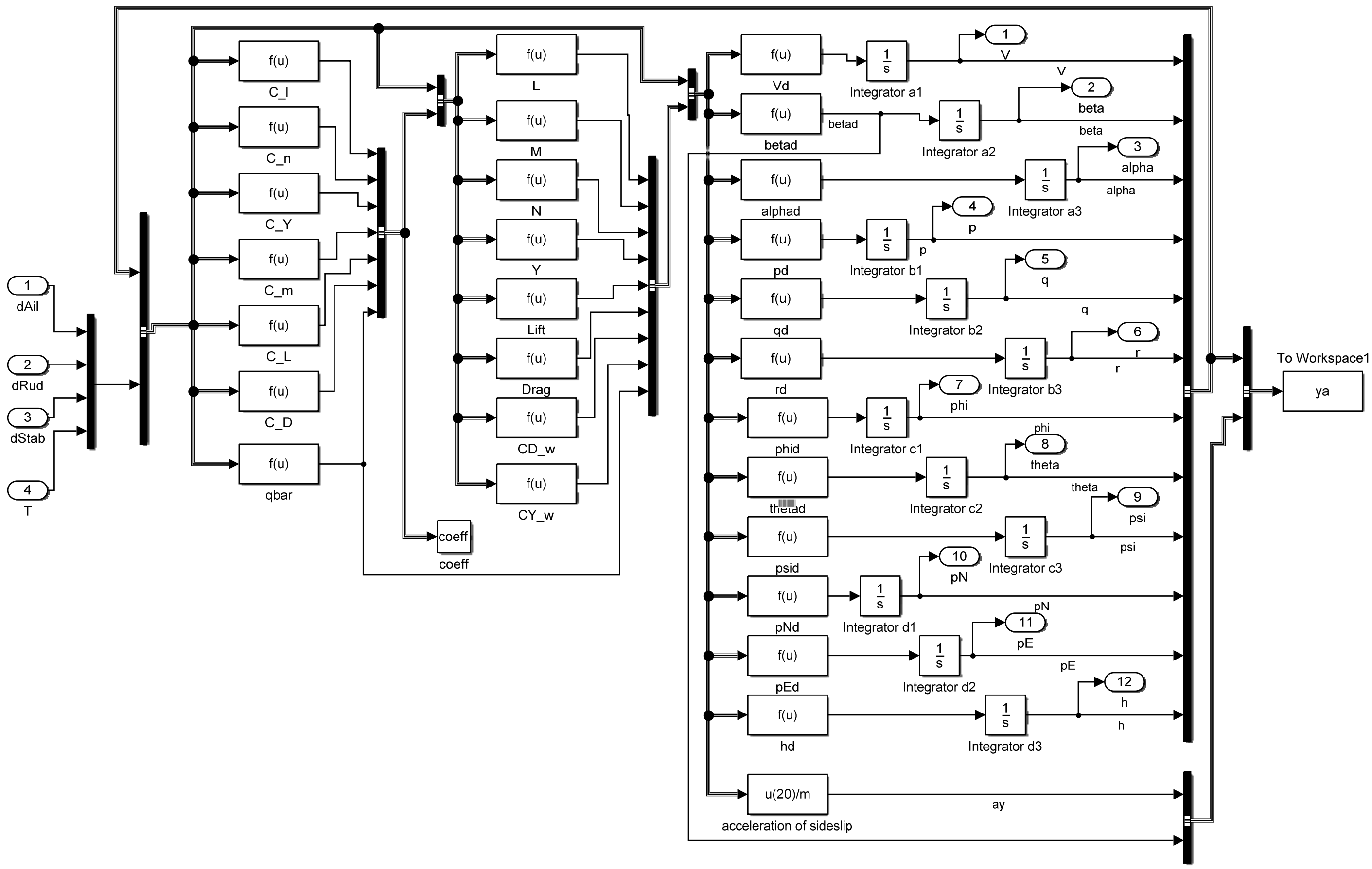
```
%opys.States(4).Known  =  0;
%opys.States(5).Known  =  0;
%opys.States(6).Known  =  0;

opys.States(7).Known  =  1;
opys.States(8).Known  =  1;
%opys.States(9).Known  =  1;
%opys.States(7).Known  =  0;
%opys.States(8).Known  =  0;
opys.States(9).Known  =  0;

opys.States(10).Known  =  0;
opys.States(11).Known  =  0;
opys.States(12).Known  =  0;


opys.States(1).steadystate = 1;
opys.States(2).steadystate = 1;
opys.States(3).steadystate = 1;
opys.States(4).steadystate = 1;
opys.States(5).steadystate = 1;
opys.States(6).steadystate = 1;
opys.States(7).steadystate = 1;
opys.States(8).steadystate = 1;
opys.States(9).steadystate = 1;
opys.States(10).steadystate = 0;
opys.States(11).steadystate = 0;
opys.States(12).steadystate = 0;


%Setting the Input value

opys.inputs(1).known = 0;
opys.inputs(2).known = 0;
opys.inputs(3).known = 0;
opys.inputs(4).known = 0;

opys.inputs(3).u = d_STAB;
opys.inputs(2).u = d_RUD;
opys.inputs(1).u = d_AIL;
opys.inputs(4).u = T;

opys.inputs(4).min = 0;
opys.inputs(4).max = 38000;
%==========================================================================
% Finding Trim/ Operating point
opt1 = optimset('MaxFunEvals',1e+04);
opt = linoptions('OptimizationOptions',opt1);
[ysop,rep] = findop('f18full_DUtrim',opys,opt);
get(ysop)



%==========================================================================
% Extracting Trim Point

x_trim = [ysop.States(1).x;  ysop.States(2).x; ysop.States(3).x;...
          ysop.States(4).x;  ysop.States(5).x; ysop.States(6).x;...
```

```
          ysop.States(7).x;  ysop.States(8).x; ysop.States(9).x;...
          ysop.States(12).x]

u_trim = [ysop.Inputs(1).u; ysop.Inputs(2).u; ysop.Inputs(3).u; ysop.Inputs(4).u]

disp('Trimmed Value')
x_trim(1)
x_trim(2:9)*r2d
x_trim(10)
u_trim(1:3)*r2d
u_trim(4)

Trim.T      = u_trim(4); % lbf
Trim.elev   = u_trim(3); % -0.0328 rad = -1.88 deg
Trim.ail    = u_trim(1); %rad
Trim.rud    = u_trim(2); %rad

Trim.V      = x_trim(1);   %ft/s
