APPENDIX

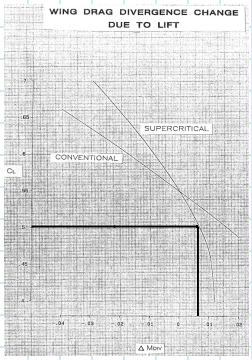
Hand Calculations

Aircraft Sizing Calculations (Compiled)

Friday, May 6, 2022 5:06 PM

1) $M = 0.82$ @ 35000ft
   $\Lambda = 35$
   $AR = 8$

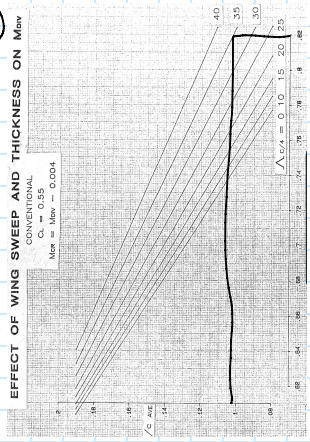
2) Assume $C_L = 0.54$, conventional



Find ΔM_{div} depending on conventional or sc

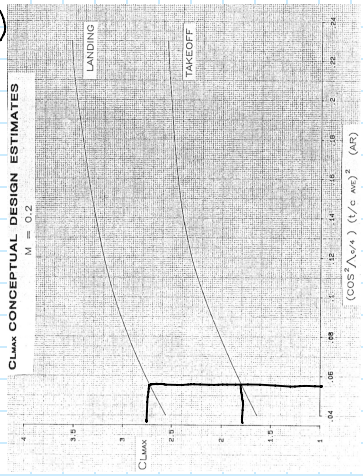
3) $\Delta M_{div} = (M_{crit} + .004) - \Delta M_{drv}$
$= .0034$

4)



Calculate t/c_{avg} based on Λ and M_{div}, $t/c = .1020$

5)



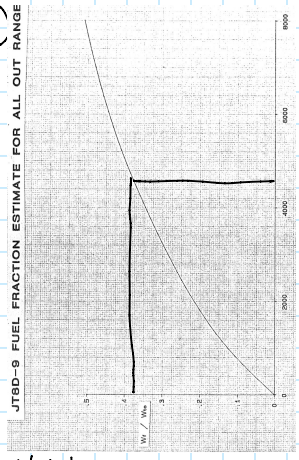
Find both $C_{L_{max}}$s, $C_{L_{max_{landing}}} = 2.73$, $C_{L_{max_{to}}} = 1.797$

6) $W/S|_{landing} = \left(\frac{V_{ap}}{1.3}\right)^2 \frac{\sigma (C_{L_{max}}|_{LOC})}{296} = 94.8$

7) $V_{cruise} = M \cdot a = 472.6$

$R_{A/6} = R_{calculated} + 200 + (V \cdot \frac{3}{4}) = 4555$

8)



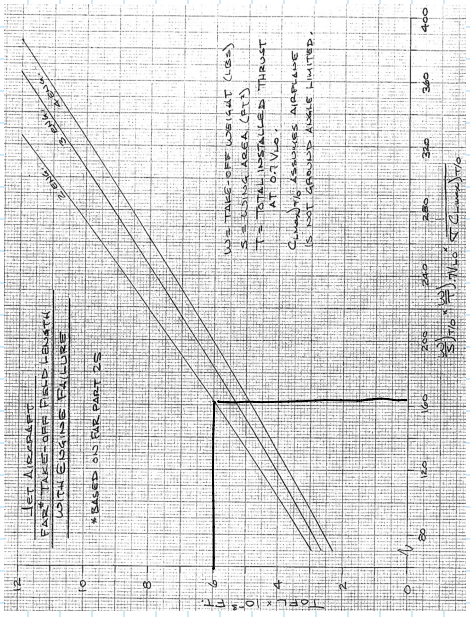
$W_f/W_{to}|_{JT8D} = 0.3744$

9) $W_f/W_{to}|_{JT9D} = W_f/W_{to}|_{JT8D} \cdot \frac{.61}{.78} = 0.2928$

10) $W/S|_{To} = W/S|_{to} / (1 - W_f/W_{to}) = 134$

11) $W/S|_{ic} = .965 \cdot W/S|_{to} = 129$

12) $C_{L_{ic}} = W/S|_{ic} / (1481 \cdot \delta \cdot M^2) = 0.55$, good enough



$k = 162$

$W/T_{0.7V_{to}} = \left(\frac{k}{W/S_{to}}\right) \cdot \sigma \cdot C_{L_{max_{to}}}$

$V_{to} = 1.2 \cdot \sqrt{\frac{296 \cdot W/S_{to}}{\sigma C_{L_{max_{to}}}}} = 182.8$

$M_{to} = \left(\frac{189}{661}\right) / \sqrt{\sigma} = 0.29$

$W/T = W/T_{0.7V_{to}} \cdot T_{A_{to}} / T_{db} = 1.69$

$T_{A_{to}} = 37200$

$T_{db} = 45500$

$W/T = {}^{W/T_{cruise}} \cdot T_{max}/T_{sls} = 1.69$

$T_{max} = 37200$

$T_{sls} = 45500$

$AR = 7$

$SWEEP = 32$

$taper = 0.35$

$t/c = 0.096$

Weight

Wing

$$W_w = \frac{.00945\, AR^{.8}(1+\lambda)^{.25} k_w \sqrt{3.75\ W_{to}}^{1.195}}{(t/c)^{.4}\cos\Lambda\ (W/S)^{.645}}, \quad t/c = t/c + .03$$

$$= .0085\, W_{to}^{1.195}$$

Fuselage

$L_f = 3.76\,(PAX/Abreast) + 33.2 = 127.2$

$D_f = (1.75\ abreast) + (1.58\ aisles) + 1 = 18.16$

$W_f = .6727 \cdot k_f \cdot L_f^{.6} \cdot D_f^{.72} \cdot W_{to}^{.235} = 1.7e3\, W_{to}^{.235}$

Landing Gear

$W_l = 0.04 \cdot W_{to} = .04 W_{to}$

Nacelle/Pylon

$W_{np} = 0.0555/(W/T) \cdot W_{to} = .0338 W_{to}$

Tail Surface

$W_{ts} = k_{ts} \cdot W_w = .0014 W_w$

Power Plant

$W_{pp} = 1/(3.58 \cdot (W/T)) \cdot W_{to} = .1699 W_{to}$

Fuel

$W_f = 1.0275 \cdot \left.\frac{W_f}{W_{to}}\right|_{TMO} = .313\, W_{to}$

Payload

$W_{pl} = 215\,PAX + W_c, \quad W_c = 4000$

$= 47000$

Fixed Equipment

$W_{fe} = (132\,PAX) + (300 \cdot N_{engine}) + (260\, N_{fc}) + (170\, N_{stew}) + .035\, W_{to} = 28370 + .035 W_{to}$

$W = W_w + W_{ts} + W_{fe} + W_l + W_{np} + W_{pp} + W_f + W_{fe} - 1 + W_{pl} = 0$

solve for W_{to}

$W_{to} = 3.85e5$

Sizing

$S = W_{to}/(W/S_{to}) = 2.93e3$

$b = \sqrt{AR \cdot S} = 143.2$

$\bar{c} = S/b = 20.46$

$T_{total} = W_{to}/(W/T) = 2.34e5$

$T_{engine} = T_{total}/N_{engine} = 1.17e5$

Drag

$Re_L = 2.852e6 \cdot \frac{1}{2} = 1426000$

Wing

$Re_w = Re_L \cdot \bar{c} = 2.9e7$

$C_{f_w} = .0027$

Wing

$Re_w = Re_L \cdot \bar{c} = 2.9e7$

$C_f = .0027$

$S_{wetw} = 2 \cdot S \cdot 1.02 = 5.97e3$

$z_w = \frac{(2-M^2)\cos\Lambda}{\sqrt{1-(M^2\cos\Lambda)}} = 1.717$

$k_w = 1 + (z_w \cdot t/c) + 100\,(t/c)^4 = 1.17$

$f_w = k_w C_{fw} S_{wetw} = 19.1$

Fuselage

$S_{wetfus} = 0.9 \cdot \pi \cdot D_f \cdot L_f = 6.53e3$

$Re_{fus} = Re_L \cdot L_f = 1.81e8$

$C_{ffus} = .002$

$fine_{fus} = L_f/D_f = 7$

$k_{fus} = 1.18$

$f_{fus} = k_{fus} \cdot C_{ffus} S_{wetfus} = 15.6$

Tail

$f_t = .38\, f_w = 7.25$

Nacelles

$Re_n = Re_L \cdot \bar{c} = 2.92e7$

$S_{wetn} = 2.1 \cdot \sqrt{T_{engine}} \cdot N_{engine} = 1.4e3$

$C_{fn} = .0027$

$k_n = 1.25$

$f_n = k_n \cdot C_{fn} \cdot S_{wetn} = 4.9$

Pylons

$f_p = .2 \cdot f_n = .98$

$f = f_w + f_{fus} + f_t + f_n + f_p = 47.7$

$C_{D0} = f/S = .016$

$e = 1/(1.015 + 0.38 C_{D0} \pi AR) = .85$

Climb:

$W_{avg} = (1 + .965) \cdot W_{to}/2 = 6.86e5$

$h_{avg} = (20/35) \cdot h = 20000$

$\sigma_{cl} = .5702$

$V_{Lomax} = \left(\frac{12.9}{(f \cdot c)^{1/4}}\right) \cdot \sqrt{W_{avg}/(\sigma \cdot b)} = 358.5$

$V_{cl} = 1.3 \cdot V_{Lomax} = 466.1$

$M_{cl} = V_{cl}/(1036/1.688) = .759$

$T_{req\,cl} = (6f \cdot V_{cl}^2/296) + (94.1/6e)\left(\left(\frac{W_{avg}}{b}\right)^2 \cdot \left(\frac{1}{V_{cl}^2}\right)\right) = 4.06e4$