Trim.alpha  = x_trim(3);   %rad
Trim.q      = 0;        %rad/s
Trim.theta  = x_trim(8);   %rad
Trim.h      = h0;      %ft

Trim.beta   = 0;   %rad
Trim.p      = 0;   %rad/s
Trim.r      = 0;   %rad/s
Trim.phi    = 0;   %rad

Trim.psi    = 0;   %rad/s
Trim.pN     = 0;   %ft
Trim.pE     = 0;    %ft

%===========================================================================
% Creating Open Loop Linearized Model

[A ,B ,C, D] = linmod('f18full_DUtrim',x_trim,u_trim);
A_trim = A([1:8], [1:8]);
B_trim = B([1:8],:);
C_trim = C([1:8], [1:8]);
D_trim = D([1:8],:);

A_longltrl = A([1 3 5 8 2 4 6 7], [1 3 5 8 2 4 6 7])
B_longltrl = B([1 3 5 8 2 4 6 7], [4 3 1 2])
C_longltrl = C([1 3 5 8 2 4 6 7], [1 3 5 8 2 4 6 7])
D_longltrl = D([1 3 5 8 2 4 6 7], [4 3 1 2])

A_longltrl9 = A([1 3 5 8 12 2 4 6 7], [1 3 5 8 12 2 4 6 7])
B_longltrl9 = B([1 3 5 8 12 2 4 6 7], [4 3 1 2])
C_longltrl9 = C([1 3 5 8 12 2 4 6 7], [1 3 5 8 12 2 4 6 7])
D_longltrl9 = D([1 3 5 8 12 2 4 6 7], [4 3 1 2])


%===========================================================================
% Decoupled longitudinal
% Longitudinal states [ V  alpha q theta ]
% Longitudinal controls [ T  d_STAB]
display('Longitudnal states: [V alpha q theta ]')
display('Longitudinal controls [ T  d_STAB]')
```

```
A_x = A_longltrl([1:4], [1:4])
B_x = B_longltrl([1:4], [1:2])

display('Longitudnal states: [V alpha q theta h ]')
display('Longitudinal controls [ T  d_STAB]')
A5_x = A_longltrl9([1:5], [1:5])
B5_x = B_longltrl([1:5], [1:2])

% Lateral states [ beta p r phi]
% Lateral controls [ d_AIL  d_RUD]
display('Lateral states: [beta p r phi]')
display('Lateral controls [ d_AIL  d_RUD]')

A_y = A_longltrl([5:8], [5:8])
B_y = B_longltrl([5:8], [3:4])

display('eigenvalues of A_longltrl')
eig(A_longltrl)
display('eigenvalues of A_longltrl9')
eig(A_longltrl9)
display('eigenvalues of A_x')
eig(A_x)
display('eigenvalues of A5_x')
eig(A5_x)
display('eigenvalues of A_y')
eig(A_y)



save f18trim Trim A_x B_x A5_x B5_x A_y B_y
```

The following mdl simulink program will automatically be called and run until the simulation is done.

f18full_DUtrim.mdl


1
dAil
2
dRud
3
dStab
4
T
f(u)
C_l
C_n
C_Y
C_m
C_L
C_D
qbar
L
M
N
Y
Lift
Drag
CD_w
CY_w
coeff
coeff
Vd
betad
alphad
pd
qd
rd
phid
thetad
psid
pNd
pEd
hd
u(20)/m
acceleration of sideslip
1/s
Integrator a1
Integrator a2
Integrator a3
Integrator b1
Integrator b2
Integrator b3
Integrator c1
Integrator c2
Integrator c3
Integrator d1
Integrator d2
Integrator d3
V
beta
alpha
p
q
r
phi
theta
psi
pN
pE
h
ay
To Workspace1
ya

```
>> Trim_f18fullDU_a
statenames =
    'V (ft/s)'
    'Beta (rad)'
    'Alpha (rad)'
    'Roll Rate (rad/s)'
    'Pitch Rate (rad/s)'
    'Yaw Rate (rad/s)'
    'Phi (rad)'
    'Theta (rad)'
    'Yaw (rad)'
    'pN (ft)'
    'pE (ft)'
    'h (ft)'
inputnames =
    'Aileron (rad)'
    'Rudder (rad)'
    'Stabilator (rad)'
    'T (lbf)'
initial values
ans =
   300
ans =
     0
    10
     0
     0
     0
     0
    10
     0
ans =
     0
     0
ans =
       25000
ans =
     0
     0
     0
ans =
        5000

Warning: The command linoptions is obsolete. Use linearizeOptions or findopOptions
instead.
> In linoptions (line 131)
  In Trim_f18fullDU_a (line 156)

Local minimum found that satisfies the constraints.

Optimization completed because the objective function is non-decreasing in
feasible directions, to within the default value of the function tolerance,
and constraints are satisfied to within the default value of the constraint tolerance.

<stopping criteria details>


 Operating Point Search Report:
```

```
--------------------------------

 Operating Report for the Model f18full_DUtrim.
 (Time-Varying Components Evaluated at time t=0)

Operating point specifications were successfully met.
States:
----------
(1.) f18full_DUtrim/Integrator a1
      x:           436      dx:     -4.56e-08 (0)
(2.) f18full_DUtrim/Integrator a2
      x:             0      dx:             0 (0)
(3.) f18full_DUtrim/Integrator a3
      x:         0.175      dx:     -1.02e-06 (0)
(4.) f18full_DUtrim/Integrator b1
      x:             0      dx:             0 (0)
(5.) f18full_DUtrim/Integrator b2
      x:             0      dx:      8.83e-09 (0)
(6.) f18full_DUtrim/Integrator b3
      x:             0      dx:             0 (0)
(7.) f18full_DUtrim/Integrator c1
      x:             0      dx:             0 (0)
(8.) f18full_DUtrim/Integrator c2
      x:         0.175      dx:             0 (0)
(9.) f18full_DUtrim/Integrator c3
      x:             0      dx:             0 (0)
(10.) f18full_DUtrim/Integrator d1
      x:             0      dx:           436
(11.) f18full_DUtrim/Integrator d2
      x:             0      dx:             0
(12.) f18full_DUtrim/Integrator d3
      x:       2.5e+04      dx:             0

Inputs:
----------
(1.) f18full_DUtrim/dAil
      u:             0    [-Inf Inf]
(2.) f18full_DUtrim/dRud
      u:             0    [-Inf Inf]
(3.) f18full_DUtrim/dStab
      u:        -0.022    [-Inf Inf]
(4.) f18full_DUtrim/T
      u:      5.47e+03    [0 3.8e+04]

Outputs:
----------
(1.) f18full_DUtrim/V
      y:           436    [-Inf Inf]
(2.) f18full_DUtrim/beta
      y:             0    [-Inf Inf]
(3.) f18full_DUtrim/alpha
      y:         0.175    [-Inf Inf]
(4.) f18full_DUtrim/p
      y:             0    [-Inf Inf]
(5.) f18full_DUtrim/q
      y:             0    [-Inf Inf]
(6.) f18full_DUtrim/r
      y:             0    [-Inf Inf]
```

```
(7.) f18full_DUtrim/phi
      y:               0    [-Inf Inf]
(8.) f18full_DUtrim/theta
      y:           0.175    [-Inf Inf]
(9.) f18full_DUtrim/psi
      y:               0    [-Inf Inf]
(10.) f18full_DUtrim/pN
      y:               0    [-Inf Inf]
(11.) f18full_DUtrim/pE
      y:               0    [-Inf Inf]
(12.) f18full_DUtrim/h
      y:         2.5e+04    [-Inf Inf]

      Model: 'f18full_DUtrim'
     States: [12x1 opcond.StatePoint]
     Inputs: [4x1 opcond.InputPoint]
       Time: 0
    Version: 2
x_trim =
   1.0e+04 *
    0.0436
         0
    0.0000
         0
         0
         0
         0
    0.0000
         0
    2.5000
u_trim =
   1.0e+03 *
         0
         0
   -0.0000
    5.4705
```