$T_{cl\,sl} = 1.715e4$

$T_{cl\,25} = 1.425e4$

$SFC_{cl} = .6427$

$T_{cl25} = 1.425e4$

$SFC_{cl} = .6427$

$SFC_{25} = .621$

$T_{cl} = 1.57e4$

$SFC_{cl} = .6318$

$T_{avbl} = 3.0125e4$

$R/C = (101(N_{eng} \cdot T_{avbl} - T_{req_{cl}})V_{cl})/W_{avg} = 3.42e3$

$Time_{cl} = h/R/C = 10.24$

$Range_{cl} = V_{cl} \cdot (Time/60) = 79.6$

$W_{fcl} = (N_{eng} \cdot T_{avbl}) \cdot SFC_{cl} \cdot (Time/60) = 9.749$

Range

$W_0 = W_{to} - W_{fcl} = 6.88e5$

$W_1 = (1 - FF_{JTO}) \cdot W_{to} = 4.259e5$

$C_{Lavg} = ((W_0 + W_1)/2)/(1481.6 \cdot M^2) = .4781$

$C_{Di} = C_{Lavg}^2/\pi Re = .0107$

$C_{Dc} = .001$

$C_D = C_{Do} + C_{Di} + C_{Dc} = .0261$

$L/D = C_{Lavg}/C_D = 18.3$

$T_{req} = ((W_0 + W_1)/2)/L/D = 3.05e4$

$T_{req_{eng}} = T_{req}/N_{eng} = 5.291e3$

$SFC = 0.6224$

$Range_{cr} = (Ma/SFC) \cdot L/D \cdot \ln(W_0/W_1) = 6.67e3$

$Range = Range_{cl} + Range_{cr} = 6.75e3$

Compare Range to Range(allout) and iterate Fuel Fraction until they match

$Range = Range_{allout}$

Top of Climb

$C_{Lictop} = (W/S)/(1481.6 \cdot M^2) = 0.5898$

$C_{Di top} = C_{Lictop}^2/\pi AR e = 0.0163$

$C_{Dtop} = C_{Do} + C_{Di top} + .001 = 0.0323$

$L/D_{top} = C_{Lictop}/C_{Dtop} = 15.05$

$T_{avtop} = W_0/L/D_{top} = 4.17e4$

$T_{reqtop} = T_{reqtop}/\#engines = 1.39e4$

L/D_{stop} : C_{Lstop}/C_{Dstop} = 15.05

$T_{req\,stop}$: $W_o/L/D_{stop}$ = 4.17e4

$T_{req\,stop\,eng}$: $T_{req\,stop}/\#engines$ = 1.39e4

$T_{req\,stop\,std}$: $(T_{sea}/T_{sea})\,T_{req\,stop\,eng}$ = 8.06e3

Climb Gradients

Segment 1

C_{L1} : $C_{Lmaxto}/1.2^2$ = 1.21

C_{L1}/C_{Lmax} : C_{L1}/C_{Lmaxto} = 0.694

C_{D1} : see fig = 0.0143

L/D_1 : C_{L1}/C_{D1} = 10.7

T_{req1} : $W_{to}/L/D_1$ = 5.94e4

T_{a1} : $(T_e/T_{sea})\cdot T_{to}$ = 6.05e4

$Grad_1$: $((\#eng-1)T_{a1}-T_{req1})\,100/W_{to}$ = 9.69

Segment 2

C_{D2} : $C_{Do}+C_{DP1}+C_{L1}^2/\pi ARe$ = 0.098

L/D_2 : C_{L1}/C_{D2} = 12.36

T_{req2} : $W_{to}/L/D_2$ = 5.15e4

$Grad_2$: $((\#eng-1)T_{a2}-T_{req2})\,100/W_{to}$ = 10.9

Segment 3

$C_{Lmaxclean}$: see Fig = 1.09

V_3 : $1.2\sqrt{2\rho\cdot W/S_{to}/.925C_{Lmaxclean}}$ = 244.14

M_3 : $V_3/659$ = 0.371

C_{L3} : $C_{Lmaxclean}/1.2^2$ = 0.755

C_{D3} : $C_{Do}+C_{L3}^2/\pi ARe$ = 0.0418

L/D_3 : C_{L3}/C_{D3} = 18.08

T_{req3} : $W_{to}/L/D_3$ = 3.52e4

T_{a3} : $(T_e/T_{sea})\cdot T_{climb}$ = 4.63e4

$Grad_3$: $((\#eng-1)T_{a3}-T_{req3})\,100/W_{to}$ = 9.04

Approach

C_{LA} : $C_{Lmaxto}/1.2^2$ = 1.0288

$C_{LA}/C_{Lmaxclean}$: C_{LA}/C_{Lmaxto} = 0.5917

C_{DpA} : see Fig = 0.011

C_{DA} : $C_{Do}+C_{DpA}+C_{LA}^2/\pi ARe$ = 0.0756

L/D_A : C_{LA}/C_{DA} = 13.6

$W_{landing}$: $W/S_{landing}\cdot S$ = 4.507e5

T_{reqA} : $W_{landing}/L/D_A$ = 3.313e4

V_A : $\sqrt{2\rho\,W/S_{landing}/.953C_{LA}}$ = 173.45

M_A : $V_A/659$ = 0.263

T_{aA} : $(T_e/T_{sea})\cdot T_{climb}$ = 5.061e4

$Grad_A$: $((\#eng-1)T_{aA}-T_{reqA})\,100/W_{to}$ = 15.1076

Landing

C_{Ll} : $C_{Lmaxlanding}^2/1.3^2$ = 1.58

C_{Dpl} : see fig = 0.0199

C_{Dl} : $C_{Do}+C_{Dpl}+C_{Dgear}+C_{Ll}^2/\pi ARe$ = 0.1669

L/D_L : C_{Ll}/C_{Dl} = 9.46

T_{reql} : $W_l/L/D_L$ = 4.76e4

V_l : $\sqrt{2\rho\,W/S_l/.953C_{Ll}}$ = 140

M_l : $V_l/659$ = 0.212

T_{al} : $(T_e/T_{sea})\cdot T_l$ = 6.39e4

$Grad_l$: $((\#eng\cdot T_{al})-T_{reql})\,100/W_{to}$ = 31.94

DOC

D : $R_{related}\cdot 1.15$ = 6.9e3

T_{gr} : $(D/(119+.04D))/60$ = 0.393

T_{ctr} : $T_{ct}/60$ = 0.18

$D = R_{adad} \cdot 1.15 = 6.9e3$

$T_{gn} = \left(D / (11.9 + .04D)\right) / 60 = 0.393$

$T_{ctbr} = T_{ct} / 60 = 0.18$

$T_d = 0$

$T_{an} = .1$

$k_a = .02D = 138$

$D_d = \left(D + k_a + 20\right) - \left(R_{ad} \cdot 1.15 + D_d\right) / V_{cr} \cdot 1.15 = 12.8$

$V_B = D / 3T = 511.8$

$F_{cran} = T_{rwgr} \cdot SFC_{cr} \cdot \left(T_{cr} + T_{an}\right) = 2.29e5$

$F_{cL} = W_{fcL} = 9.18e3$

$F_b = F_{cran} + F_{cL} = 2.38e5$

$P = W_{pl} / 2000 = 35.6$

$C_{ptbr} = 17.84 \cdot \left(\frac{V_{cr} \cdot W_{to}}{100000}\right)^{.3} + 40.83 = 238.1$

$C_{TMfc} = C_{ptbr} / V_B \cdot P = 0.0131$

$C_{ft} = .0438$

$C_{ot} = 2.15$

$C_{TMfo} = \left(1.02 \cdot F_b \cdot C_{ft} + \#_{eng} C_{ot} \cdot T_b \cdot .135\right) / D \cdot P = 0.0433$

$W_a = W_{to}\left(1 - FF\right) - W_{pl} - \left(W_{ff} \cdot W_{to}\right) = 2.52e5$

$C_a = 2.4e6 + \left(87.5 \cdot W_a\right) = 2.45e7$

$C_e = 590000 + \left(16 T_e\right) = 1.31e6$

$C_T = \left(\#_{eng} \cdot C_e\right) + C_a = 2.84e7$

$IR_a = .01$

$U = 630 + 4000 / \left(11 / (T_b + 35)\right) = 4.36e3$

$C_{TMIR} = IR_A \cdot C_T / \left(U \cdot V_B \cdot P\right) = 0.0036$

$K_{FHa} = 4.9169 \cdot \log\left(W_a / 1000\right) - 6.425 = 5.38$

$K_{FCa} = .213 \cdot \log\left(W_a / 1000\right)^{3.74} = 5.62$

$T_F = T_b - T_{gn} = 13.1$

$R_L = 8.6$

$C_{TMa} = \left(K_{FHa} \cdot T_F + K_{FCa}\right) \cdot R_L \cdot \left(1 + .29\left(M - 1\right)^{1.5}\right) / \left(V_B T_b P\right) = 0.0025$

$C_{FHa} = \frac{1.5 C_a}{1000000} + 3.4 = 42.5$

$C_{FCa} = \frac{1.923 C_a}{1000000} + 2.25 = 49.3$

$C_{TMb} = \left(C_{FHa} \cdot T_F + C_{FCa}\right) / \left(V_B T_b P\right) = 0.0025$

$K_{FHe} = \left(\#_{eng} \cdot T_e / 1000\right) / \left(.827 \cdot \left(T_e / 1000\right) + 13.639\right) = 3$

$K_{FCe} = .2 \cdot \#_{eng} = 0.6$

$C_{TMe} = \left(K_{FHe} \cdot T_F + K_{FCe}\right) / \left(V_B T_b P\right) = 0.0014$

$C_{FHa} = \left(\left(28.2 C_e / 1000000\right) - 6.5\right) \cdot \#_{eng} = 91.8$

$C_{FCa} = \left(\left(3.67 C_e / 1000000\right) + 1.36\right) \cdot \#_{eng} = 18.6$

$C_{TMd} = \left(C_{FHe} \cdot T_F + C_{FCe}\right) / \left(V_B T_b P\right) = 0.005$

$C_{TMc} = 2\left(C_{TMa} + C_{TMb} + C_{TMe} + C_{TMd}\right) = 0.0226$

$D_a = 14$

$C_{TMdep} = \left(\frac{1}{V_B P}\right) \cdot \left(C_T + .06\left(C_T - \#_{eng} C_e\right) + .3 \#_{eng} C_e\right) / \left(D_a U\right) = 0.0279$

$DOC_{mi} = C_{TMfc} + C_{TMfo} + C_{TMIR} + C_{TMc} + C_{TMdep} = 0.111$

$DOC_{mx} = DOC_{mi} \cdot \frac{P}{PAX} = 0.0143$

Code for 3D Plots and
Overall Optimizations

```
%% Start
close all
clear
clc

%% Prompts

prompt_0 = "Is this unique (1), sample (2), or quiz (3) configuration? ";
type_inputs = input(prompt_0);

prompt_1 = "Would you like to draw graphs and fit curves? 1 for no, 2 for yes: ";
type_graph = input(prompt_1);

prompt_2 = "Please input 1 for conventional wing, 2 for supercritical: ";
type_airfoil = input(prompt_2);

prompt_3 = "Please enter what type of engine you are using, 1 for JT9D Advanced, 2 for JT9D, and 3 for JT8D↙
";
type_engine = input(prompt_3);

prompt_4 = "Please enter how many engines the aircraft has (2, 3, or 4): ";
type_engineqty = input(prompt_4);

if type_engineqty == 1
    type_engineqty = 2;
end

prompt_5 = "Are the engines mounted on the wing (1) or fuselage (2)? ";
type_mount = input(prompt_5);

prompt_6 = "Is the structure conventional (1), hybrid (2), or composites (3)? ";
type_structure = input(prompt_6);

prompt_7 = "Is this flight domestic (1) or international (2)? ";
type_flight = input(prompt_7);

if type_mount == 1
    K_w = 1;
    K_ts = 0.17;
elseif type_mount ==2
    K_w = 1.03;
    K_ts = 0.25;
end

%% Fixed Variables

l = 5; % number of lines in graphs at the end

range_AR = [8:.5:14];
range_Sweep = [20:1:38];

if type_inputs == 1 %Unique
    Taper = 0.35;

    height = 35000; %ft
    M = 0.82;
    Range_scheduled = 4000;
    FF_max = 1;
    V_approach = 135;

    TOFL = 6000;
```

```
    PAX = 200; %passengers
    Abreast = 6; %number of passengers abreast
    Aisles = 2;
    Weight_cargo = 4000; %weight of cargo
    Number_flightcrew = 2; %number of flight crew
    Number_stewards = round((PAX/50)+0.49); %number of stewards
elseif type_inputs == 2 %Sample
    Taper = 0.35;

    height = 35000; %ft
    M = 0.82;
    Range_scheduled = 6000;
    FF_max = .75;
    V_approach = 140;
    TOFL = 9000;

    PAX = 275; %passengers
    Abreast = 8; %number of passengers abreast
    Aisles = 2;
    Weight_cargo = 12000; %weight of cargo
    Number_flightcrew = 2; %number of flight crew
    Number_stewards = round((PAX/50)+0.49); %number of stewards

    type_engineqty = 3;
elseif type_inputs ==3 %Quiz
    Taper = 0.35;

    height = 35000; %ft
    M = 0.82;
    Range_scheduled = 4000;
    FF_max = 1;
    V_approach = 135;
    TOFL = 6000;

    PAX = 200; %passengers
    Abreast = 6; %number of passengers abreast
    Aisles = 2;
    Weight_cargo = 4000; %weight of cargo
    Number_flightcrew = 2; %number of flight crew
    Number_stewards = round((PAX/50)+0.49); %number of stewards
end

gamma = 1.4;
R = 1718;
T = 218.9; %K
rho = .023751; %lb/ft^3
sigma = .953; %Ratio between local and sea level densities
delta = .2360; %Ratio between local and sea level pressures

%a = sqrt(gamma*R*T);
a = 576.4;

%% Digitized Plots

Table = readtable('Tables.xlsx');
Table = Table{:,:};

if type_airfoil == 1
% Figure 1a: Effect of Wing Sweep and Thickness on Mdiv (Conventional)
    x1 = Table(1:165,1);
    y1 = Table(1:165,3);
    z1 = Table(1:165,2);
```

```
    [fit_TC,info_TC] = fit([x1,y1],z1,'poly23');

% Figure 1b: Effect of Wing Sweep and Thickness on Mdiv (Supercritical)
elseif type_airfoil == 2
    x2 = Table(1:211,4);
    y2 = Table(1:211,6);
    z2 = Table(1:211,5);
    [fit_TC,info_TC] = fit([x2,y2],z2,'poly35');
end
% Figure 2: Wing Drag Divergence Change Due to Lift
if type_airfoil == 1
    x3 = Table(1:25,7);
    y3 = Table(1:25,8);
    [fit_MDIVdelta,info_MDIVdelta] = fit(y3,x3,'power2');
elseif type_airfoil == 2
    x4 = Table(1:29,9);
    y4 = Table(1:29,10);
    [fit_MDIVdelta,info_MDIVdelta] = fit(y4,x4,'power2');
end

% Figure 3: CLmax Conceptual Design Estimates
x5 = Table(1:34,11);
y5 = Table(1:34,12);
[fit_CLmax_takeoff,info_CLmax_takeoff] = fit(x5,y5,'smoothingspline');
x6 = Table(1:31,13);
y6 = Table(1:31,14);
[fit_CLmax_landing,info_CLmax_landing] = fit(x6,y6,'smoothingspline');

% Figure 4: JT8D-9 Fuel Fraction Estimate for All Out Range
x7 = Table(1:41,15);
y7 = Table(1:41,16);
[fit_FF,info_FF] = fit(x7,y7,'cubicinterp'); % FF is Fuel Fraction or W_f/W_to

% Figure 5: Jet Aircraft TOFL with Engine Failure
x8 = Table(1:58,17);
y8 = Table(1:58,18);
[fit_k_2ENG,info_k_2ENG] = fit(y8,x8,'poly2');
x9 = Table(1:63,19);
y9 = Table(1:63,20);
[fit_k_3ENG,info_k_3ENG] = fit(y9,x9,'poly2');
x10 = Table(1:67,21);
y10 = Table(1:67,22);
[fit_k_4ENG,info_k_4ENG] = fit(y10,x10,'poly2');

% Cf
x11 = Table(1:50,23);
y11 = Table(1:50,24);
y11 = y11.*(1e-3);
[fit_Cf, info_Cf] = fit(x11,y11,'power2');

% Body Form Factor
x12 = Table(1:12,25);
y12 = Table(1:12,26);
[fit_k, info_k] = fit(x12,y12,'exp2');

% Max Climb Thrust @ 15000ft
x13 = Table(1:34,27);
y13 = Table(1:34,28);
[fit_T_avbl15, info_T_avbl15] = fit(x13,y13,'smoothingspline');
x14 = Table(1:9,29);
y14 = Table(1:9,30);
[fit_SFC_15, info_SFC_15] = fit(x14,y14,'smoothingspline');
```

```
% Max Climb Thrust @ 25000ft
x15 = Table(1:30,31);
y15 = Table(1:30,32);
[fit_T_avbl25, info_T_avbl25] = fit(x15,y15,'smoothingspline');
x = Table(1:6,33);
y = Table(1:6,34);
[fit_SFC_25, info_SFC_25] = fit(x,y,'smoothingspline');

% Max Climb and Thrust @ 35000ft
x16 = Table(1:9,37);
y16 = Table(1:9,38);
[fit_SFC_35, info_SFC_35] = fit(x16,y16,'lin');

% Profile Drag at Takeoff
x17 = Table(1:60,39);
y17 = Table(1:60,40);
[fit_CDp_takeoff, info_CDp_takeoff] = fit(y17,x17,'poly5');

% Profile Drag at Landing
x18 = Table(1:60,41);
y18 = Table(1:60,42);
[fit_CDp_landing, info_CDp_landing] = fit(y18,x18,'poly5');

% Clean Wing CLmax
x19 = Table(1:69,43);
y19 = Table(1:69,45);
z19 = Table(1:69,44);
[fit_CLmax_clean, info_CLmax_clean] = fit([x19,y19],z19,'poly32');

% JT9D Dry Takeoff
x20 = Table(1:18,46);
y20 = Table(1:18,47);
[fit_JT9D_takeoffthrust, info_JT9D_takeoffthrust] = fit(x20,y20,'lin');

% JT9D Climb
x21 = Table(1:26,48);
y21 = Table(1:26,49);
[fit_JT9D_climbthrust, info_JT9D_climbthrust] = fit(x21,y21,'lin');

%% Figures

if type_graph == 2

    figure
    plot(fit_TC_conv,[x1,y1],z1);

    figure
    plot(fit_TC_sc,[x2,y2],z2);
    hold on
    ylim([0 40]);
    zlim([0.07 .2]);
    hold off

    figure
    hold on
    plot(x3,y3)
    plot(x4,y4)
    xlim([-.04 .02])
    ylim([.4 .7])
    % plot(fit_MDIVdelta_conv)
    % plot(fit_MDIVdelta_sc)
```

```
    % xlim([-1 2])
    hold off

    figure
    hold on
    plot(x5,y5);
    plot(x6,y6);
    hold off

    figure
    plot(fit_FF);

    figure
    hold on
    plot(x8,y8,x9,y9,x10,y10)
    hold off

    figure
    plot(fit_Cf,x11,y11)
    hold on
    set(gca,'xscale','log')
    title('Skin Friction Coefficient wrt Reynolds Number')
    xlabel('Reynolds Number (Re)')
    ylabel('Coefficient of Skin Friction (k)')
    hold off

    figure
    plot(fit_k,x12,y12);
    hold on
    title('Body Form Factor wrt Fineness Ratio')
    xlabel('Fineness Ratio (l/d)')
    ylabel('Body Form Factor (k)')
    hold off

    figure
    plot(x17,y17,x18,y18)

    figure
    hold on
%   plot([z19,x19],y19)
    plot(fit_CLmax_clean)
    xlim([.04 .16]);
    ylim([0 35]);
    zlim([.8 1.3]);
    hold off
end

%% Placeholders

iteration = 1;
table3d_AR = zeros(1,(length(range_AR)*length(range_Sweep)));
table3d_Sweep = zeros(1,(length(range_AR)*length(range_Sweep)));
table3d_DOC = zeros(1,(length(range_AR)*length(range_Sweep)));
table3d_Weight = zeros(1,(length(range_AR)*length(range_Sweep)));
table_CruiseSFC = zeros(1,(length(range_AR)*length(range_Sweep)));
table_Vcruise = zeros(1,(length(range_AR)*length(range_Sweep)));
table_LDcruise = zeros(1,(length(range_AR)*length(range_Sweep)));
table_FF = zeros(1,(length(range_AR)*length(range_Sweep)));
table_Wfuel = zeros(1,(length(range_AR)*length(range_Sweep)));
table_Payload = zeros(1,(length(range_AR)*length(range_Sweep)));
table_Thrust = zeros(1,(length(range_AR)*length(range_Sweep)));
table_S = zeros(1,(length(range_AR)*length(range_Sweep)));
```

```
table_AR = zeros(length(range_Sweep),length(range_AR));