Trimmed Value

```
ans =
  435.9249
ans =
     0
    10
     0
     0
     0
     0
    10
     0
ans =
       25000
ans =
         0
         0
   -1.2616
ans =
   5.4705e+03
```

```
Warning: Model 'f18full_DUtrim' is using a default value of 0.2 for maximum step size.
You can disable this
diagnostic by setting 'Automatic solver parameter selection' diagnostic to 'none' in the
Diagnostics page of the
configuration parameters dialog
> In dlinmod (line 195)
  In linmod (line 59)
  In Trim_f18fullDU_a (line 202)
Warning: Extra states are being set to zero.
> In DAStudio.warning (line 28)
  In dlinmod (line 217)
  In linmod (line 59)
  In Trim_f18fullDU_a (line 202)
A_longltrl =
   -0.0239  -28.3172         0  -32.2000         0         0         0         0
   -0.0003   -0.3621    1.0000         0         0         0         0         0
    0.0000   -2.2115   -0.2532         0         0         0         0         0
         0         0    1.0000         0         0         0         0         0
         0         0         0         0   -0.0374    0.1736   -0.9848    0.0727
         0         0         0         0   -8.5430   -0.8883    0.8762         0
         0         0         0         0    0.8860    0.0399   -0.1895         0
         0         0         0         0         0    1.0000    0.1763         0
B_longltrl =
    0.0010   -3.8114         0         0
   -0.0000   -0.0515         0         0
         0   -2.8791         0         0
         0         0         0         0
         0         0   -0.0149    0.0207
         0         0    8.3321    0.9541
         0         0   -0.0420   -0.6277
         0         0         0         0
C_longltrl =
     1     0     0     0     0     0     0     0
     0     1     0     0     0     0     0     0
     0     0     1     0     0     0     0     0
     0     0     0     1     0     0     0     0
     0     0     0     0     1     0     0     0
     0     0     0     0     0     1     0     0
     0     0     0     0     0     0     1     0
     0     0     0     0     0     0     0     1
D_longltrl =
     0     0     0     0
     0     0     0     0
     0     0     0     0
     0     0     0     0
     0     0     0     0
     0     0     0     0
     0     0     0     0
     0     0     0     0

A_longltrl9 =
   -0.0239  -28.3172         0  -32.2000         0         0         0         0         0
   -0.0003   -0.3621    1.0000         0         0         0         0         0         0
    0.0000   -2.2115   -0.2532         0         0         0         0         0         0
         0         0    1.0000         0         0         0         0         0         0
         0 -435.9249         0  435.9249         0         0         0         0         0
         0         0         0         0         0   -0.0374    0.1736   -0.9848    0.0727
         0         0         0         0         0   -8.5430   -0.8883    0.8762         0
```

```
        0         0         0         0         0    0.8860    0.0399   -0.1895         0
        0         0         0         0         0         0    1.0000    0.1763         0
B_longltrl9 =
    0.0010   -3.8114         0         0
   -0.0000   -0.0515         0         0
         0   -2.8791         0         0
         0         0         0         0
         0         0         0         0
         0         0   -0.0149    0.0207
         0         0    8.3321    0.9541
         0         0   -0.0420   -0.6277
         0         0         0         0
C_longltrl9 =
     1     0     0     0     0     0     0     0     0
     0     1     0     0     0     0     0     0     0
     0     0     1     0     0     0     0     0     0
     0     0     0     1     0     0     0     0     0
     0     0     0     0     1     0     0     0     0
     0     0     0     0     0     1     0     0     0
     0     0     0     0     0     0     1     0     0
     0     0     0     0     0     0     0     1     0
     0     0     0     0     0     0     0     0     1
D_longltrl9 =
     0     0     0     0
     0     0     0     0
     0     0     0     0
     0     0     0     0
     0     0     0     0
     0     0     0     0
     0     0     0     0
     0     0     0     0
     0     0     0     0
```