table_Sweep = zeros(length(range_Sweep),length(range_AR));
table_DOC = zeros(length(range_Sweep),length(range_AR));
table_Weight = zeros(length(range_Sweep),length(range_AR));

SFC_jt9d = 0.61;
if type_engine == 1
    SFC_jt9d = 0.9 * SFC_jt9d;
end

%% Configuration Loop

i = 1;
for Sweep = range_Sweep
    j = 1;
    for AR = range_AR
        Weightincrement = 0;
        Thrust_reqtop_JT9D = 10001;
        while Thrust_reqtop_JT9D > 10000
%% Range Loop
            Range = 0;
            Range_ao = 500000;
            Fuelincrement = 0;
            while abs(Range - Range_ao) > 2
%% CL Loop
                CL_guess = 0.5;
                CL_ic = 0;
                while abs(CL_ic - CL_guess) > 0.001
                    Mdiv_delta = fit_MDIVdelta(CL_guess);
                    Mdiv = (M + .004) - Mdiv_delta;
                    TC = fit_TC(Mdiv,Sweep);
                    CLmax_landing = fit_CLmax_landing((((cosd(Sweep))^2)*((TC)^2)*AR));
                    CLmax_takeoff = fit_CLmax_takeoff((((cosd(Sweep))^2)*((TC)^2)*AR));%5
                    WS_landing = ((V_approach/1.3)^2) * (sigma * CLmax_landing) / 296;%6
                    V_cruise = M * a;
                    Range_ao = Range_scheduled + 200 + (V_cruise*(3/4));%7
                    FF_jt8d = fit_FF(Range_ao); %8
                    if type_inputs == 1
                        FF_jt9d = (FF_jt8d * (SFC_jt9d/.78)); %9
                    elseif type_inputs == 2
                        FF_jt9d = (FF_jt8d * (SFC_jt9d/.78)); %9
                    elseif type_inputs == 3
                        FF_jt9d = (FF_jt8d * (SFC_jt9d/.78)); %9
                    end
                    if type_engine == 1 || type_engine == 2
                        FF = FF_jt9d + Fuelincrement;
                    elseif type_engine == 3
                        FF = FF_jt8d + Fuelincrement;
                    end
                    WS_takeoff = WS_landing/(1-(FF_max*FF));
                    WS_ic = .965 * WS_takeoff;
                    CL_ic = WS_ic / (1481 * delta * M^2);
                    if abs(CL_ic - CL_guess) > 0.001
                        CL_guess = (CL_guess + CL_ic)/2;
                    end
                end
%% TOFL
                if type_engineqty == 2
                    k_1 = fit_k_2ENG((TOFL/1000));
                elseif type_engineqty == 3
                    k_1 = fit_k_3ENG((TOFL/1000));
```

```
                elseif type_engineqty == 4
                    k_1 = fit_k_4ENG((TOFL/1000));
                end
                WT_7V_lo = (k_1/WS_takeoff) * sigma * CLmax_takeoff;
                V_lo = 1.2 *sqrt((296*WS_takeoff)/(sigma*CLmax_takeoff));
                M_lo = V_lo/(661/sqrt(sigma));
                Thrust_sls = fit_JT9D_takeoffthrust(0);
                Thrust_M = fit_JT9D_takeoffthrust(M_lo*0.7);
                WT = (WT_7V_lo * Thrust_M/Thrust_sls) + Weightincrement;
%% Weight
                eta = 1.5*2.5;
                % Wing
                TC_bar = TC + 0.03;
                W_w = (.00945 * (AR^0.8) * ((1+Taper)^0.25) * K_w * (eta^0.5))/((TC_bar^0.4) * cosd(Sweep) ↵
(WS_takeoff^0.695)); %times W_takeoff^1.195
                % Fuselage
                K_f = 11.5;
                L_f = (3.76 * (PAX/Abreast) + 33.2);
                D_f = ((1.75*Abreast) + (1.58*Aisles) + 1);
                if type_flight == 2
                    L_f = 1.1 * L_f;
                    D_f = 1.1 * D_f;
                end
                W_fus = 0.6727 * K_f * (L_f^0.6) * (D_f^0.72) * (eta^0.3);%times W_takeoff^0.235
                % Landing Gear
                W_lg = 0.04; %times W_takeoff
                % Nacelle & Pylon
                W_np = (0.0555/WT); %times W_takeoff
                % Tail Surface
                W_ts = K_ts * W_w; %times W_takeoff^1.195
                % Power Plant
                W_pp = 1/(3.58*WT); %times W_takeoff
                if type_engine == 1
                    W_pp = W_pp * 1.1;
                end
                % Fuel
                W_f = 1.0275 * (FF);
                % Payload
                W_pl = 215*PAX + Weight_cargo;
                % Fixed Equipment
                W_fe = (132*PAX) + (300*type_engineqty) + (260*Number_flightcrew) + (170*Number_stewards);
                W_fe_wto = (0.035); %times W_takeoff
                if type_structure == 2
                    W_w = 0.7 * W_w;
                    W_ts = 0.7 * W_ts;
                    W_np = 0.8 * W_np;
                elseif type_structure == 3
                    W_w = 0.7 * W_w;
                    W_ts = 0.7 * W_ts;
                    W_fus = 0.85 * W_fus;
                    W_fe = 0.9 * W_fe;
                    W_fe_wto = 0.9 * W_fe_wto;
                    W_np = 0.8 * W_np;
                end
                a_W = W_w + W_ts;
                b_W = W_fus;
                c_W = W_lg + W_np + W_pp + W_f + W_fe_wto - 1;
                d_W = W_pl + W_fe;
                W_takeoff = 300000;
                difference_1 = W_takeoff;
                while abs(difference_1) > 0.1
                    if abs(difference_1) > 0.5
```

```
                    difference_1 = a_W*(W_takeoff^1.195) + b_W*(W_takeoff^0.235) + c_W*W_takeoff + d_W;
                    W_takeoff = W_takeoff + 4*(difference_1);
                elseif abs(difference_1) <= 0.5
                    difference_1 = a_W*(W_takeoff^1.195) + b_W*(W_takeoff^0.235) + c_W*W_takeoff + d_W;
                    W_takeoff = W_takeoff + (difference_1);
                end
            end
            W_fuel = W_f * W_takeoff;
%% Wing Sizing
            S = W_takeoff/WS_takeoff;
            b = sqrt(AR*S);
            c_bar = S/b;
            Thrust_total = W_takeoff/WT;
            Thrust_engine = Thrust_total/type_engineqty;
%% Drag
            Re_l = 2.852e6 * 0.5; % 0.5 * rho * v / mu
            % Wing
            Re_w = Re_l * c_bar;
            Cf_w = fit_Cf(Re_w);
            Swet_w = 2 * S * 1.02;
            z_w = ((2-M^2)*cosd(Sweep))/sqrt(1-((M^2).*(cosd(Sweep))));
            k_w = 1+(z_w*TC)+100*(TC^4);
            f_w = k_w*Cf_w*Swet_w;
            % Fuselage
            Swet_fus = 0.9*pi*D_f*L_f;
            Re_fus = Re_l * L_f;
            Cf_fus = fit_Cf(Re_fus);
            fineness_fus = L_f/D_f;
            k_fus = fit_k(fineness_fus);
            f_fus = k_fus*Cf_fus*Swet_fus;
            % Tail
            f_t = 0.35*f_w;
            % Nacelles
            Re_n = Re_l * c_bar;
            Swet_n = 2.1 * sqrt(Thrust_engine) * type_engineqty;
            Cf_n = fit_Cf(Re_n);
            k_n = 1.25;
            f_n = k_n*Cf_n*Swet_n;
            % Pylons
            f_p = 0.2*f_n;
            %Total
            f = f_w + f_fus + f_t + f_n + f_p;
            CD_overall = f/S;
            e = 1/(1.035 + (.38*CD_overall*pi*AR));
            CD_gear = CD_overall;
%% Climb
            W_avgcl = (1.965) * W_takeoff / 2;
            h_avgcl = (20/35) * height;
            sigma_cl = .5702;
            V_ldmax = (12.9/(f*e)^(1/4)) * sqrt(W_avgcl/(sigma_cl*b));
            V_cl = 1.3 * V_ldmax;
            M_cl = V_cl/(973.1/1.68781);
            Thrust_reqcl = (sigma_cl*f*(V_cl^2)/296) + ((94.1/(sigma_cl*e))*((W_avgcl/b)^2)*(1/
(V_cl^2)));
            Thrust_cl15 = fit_T_avbl15(M_cl) * 1000;
            Thrust_cl25 = fit_T_avbl25(M_cl) * 1000;
            SFC_15 = fit_SFC_15(M_cl);
            SFC_25 = fit_SFC_25(M_cl);
            Thrust_cl = (Thrust_cl15 + Thrust_cl25)/2;
            SFC_cl = (SFC_15 + SFC_25)/2;
            if type_engine == 1
                SFC_cl = SFC_cl * 0.9;
```

```
                end
                Thrust_avbl = (Thrust_engine/Thrust_sls) * Thrust_cl;
                RC = (101*((type_engineqty*Thrust_avbl)-Thrust_reqcl)*V_cl)/W_avgcl;
                Time_cl = height/RC;
                Range_cl = V_cl * (Time_cl/60);
                W_fcl = (type_engineqty*Thrust_avbl) * SFC_cl * (Time_cl/60);
%% Range
                W_0 = W_takeoff-W_fcl;
                W_I = (1-FF) * W_takeoff;
                CL_avg = ((W_0+W_I)/(2*S)) / (1481*delta*(M^2));
                CD_i = (CL_avg^2)/(pi*AR*e);
                CD_c = 0.001;
                CD = CD_overall + CD_i + CD_c;
                LD = CL_avg/CD;
                Thrust_reqcr = ((W_0 + W_I)/(2*LD));
                Thrust_reqcrjt9d = Thrust_reqcr * (Thrust_sls/Thrust_engine);
                Thrust_reqcreng = Thrust_reqcrjt9d / type_engineqty;
                SFC_cr = fit_SFC_35(Thrust_reqcreng);
                if type_engine == 1
                    SFC_cr = SFC_cr * 0.9;
                end
                Range_cr = (V_cruise/SFC_cr) * LD * log(W_0/W_I);
                Range = Range_cl + Range_cr;
                difference_2 = Range-Range_ao;
                if     difference_2 <  -200 %200 or less
                    Fuelincrement = Fuelincrement + 0.01;
                elseif difference_2 >= 200 %200 or more
                    Fuelincrement = Fuelincrement - 0.01;
                elseif difference_2 <  -18 && (Range-Range_ao) >= -200 %Between -20 and -200
                    Fuelincrement = Fuelincrement + 0.001;
                elseif difference_2 >  18  && (Range-Range_ao) <= 200 %Between 20 and 200
                    Fuelincrement = Fuelincrement - 0.001;
                elseif difference_2 <  -3    && (Range-Range_ao) >= -18 %Between 0 and -20
                    Fuelincrement = Fuelincrement + 0.0001;
                elseif difference_2 >= 3    && (Range-Range_ao) <= 18 %Between 0 and 20
                    Fuelincrement = Fuelincrement - 0.0001;
                elseif difference_2 <  0    && (Range-Range_ao) >= -3 %Between 0 and -20
                    Fuelincrement = Fuelincrement + 0.00001;
                elseif difference_2 >= 0    && (Range-Range_ao) <= 3 %Between 0 and 20
                    Fuelincrement = Fuelincrement - 0.00001;
                end
            end
%% Top of Climb
            CL_ic_top = (W_0/S)/(1481*delta*M^2);
            CDi_top = (CL_ic_top^2) / (pi*AR*e);
            CD_top = CD_overall + CDi_top + 0.001;
            LD_top = CL_ic_top / CD_top;
            if type_inputs == 2
                LD_top = 15.05; % SAMPLE
            end
            Thrust_reqtop = W_0 / LD_top;
            Thrust_reqtop_engine = Thrust_reqtop / type_engineqty;
            Thrust_reqtop_JT9D = (Thrust_sls/Thrust_engine) * Thrust_reqtop_engine;

%% Climb Gradients
            % 1st Segment
            CL_seg1 = CLmax_takeoff/(1.2^2);
            CL_to_CLmax_to = CL_seg1/CLmax_takeoff;
            CDp_seg1 = fit_CDp_takeoff(CL_to_CLmax_to);
            CD_seg1 = CD_overall + CDp_seg1 + CD_gear + ((CL_seg1^2)/(pi*AR*e));
            LD_seg1 = CL_seg1/CD_seg1;
            Thrust_reqseg1 = W_takeoff/LD_seg1;
```

```
            Thrust_avblseg1 = (Thrust_engine/Thrust_sls) * fit_JT9D_takeoffthrust(M_lo);
            Grad_1 = ((type_engineqty-1)*Thrust_avblseg1-Thrust_reqseg1)*100/W_takeoff;
            % 2nd Segment
            CD_seg2 = CD_overall + CDp_seg1 + ((CL_seg1^2)/(pi*AR*e));
            LD_seg2 = CL_seg1/CD_seg2;
            Thrust_reqseg2 = W_takeoff/LD_seg2;
            Grad_2 = ((type_engineqty-1)*Thrust_avblseg1-Thrust_reqseg2)*100/W_takeoff;
            % 3rd Segment
            CLmax_clean = fit_CLmax_clean(TC,Sweep);
            V_seg3 = 1.2*sqrt((296*WS_takeoff)/(.925*CLmax_clean));
            M_seg3 = V_seg3/659;
            CL_seg3 = CLmax_clean/(1.2^2);
            CD_seg3 = CD_overall + ((CL_seg3^2)/(pi*AR*e));
            LD_seg3 = CL_seg3/CD_seg3;
            Thrust_reqseg3 = W_takeoff/LD_seg3;
            Thrust_avblseg3 = (Thrust_engine/Thrust_sls) * fit_JT9D_climbthrust(M_seg3);
            Grad_3 = ((type_engineqty-1)*Thrust_avblseg3-Thrust_reqseg3)*100/W_takeoff;
            % Approach
            CL_approach = CLmax_takeoff/(1.3^2);
            CL_approach_CLmax_landing = CL_approach/CLmax_takeoff;
            CDp_approach = fit_CDp_takeoff(CL_approach_CLmax_landing);
            CD_approach = CD_overall + CDp_approach + (CL_approach^2)/(pi*AR*e);
            LD_approach = CL_approach/CD_approach;
            W_landing = WS_landing * S;
            Thrust_reqapproach = W_landing/LD_approach;
            V_approach_grad = sqrt((296*WS_landing)/(.953*CL_approach));
            M_approach = V_approach_grad/659;
            Thrust_avblapproach = (Thrust_engine/Thrust_sls) * fit_JT9D_climbthrust(M_approach);
            Grad_approach = ((type_engineqty-1)*Thrust_avblapproach-Thrust_reqapproach)*100/W_landing;
            %Landing
            CL_landing = CLmax_landing/1.3^2;
            CDp_landing = fit_CDp_landing(1/1.3^2);
            CD_landing = CD_overall + CDp_landing + CD_gear + (CL_landing^2)/(pi*AR*e);
            LD_landing = CL_landing/CD_landing;
            Thrust_reqlanding = W_landing/LD_landing;
            V_landing = sqrt((296*WS_landing)/(0.953*CL_landing));
            M_landing = V_landing/659;
            Thrust_avbllanding = (Thrust_engine/Thrust_sls) * fit_JT9D_takeoffthrust(M_landing);
            Grad_landing = ((type_engineqty*Thrust_avbllanding) - Thrust_reqlanding) * 100 / W_landing;
            % Climb Requirements (pg. 39)
            if type_engineqty == 2
                Grad_1_min = 0;
                Grad_2_min = 2.4;
                Grad_3_min = 1.2;
                Grad_approach_min = 2.1;
                Grad_landing_min = 3.2;
            elseif type_engineqty == 3
                Grad_1_min = 0.3;
                Grad_2_min = 2.7;
                Grad_3_min = 1.5;
                Grad_approach_min = 2.4;
                Grad_landing_min = 3.2;
            elseif type_engineqty == 4
                Grad_1_min = 0.5;
                Grad_2_min = 3.0;
                Grad_3_min = 1.7;
                Grad_approach_min = 2.7;
                Grad_landing_min = 3.2;
            end

            if Thrust_reqtop_JT9D > 10000 || Grad_1 < Grad_1_min || Grad_2 < Grad_2_min || Grad_3 <
Grad_3_min || Grad_approach < Grad_approach_min || Grad_landing < Grad_landing_min
```

```
                Weightincrement = Weightincrement - 0.1;
                disp('Gradient Failed')
            end
        end

%% DOC
        D = Range_scheduled * 1.15;
        Time_gm = (D / (11.866+0.040669*D))/60;
        Time_cl_hr = Time_cl/60;
        Time_d = 0;
        Time_am = 0.10;
        if D <= 1400
            k_a = (7+0.015*D);
        elseif D > 1400
            k_a = 0.02*D;
        end
        D_d = 0;
        Time_cr = ((D+k_a+20)-(Range_cl*1.15+D_d))/(V_cruise*1.15);
        V_B = D / (Time_gm + Time_cl_hr + Time_d + Time_cr + Time_am);
        % Block Time
        Time_b = Time_gm + Time_cl_hr + Time_d + Time_cr + Time_am;
        % Block Fuel
        F_cr_am = Thrust_reqcr * SFC_cr * (Time_cr+Time_am);
        F_cl = W_fcl;
        F_b = F_cr_am + F_cl;
        % Flying Operations Cost
        % a. Flight Crew
        P = W_pl / 2000;
        if Number_flightcrew == 2
            CostPerBlockHr = 17.849 * (V_cruise*W_takeoff/1e5)^0.3 + 40.83;
        else
            CostPerBlockHr = 24.261 * (V_cruise*W_takeoff/1e5)^0.3 + 57.62;
        end
        C_TM_FlightCrew = CostPerBlockHr / (V_B * P);
        % b. Fuel and Oil
        if type_flight == 1
            C_ft = 0.0438;
        elseif type_flight == 2
            C_ft = 0.0625;
        end
        C_ot = 2.15;
        C_TM_FuelOil = ( 1.02*F_b*C_ft + type_engineqty*C_ot*Time_b*0.135 ) / (D*P);
        % c. Hull Insurance Costs
        W_a = W_takeoff * (1-FF) - W_pl - (W_pp*W_takeoff);
        C_a = 2.4e6 + (87.5*W_a);
        C_e = 590000 + (16*Thrust_engine);
        if type_inputs == 2
            C_e = 1314942;
        end
        C_T = (type_engineqty*C_e) + C_a;
        IR_A = 0.01;
        U = 630 + 4000/( 1 + 1/(Time_b + 0.5)); % block hrs/yr
        C_TM_HullInsurance = IR_A * C_T / (U * V_B * P);
        % Direct Maintenance
        % a. Airframe-Labor
        K_FHa = 4.9169 * log10(W_a/1000) - 6.425;
        K_FCa = 0.21256 * log10(W_a/1000)^3.7375;