Longitudnal states: [V alpha q theta]
Longitudinal controls [T d_STAB]

```
A_x =
   -0.0239  -28.3172         0  -32.2000
   -0.0003   -0.3621    1.0000         0
    0.0000   -2.2115   -0.2532         0
         0         0    1.0000         0
B_x =
    0.0010   -3.8114
   -0.0000   -0.0515
         0   -2.8791
         0         0
```

Longitudnal states: [V alpha q theta h]
Longitudinal controls [T d_STAB]

```
A5_x =
   -0.0239  -28.3172         0  -32.2000         0
   -0.0003   -0.3621    1.0000         0         0
    0.0000   -2.2115   -0.2532         0         0
         0         0    1.0000         0         0
         0 -435.9249         0  435.9249         0
B5_x =
    0.0010   -3.8114
   -0.0000   -0.0515
         0   -2.8791
         0         0
         0         0
```

```
Lateral states: [beta p r phi]
Lateral controls [ d_AIL  d_RUD]
A_y =
   -0.0374    0.1736   -0.9848    0.0727
   -8.5430   -0.8883    0.8762         0
    0.8860    0.0399   -0.1895         0
         0    1.0000    0.1763         0
B_y =
   -0.0149    0.0207
    8.3321    0.9541
   -0.0420   -0.6277
         0         0
eigenvalues of A_longltrl
ans =
  -0.3094 + 1.4799i
  -0.3094 - 1.4799i
  -0.0101 + 0.1008i
  -0.0101 - 0.1008i
  -0.2873 + 1.4530i
  -0.2873 - 1.4530i
  -0.4888 + 0.0000i
  -0.0518 + 0.0000i
eigenvalues of A_longltrl9
ans =
   0.0000 + 0.0000i
  -0.3094 + 1.4799i
  -0.3094 - 1.4799i
  -0.0101 + 0.1008i
  -0.0101 - 0.1008i
  -0.2873 + 1.4530i
  -0.2873 - 1.4530i
  -0.4888 + 0.0000i
  -0.0518 + 0.0000i
```

eigenvalues of A_x **Poles of the Longitudinal Dynamics**

```
ans =
  -0.3094 + 1.4799i
  -0.3094 - 1.4799i
  -0.0101 + 0.1008i
```

$\rightarrow \quad \omega = 0.1008 \text{ rad/s}, \ \text{period } T = 2\pi / \omega = 62 \text{ s}$

```
  -0.0101 - 0.1008i
```

$\omega_n = 0.1013 \text{ rad/s}, \ \varsigma = 0.0997 \quad \rightarrow \quad \max OS = e^{-\varsigma\pi/\sqrt{1-\varsigma^2}} = 0.73 \ = 73\%$

```
eigenvalues of A5_x
ans =
   0.0000 + 0.0000i
  -0.3094 + 1.4799i
  -0.3094 - 1.4799i
  -0.0101 + 0.1008i
  -0.0101 - 0.1008i
```

eigenvalues of A_y **Poles of the Lateral Dynamics**

```
ans =

  -0.2873 + 1.4530i
```

$\rightarrow \quad \omega = 1.453 \text{ rad/s}$

$\omega_n = 1.4811 \text{ rad/s}, \ \varsigma = 0.1939$

```
  -0.2873 - 1.4530i
  -0.4888 + 0.0000i
  -0.0518 + 0.0000i
>>
```