        T_F = Time_b - Time_gm;
        R_L = 8.6;
        C_TM_a_AirframeLabor = (K_FHa * T_F + K_FCa) * R_L * (1 + 0.29*(M-1))^1.5 / (V_B * Time_b * P);
        % b. Airframe Material
        C_FHa = ((1.5994*C_a)/1e6) + 3.4263;
```

```
        C_FCa = ((1.9229*C_a)/1e6) + 2.2504;
        C_TM_b_AirframeMaterial = (C_FHa*T_F+C_FCa) / (V_B*Time_b*P);
        % c. Engine Labor
        K_FHe = (type_engineqty*(Thrust_engine/1e3))/(0.82715*(Thrust_engine/1e3)+ 13.639);
        K_FCe = 0.2 * type_engineqty;
        C_TM_c_EngineLabor = ((K_FHe*T_F+K_FCe)*R_L) / (V_B*Time_b*P);
        % d. Engine Material
        C_FHe = (((28.2353*C_e)/1e6) - 6.5176) * type_engineqty;
        C_FCe = (((3.6698*C_e)/1e6) + 1.3685) * type_engineqty;
        C_TM_d_EngineMaterial = (C_FHe*T_F+C_FCe) / (V_B*Time_b*P);
        if type_engine == 1
            C_TM_d_EngineMaterial = C_TM_d_EngineMaterial * 1.1;
        end
        % e. Total Maintenance - Burdened
        C_TM_e_Maintenance = (C_TM_a_AirframeLabor + C_TM_b_AirframeMaterial + C_TM_c_EngineLabor +
C_TM_d_EngineMaterial)*2;
        % Depreciation
        D_a = 14;
        C_TM_Depreciation = (1/(V_B*P)) * (C_T + 0.06*(C_T-type_engineqty*C_e)+0.3*type_engineqty*C_e) /
(D_a*U);
        % Total DOC
        DOC_perTonMile = C_TM_FlightCrew + C_TM_FuelOil + C_TM_HullInsurance + C_TM_e_Maintenance +
C_TM_Depreciation;
        DOC_perPAXMile = DOC_perTonMile * P/PAX;

%% Data Collection
        % Plots
        table3d_AR(iteration) = AR;
        table3d_Sweep(iteration) = Sweep;
        table3d_DOC(iteration) = DOC_perPAXMile;
        table3d_Weight(iteration) = W_takeoff;
        table_AR(i,j) = AR;
        table_Sweep(i,j) = Sweep;
        table_DOC(i,j) = DOC_perPAXMile;
        table_Weight(i,j) = W_takeoff;
        table_b(i,j) = b;

        % Range Chart
        table_CruiseSFC(iteration) = SFC_cr;
        table_Vcruise(iteration) = V_cruise;
        table_LDcruise(iteration) = LD;
        table_FF(iteration) = FF;
        table_Wfuel(iteration) = W_fuel;
        table_Payload(iteration) = W_pl;
        table_Thrust(iteration) = Thrust_engine;
        table_S(iteration) = S;

        iteration = iteration + 1
        j = j+1;
    end
    i = i+1;
end

%% Results

[fit_DOC,info_DOC] = fit([table3d_AR',table3d_Sweep'],table3d_DOC','loess');
[fit_Weight,info_Weight] = fit([table3d_AR',table3d_Sweep'],table3d_Weight','loess');
[Min_DOC,optimallocation_DOC] = min(table3d_DOC);
[Min_Weight,optimallocation_Weight] = min(table3d_Weight);
optimal_DOC = table3d_DOC(optimallocation_DOC)
optimal_AR = table3d_AR(optimallocation_DOC)
optimal_Sweep = table3d_Sweep(optimallocation_DOC)
```

```
optimal_Weight = table3d_Weight(optimallocation_DOC)
optimal_Thrust = table_Thrust(optimallocation_DOC)
optimal_FF = table_FF(optimallocation_DOC)
optimal_b = table_b(optimallocation_DOC)
optimal_S = table_S(optimallocation_DOC)
optimal_Weight_Weight = table3d_Weight(optimallocation_Weight);
optimal_Weight_AR = table3d_AR(optimallocation_Weight);
optimal_Weight_Sweep = table3d_Sweep(optimallocation_Weight);

%% Range Chart

% Values
max_Weight = table3d_Weight(optimallocation_DOC);
optimal_CruiseSFC = table_CruiseSFC(optimallocation_DOC);
optimal_Vcruise = table_Vcruise(optimallocation_DOC);
optimal_LDcruise = table_LDcruise(optimallocation_DOC);
optimal_FF = table_FF(optimallocation_DOC);
optimal_Wfuel = table_Wfuel(optimallocation_DOC);
optimal_Payload = table_Payload(optimallocation_DOC);
max_Payload = 52000;
delta_Payload = max_Payload - optimal_Payload;
% Point 1
p1_Weight0 = max_Weight;
p1_Payload = max_Payload;
p1_Range = 0;
% Point 2
p2_Weight0 = max_Weight;
p2_Payload = max_Payload;
p2_Wfuel = optimal_Wfuel - delta_Payload;
p2_Weight1 = p2_Weight0 - p2_Wfuel;
p2_Range = (optimal_Vcruise/optimal_CruiseSFC)*optimal_LDcruise*log(p2_Weight0/p2_Weight1);
% Point 3
p3_Weight0 = max_Weight;
p3_Payload = optimal_Payload;
p3_Wfuel = optimal_Wfuel;
p3_Weight1 = p3_Weight0 - p3_Wfuel;
p3_Range = (optimal_Vcruise/optimal_CruiseSFC)*optimal_LDcruise*log(p3_Weight0/p3_Weight1);
% Point 4
p4_Weight0 = max_Weight - optimal_Payload;
p4_Wfuel = optimal_Wfuel;
p4_Payload = 0;
p4_Weight1 = p4_Weight0 - p4_Wfuel;
p4_Range = (optimal_Vcruise/optimal_CruiseSFC)*optimal_LDcruise*log(p4_Weight0/p4_Weight1);
chart_Payload = [p1_Payload p2_Payload p3_Payload p4_Payload];
chart_Range = [p1_Range p2_Range p3_Range p4_Range];

%% Plots

% 3D DOC Plot
figure
scatter3(table3d_AR,table3d_Sweep,table3d_DOC);
hold on
plot(fit_DOC)
plot3(optimal_AR,optimal_Sweep,optimal_DOC,'r.','MarkerSize',30,'DisplayName','Point of Lowest DOC
Possible')
title('DOC vs AR and Sweep')
xlabel('Aspect Ratio')
ylabel('Sweep Angle')
zlabel('DOC ($)')
hold off
% Contour DOC Plot
figure
```

```
[mesh_AR,mesh_Sweep] = meshgrid(range_AR,range_Sweep);
mesh_DOC = griddata(table3d_AR,table3d_Sweep,table3d_DOC,mesh_AR,mesh_Sweep);
[C,h] = contourf(mesh_AR,mesh_Sweep,mesh_DOC,20);
clabel(C,h)
hold on
plot(optimal_AR,optimal_Sweep,'r.','MarkerSize',30,'DisplayName',sprintf('Point of Lowest DOC Possible: %0.4f$',optimal_DOC))
plot(optimal_Weight_AR,optimal_Weight_Sweep,'y.','MarkerSize',30,'DisplayName',sprintf('Point of Lowest Weight Possible: %0.4flbs',optimal_Weight_Weight))
legend
title('DOC vs AR and Sweep: Composite')
xlabel('Aspect Ratio')
ylabel('Sweep Angle')
hold off
% 3D Weight Plot
figure
scatter3(table3d_AR,table3d_Sweep,table3d_Weight);
hold on
plot(fit_Weight)
plot3(optimal_AR,optimal_Sweep,optimal_Weight,'r.','MarkerSize',30,'DisplayName','Point of Lowest DOC Possible')
title('Take-off Weight vs AR and Sweep')
xlabel('Aspect Ratio')
ylabel('Sweep Angle')
zlabel('Weight (lbs)')
hold off
% Contour Weight Plot
figure
mesh_Weight = griddata(table3d_AR,table3d_Sweep,table3d_Weight,mesh_AR,mesh_Sweep);
[C,h] = contourf(mesh_AR,mesh_Sweep,mesh_Weight,20);
clabel(C,h)
hold on
plot(optimal_AR,optimal_Sweep,'r.','MarkerSize',30,'DisplayName',sprintf('Point of Lowest DOC Possible: %0.1flbs',optimal_Weight))
plot(optimal_Weight_AR,optimal_Weight_Sweep,'y.','MarkerSize',30,'DisplayName',sprintf('Point of Lowest Weight Possible: %0.1flbs',optimal_Weight_Weight))
legend
title('Take-off Weight vs AR and Sweep')
xlabel('Aspect Ratio')
ylabel('Sweep Angle')
hold off
% DOC wrt AR Plot
v = length(range_Sweep)/1;
figure
hold on
for i = 1:1
    plot(table_AR(round(i*v),:),table_DOC(round(i*v),:),'DisplayName',sprintf('%0.1f',range_Sweep(round(i*v))))
end
plot(optimal_AR,optimal_DOC,'r.','MarkerSize',30,'DisplayName','Lowest DOC Possible')
legend
title(legend,'Sweep Angle')
title('DOC wrt Aspect Ratio: Composite')
xlabel('Aspect Ratio')
ylabel('DOC ($)')
hold off
% DOC wrt Sweep Plot
v = length(range_AR)/1;
figure
hold on
for i = 1:1
    plot(table_Sweep(:,round(i*v)),table_DOC(:,round(i*v)),'DisplayName',sprintf('%0.1f',range_AR(round
```

```
(i*v))))
end
plot(optimal_Sweep,optimal_DOC,'r.','MarkerSize',30,'DisplayName','Lowest DOC Possible')
legend
title(legend,'Aspect Ratio')
title('DOC wrt Sweep Angle: Composite')
xlabel('Sweep Angle')
ylabel('DOC ($)')
hold off
% Weight wrt AR Plot
v = length(range_Sweep)/1;
figure
hold on
for i = 1:1
    plot(table_AR(round(i*v),:),table_Weight(round(i*v),:),'DisplayName',sprintf('%0.1f',range_Sweep(round
(i*v))))
end
plot(optimal_AR,optimal_Weight,'r.','MarkerSize',30,'DisplayName','Lowest DOC Possible')
legend
title(legend,'Sweep Angle')
title('Weight wrt Aspect Ratio')
xlabel('Aspect Ratio')
ylabel('Weight (lbs)')
hold off
% Weight wrt Sweep Plot
v = length(range_AR)/1;
figure
hold on
for i = 1:1
    plot(table_Sweep(:,round(i*v)),table_Weight(:,round(i*v)),'DisplayName',sprintf('%0.1f',range_AR(round
(i*v))))
end
plot(optimal_Sweep,optimal_Weight,'r.','MarkerSize',30,'DisplayName','Lowest DOC Possible')
legend
title(legend,'Aspect Ratio')
title('Weight wrt Sweep Angle')
xlabel('Sweep Angle')
ylabel('Weight (lbs)')
hold off
% Range Chart
figure
plot(chart_Range,chart_Payload);
hold on
title('Effect of Payload Weight on Range')
xlabel('Range (mi)')
ylabel('Payload (lbs)')
hold off
```

Code for Comparison
Plots for DOC

```
%% Start
close all
clear
clc

%% Prompts

prompt_0 = "Is this unique (1), sample (2), or quiz (3) configuration? ";
type_inputs = input(prompt_0);

prompt_1 = "Would you like to draw graphs and fit curves? 1 for no, 2 for yes: ";
type_graph = input(prompt_1);

prompt_2 = "Please input 1 for conventional wing, 2 for supercritical: ";
type_airfoil = input(prompt_2);

prompt_3 = "Please enter what type of engine you are using, 1 for JT9D Advanced, 2 for JT9D, and 3 for JT8D↙
";
type_engine = input(prompt_3);

prompt_4 = "Please enter how many engines the aircraft has (2, 3, or 4): ";
type_engineqty = input(prompt_4);

if type_engineqty == 1
    type_engineqty = 2;
end

prompt_5 = "Are the engines mounted on the wing (1) or fuselage (2)? ";
type_mount = input(prompt_5);

prompt_6 = "Is the structure conventional (1), hybrid (2), or composites (3)? ";
type_structure = input(prompt_6);

prompt_7 = "Is this flight domestic (1) or international (2)? ";
type_flight = input(prompt_7);

if type_mount == 1
    K_w = 1;
    K_ts = 0.17;
elseif type_mount ==2
    K_w = 1.03;
    K_ts = 0.25;
end

%% Fixed Variables

l = 5; % number of lines in graphs at the end

range_AR = [6:.5:14];
Sweep = 36;

if type_inputs == 1 %Unique
    Taper = 0.35;

    height = 35000; %ft
    M = 0.82;
    Range_scheduled = 4000;
    FF_max = 1;
    V_approach = 135;

    TOFL = 6000;
```

```
    PAX = 200; %passengers
    Abreast = 6; %number of passengers abreast
    Aisles = 1;
    Weight_cargo = 4000; %weight of cargo
    Number_flightcrew = 2; %number of flight crew
    Number_stewards = round((PAX/50)+0.49); %number of stewards
elseif type_inputs == 2 %Sample
    Taper = 0.35;

    height = 35000; %ft
    M = 0.82;
    Range_scheduled = 6000;
    FF_max = .75;
    V_approach = 140;
    TOFL = 9000;

    PAX = 275; %passengers
    Abreast = 8; %number of passengers abreast
    Aisles = 2;
    Weight_cargo = 12000; %weight of cargo
    Number_flightcrew = 2; %number of flight crew
    Number_stewards = round((PAX/50)+0.49); %number of stewards

    type_engineqty = 3;
elseif type_inputs ==3 %Quiz
    Taper = 0.35;

    height = 35000; %ft
    M = 0.82;
    Range_scheduled = 4000;
    FF_max = 1;
    V_approach = 135;
    TOFL = 6000;

    PAX = 200; %passengers
    Abreast = 6; %number of passengers abreast
    Aisles = 2;
    Weight_cargo = 4000; %weight of cargo
    Number_flightcrew = 2; %number of flight crew
    Number_stewards = round((PAX/50)+0.49); %number of stewards
end

gamma = 1.4;
R = 1718;
T = 218.9; %K
rho = .023751; %lb/ft^3
sigma = .953; %Ratio between local and sea level densities
delta = .2360; %Ratio between local and sea level pressures

%a = sqrt(gamma*R*T);
a = 576.4;

%% Digitized Plots

Table = readtable('Tables.xlsx');
Table = Table{:,:};

if type_airfoil == 1
% Figure 1a: Effect of Wing Sweep and Thickness on Mdiv (Conventional)
    x1 = Table(1:165,1);
    y1 = Table(1:165,3);
    z1 = Table(1:165,2);
```

```
    [fit_TC,info_TC] = fit([x1,y1],z1,'poly23');

% Figure 1b: Effect of Wing Sweep and Thickness on Mdiv (Supercritical)
elseif type_airfoil == 2
    x2 = Table(1:211,4);
    y2 = Table(1:211,6);
    z2 = Table(1:211,5);
    [fit_TC,info_TC] = fit([x2,y2],z2,'poly35');
end
% Figure 2: Wing Drag Divergence Change Due to Lift
if type_airfoil == 1
    x3 = Table(1:25,7);
    y3 = Table(1:25,8);
    [fit_MDIVdelta,info_MDIVdelta] = fit(y3,x3,'power2');
elseif type_airfoil == 2
    x4 = Table(1:29,9);
    y4 = Table(1:29,10);
    [fit_MDIVdelta,info_MDIVdelta] = fit(y4,x4,'power2');
end

% Figure 3: CLmax Conceptual Design Estimates
x5 = Table(1:34,11);
y5 = Table(1:34,12);
[fit_CLmax_takeoff,info_CLmax_takeoff] = fit(x5,y5,'smoothingspline');
x6 = Table(1:31,13);
y6 = Table(1:31,14);
[fit_CLmax_landing,info_CLmax_landing] = fit(x6,y6,'smoothingspline');

% Figure 4: JT8D-9 Fuel Fraction Estimate for All Out Range
x7 = Table(1:41,15);
y7 = Table(1:41,16);
[fit_FF,info_FF] = fit(x7,y7,'cubicinterp'); % FF is Fuel Fraction or W_f/W_to

% Figure 5: Jet Aircraft TOFL with Engine Failure
x8 = Table(1:58,17);
y8 = Table(1:58,18);
[fit_k_2ENG,info_k_2ENG] = fit(y8,x8,'poly2');
x9 = Table(1:63,19);
y9 = Table(1:63,20);
[fit_k_3ENG,info_k_3ENG] = fit(y9,x9,'poly2');
x10 = Table(1:67,21);
y10 = Table(1:67,22);
[fit_k_4ENG,info_k_4ENG] = fit(y10,x10,'poly2');

% Cf
x11 = Table(1:50,23);
y11 = Table(1:50,24);
y11 = y11.*(1e-3);
[fit_Cf, info_Cf] = fit(x11,y11,'power2');

% Body Form Factor
x12 = Table(1:12,25);
y12 = Table(1:12,26);
[fit_k, info_k] = fit(x12,y12,'exp2');

% Max Climb Thrust @ 15000ft
x13 = Table(1:34,27);
y13 = Table(1:34,28);
[fit_T_avbl15, info_T_avbl15] = fit(x13,y13,'smoothingspline');
x14 = Table(1:9,29);
y14 = Table(1:9,30);
[fit_SFC_15, info_SFC_15] = fit(x14,y14,'smoothingspline');
```

```
% Max Climb Thrust @ 25000ft
x15 = Table(1:30,31);
y15 = Table(1:30,32);
[fit_T_avbl25, info_T_avbl25] = fit(x15,y15,'smoothingspline');
x = Table(1:6,33);
y = Table(1:6,34);
[fit_SFC_25, info_SFC_25] = fit(x,y,'smoothingspline');

% Max Climb and Thrust @ 35000ft
x16 = Table(1:9,37);
y16 = Table(1:9,38);
[fit_SFC_35, info_SFC_35] = fit(x16,y16,'lin');

% Profile Drag at Takeoff
x17 = Table(1:60,39);
y17 = Table(1:60,40);
[fit_CDp_takeoff, info_CDp_takeoff] = fit(y17,x17,'poly5');

% Profile Drag at Landing
x18 = Table(1:60,41);
y18 = Table(1:60,42);
[fit_CDp_landing, info_CDp_landing] = fit(y18,x18,'poly5');

% Clean Wing CLmax
x19 = Table(1:69,43);
y19 = Table(1:69,45);
z19 = Table(1:69,44);
[fit_CLmax_clean, info_CLmax_clean] = fit([x19,y19],z19,'poly32');

% JT9D Dry Takeoff
x20 = Table(1:18,46);
y20 = Table(1:18,47);
[fit_JT9D_takeoffthrust, info_JT9D_takeoffthrust] = fit(x20,y20,'lin');

% JT9D Climb
x21 = Table(1:26,48);
y21 = Table(1:26,49);
[fit_JT9D_climbthrust, info_JT9D_climbthrust] = fit(x21,y21,'lin');

%% Figures

if type_graph == 2

    figure
    plot(fit_TC_conv,[x1,y1],z1);

    figure
    plot(fit_TC_sc,[x2,y2],z2);
    hold on
    ylim([0 40]);
    zlim([0.07 .2]);
    hold off

    figure
    hold on
    plot(x3,y3)
    plot(x4,y4)
    xlim([-.04 .02])
    ylim([.4 .7])
    % plot(fit_MDIVdelta_conv)
    % plot(fit_MDIVdelta_sc)
```

```
    % xlim([-1 2])
    hold off

    figure
    hold on
    plot(x5,y5);
    plot(x6,y6);
    hold off

    figure
    plot(fit_FF);

    figure
    hold on
    plot(x8,y8,x9,y9,x10,y10)
    hold off

    figure
    plot(fit_Cf,x11,y11)
    hold on
    set(gca,'xscale','log')
    title('Skin Friction Coefficient wrt Reynolds Number')
    xlabel('Reynolds Number (Re)')
    ylabel('Coefficient of Skin Friction (k)')
    hold off

    figure
    plot(fit_k,x12,y12);
    hold on
    title('Body Form Factor wrt Fineness Ratio')
    xlabel('Fineness Ratio (l/d)')
    ylabel('Body Form Factor (k)')
    hold off

    figure
    plot(x17,y17,x18,y18)

    figure
    hold on
%   plot([z19,x19],y19)
    plot(fit_CLmax_clean)
    xlim([.04 .16]);
    ylim([0 35]);
    zlim([.8 1.3]);
    hold off
end

%% Placeholders

iteration = 1;

% table_engineqty_AR = AR;
% table_engineqty_EngineQty = type_engineqty;
% table_engineqty_DOC = DOC_perPAXMile;
% table_engineqty_Weight = W_takeoff;

SFC_jt9d = 0.61;
if type_engine == 1
    SFC_jt9d = 0.9 * SFC_jt9d;
end

%% Configuration Loop: Engine
```

```
i = 1;
for type_engineqty = 2:1:4
    j = 1;
    for AR = range_AR
        Weightincrement = 0;
        Thrust_reqtop_JT9D = 10001;
        while Thrust_reqtop_JT9D > 10000
    %% Range Loop
            Range = 0;
            Range_ao = 500000;
            Fuelincrement = 0;
            while abs(Range - Range_ao) > 2
    %% CL Loop
                CL_guess = 0.5;
                CL_ic = 0;
                while abs(CL_ic - CL_guess) > 0.001
                    Mdiv_delta = fit_MDIVdelta(CL_guess);
                    Mdiv = (M + .004) - Mdiv_delta;
                    TC = fit_TC(Mdiv,Sweep);
                    CLmax_landing = fit_CLmax_landing((((cosd(Sweep))^2)*((TC)^2)*AR));
                    CLmax_takeoff = fit_CLmax_takeoff((((cosd(Sweep))^2)*((TC)^2)*AR));%5
                    WS_landing = ((V_approach/1.3)^2) * (sigma * CLmax_landing) / 296;%6
                    V_cruise = M * a;
                    Range_ao = Range_scheduled + 200 + (V_cruise*(3/4));%7
                    FF_jt8d = fit_FF(Range_ao); %8
                    if type_inputs == 1
                        FF_jt9d = (FF_jt8d * (SFC_jt9d/.78)); %9
                    elseif type_inputs == 2
                        FF_jt9d = (FF_jt8d * (SFC_jt9d/.78)); %9
                    elseif type_inputs == 3
                        FF_jt9d = (FF_jt8d * (SFC_jt9d/.78)); %9
                    end
                    if type_engine == 1 || type_engine == 2
                        FF = FF_jt9d + Fuelincrement;
                    elseif type_engine == 3
                        FF = FF_jt8d + Fuelincrement;
                    end
                    WS_takeoff = WS_landing/(1-(FF_max*FF));
                    WS_ic = .965 * WS_takeoff;
                    CL_ic = WS_ic / (1481 * delta * M^2);
                    if abs(CL_ic - CL_guess) > 0.001
                        CL_guess = (CL_guess + CL_ic)/2;
                    end
                end
    %% TOFL
                if type_engineqty == 2
                    k_1 = fit_k_2ENG((TOFL/1000));
                elseif type_engineqty == 3
                    k_1 = fit_k_3ENG((TOFL/1000));
                elseif type_engineqty == 4
                    k_1 = fit_k_4ENG((TOFL/1000));
                end
                WT_7V_lo = (k_1/WS_takeoff) * sigma * CLmax_takeoff;
                V_lo = 1.2 *sqrt((296*WS_takeoff)/(sigma*CLmax_takeoff));
                M_lo = V_lo/(661/sqrt(sigma));
                Thrust_sls = fit_JT9D_takeoffthrust(0);
                Thrust_M = fit_JT9D_takeoffthrust(M_lo*0.7);
                WT = (WT_7V_lo * Thrust_M/Thrust_sls) + Weightincrement;
    %% Weight
                eta = 1.5*2.5;
                % Wing
                TC_bar = TC + 0.03;
```

```
                W_w = (.00945 * (AR^0.8) * ((1+Taper)^0.25) * K_w * (eta^0.5))/((TC_bar^0.4) * cosd(Sweep) ↵
(WS_takeoff^0.695)); %times W_takeoff^1.195
                % Fuselage
                K_f = 11.5;
                L_f = (3.76 * (PAX/Abreast) + 33.2);
                D_f = ((1.75*Abreast) + (1.58*Aisles) + 1);
                if type_flight == 2
                    L_f = 1.1 * L_f;
                    D_f = 1.1 * D_f;
                end
                W_fus = 0.6727 * K_f * (L_f^0.6) * (D_f^0.72) * (eta^0.3); %times W_takeoff^0.235
                % Landing Gear
                W_lg = 0.04; %times W_takeoff
                % Nacelle & Pylon
                W_np = (0.0555/WT); %times W_takeoff
                % Tail Surface
                W_ts = K_ts * W_w; %times W_takeoff^1.195
                % Power Plant
                W_pp = 1/(3.58*WT); %times W_takeoff
                if type_engine == 1
                    W_pp = W_pp * 1.1;
                end
                % Fuel
                W_f = 1.0275 * (FF);
                % Payload
                W_pl = 215*PAX + Weight_cargo;
                % Fixed Equipment
                W_fe = (132*PAX) + (300*type_engineqty) + (260*Number_flightcrew) + (170*Number_stewards);
                W_fe_wto = (0.035); %times W_takeoff
                if type_structure == 2
                    W_w = 0.7 * W_w;
                    W_ts = 0.7 * W_ts;
                    W_np = 0.8 * W_np;
                elseif type_structure == 3
                    W_w = 0.7 * W_w;
                    W_ts = 0.7 * W_ts;
                    W_fus = 0.85 * W_fus;
                    W_fe = 0.9 * W_fe;
                    W_fe_wto = 0.9 * W_fe_wto;
                    W_np = 0.8 * W_np;
                end
                a_W = W_w + W_ts;
                b_W = W_fus;
                c_W = W_lg + W_np + W_pp + W_f + W_fe_wto - 1;
                d_W = W_pl + W_fe;
                W_takeoff = 300000;
                difference_1 = W_takeoff;
                while abs(difference_1) > 0.1
                    if abs(difference_1) > 0.5
                        difference_1 = a_W*(W_takeoff^1.195) + b_W*(W_takeoff^0.235) + c_W*W_takeoff + d_W;
                        W_takeoff = W_takeoff + 4*(difference_1);
                    elseif abs(difference_1) <= 0.5
                        difference_1 = a_W*(W_takeoff^1.195) + b_W*(W_takeoff^0.235) + c_W*W_takeoff + d_W;
                        W_takeoff = W_takeoff + (difference_1);
                    end
                end
                W_fuel = W_f * W_takeoff;
    %% Wing Sizing
                S = W_takeoff/WS_takeoff;
                b = sqrt(AR*S);
                c_bar = S/b;
                Thrust_total = W_takeoff/WT;
```

```
            Thrust_engine = Thrust_total/type_engineqty;
    %% Drag
            Re_l = 2.852e6 * 0.5; % 0.5 * rho * v / mu
            % Wing
            Re_w = Re_l * c_bar;
            Cf_w = fit_Cf(Re_w);
            Swet_w = 2 * S * 1.02;
            z_w = ((2-M^2)*cosd(Sweep))/sqrt(1-((M^2).*(cosd(Sweep))));
            k_w = 1+(z_w*TC)+100*(TC^4);
            f_w = k_w*Cf_w*Swet_w;
            % Fuselage
            Swet_fus = 0.9*pi*D_f*L_f;
            Re_fus = Re_l * L_f;
            Cf_fus = fit_Cf(Re_fus);
            fineness_fus = L_f/D_f;
            k_fus = fit_k(fineness_fus);
            f_fus = k_fus*Cf_fus*Swet_fus;
            % Tail
            f_t = 0.35*f_w;
            % Nacelles
            Re_n = Re_l * c_bar;
            Swet_n = 2.1 * sqrt(Thrust_engine) * type_engineqty;
            Cf_n = fit_Cf(Re_n);
            k_n = 1.25;
            f_n = k_n*Cf_n*Swet_n;
            % Pylons
            f_p = 0.2*f_n;
            %Total
            f = f_w + f_fus + f_t + f_n + f_p;
            CD_overall = f/S;
            e = 1/(1.035 + (.38*CD_overall*pi*AR));
            CD_gear = CD_overall;
    %% Climb
            W_avgcl = (1.965) * W_takeoff / 2;
            h_avgcl = (20/35) * height;
            sigma_cl = .5702;
            V_ldmax = (12.9/(f*e)^(1/4)) * sqrt(W_avgcl/(sigma_cl*b));
            V_cl = 1.3 * V_ldmax;
            M_cl = V_cl/(973.1/1.68781);
            Thrust_reqcl = (sigma_cl*f*(V_cl^2)/296) + ((94.1/(sigma_cl*e))*((W_avgcl/b)^2)*(1/
(V_cl^2)));
            Thrust_cl15 = fit_T_avbl15(M_cl) * 1000;
            Thrust_cl25 = fit_T_avbl25(M_cl) * 1000;
            SFC_15 = fit_SFC_15(M_cl);
            SFC_25 = fit_SFC_25(M_cl);
            Thrust_cl = (Thrust_cl15 + Thrust_cl25)/2;
            SFC_cl = (SFC_15 + SFC_25)/2;
            if type_engine == 1
                SFC_cl = SFC_cl * 0.9;
            end
            Thrust_avbl = (Thrust_engine/Thrust_sls) * Thrust_cl;
            RC = (101*((type_engineqty*Thrust_avbl)-Thrust_reqcl)*V_cl)/W_avgcl;
            Time_cl = height/RC;
            Range_cl = V_cl * (Time_cl/60);
            W_fcl = (type_engineqty*Thrust_avbl) * SFC_cl * (Time_cl/60);
    %% Range
            W_0 = W_takeoff-W_fcl;
            W_I = (1-FF) * W_takeoff;
            CL_avg = ((W_0+W_I)/(2*S)) / (1481*delta*(M^2));
            CD_i = (CL_avg^2)/(pi*AR*e);
            CD_c = 0.001;
            CD = CD_overall + CD_i + CD_c;
```

```
            LD = CL_avg/CD;
            Thrust_reqcr = ((W_0 + W_I)/(2*LD));
            Thrust_reqcrjt9d = Thrust_reqcr * (Thrust_sls/Thrust_engine);
            Thrust_reqcreng = Thrust_reqcrjt9d / type_engineqty;
            SFC_cr = fit_SFC_35(Thrust_reqcreng);
            if type_engine == 1
                SFC_cr = SFC_cr * 0.9;
            end
            Range_cr = (V_cruise/SFC_cr) * LD * log(W_0/W_I);
            Range = Range_cl + Range_cr;
            difference_2 = Range-Range_ao;
            if     difference_2 <  -200 %200 or less
                Fuelincrement = Fuelincrement + 0.01;
            elseif difference_2 >= 200 %200 or more
                Fuelincrement = Fuelincrement - 0.01;
            elseif difference_2 <  -18 && (Range-Range_ao) >= -200 %Between -20 and -200
                Fuelincrement = Fuelincrement + 0.001;
            elseif difference_2 >  18  && (Range-Range_ao) <= 200 %Between 20 and 200
                Fuelincrement = Fuelincrement - 0.001;
            elseif difference_2 <  -3    && (Range-Range_ao) >= -18 %Between 0 and -20
                Fuelincrement = Fuelincrement + 0.0001;
            elseif difference_2 >= 3    && (Range-Range_ao) <= 18 %Between 0 and 20
                Fuelincrement = Fuelincrement - 0.0001;
            elseif difference_2 <  0    && (Range-Range_ao) >= -3 %Between 0 and -20
                Fuelincrement = Fuelincrement + 0.00001;
            elseif difference_2 >= 0    && (Range-Range_ao) <= 3 %Between 0 and 20
                Fuelincrement = Fuelincrement - 0.00001;
            end
        end
    %% Top of Climb
        CL_ic_top = (W_0/S)/(1481*delta*M^2);
        CDi_top = (CL_ic_top^2) / (pi*AR*e);
        CD_top = CD_overall + CDi_top + 0.001;
        LD_top = CL_ic_top / CD_top;
        if type_inputs == 2
            LD_top = 15.05; % SAMPLE
        end
        Thrust_reqtop = W_0 / LD_top;
        Thrust_reqtop_engine = Thrust_reqtop / type_engineqty;
        Thrust_reqtop_JT9D = (Thrust_sls/Thrust_engine) * Thrust_reqtop_engine;

    %% Climb Gradients
        % 1st Segment
        CL_seg1 = CLmax_takeoff/(1.2^2);
        CL_to_CLmax_to = CL_seg1/CLmax_takeoff;
        CDp_seg1 = fit_CDp_takeoff(CL_to_CLmax_to);
        CD_seg1 = CD_overall + CDp_seg1 + CD_gear + ((CL_seg1^2)/(pi*AR*e));
        LD_seg1 = CL_seg1/CD_seg1;
        Thrust_reqseg1 = W_takeoff/LD_seg1;
        Thrust_avblseg1 = (Thrust_engine/Thrust_sls) * fit_JT9D_takeoffthrust(M_lo);
        Grad_1 = ((type_engineqty-1)*Thrust_avblseg1-Thrust_reqseg1)*100/W_takeoff;
        % 2nd Segment
        CD_seg2 = CD_overall + CDp_seg1 + ((CL_seg1^2)/(pi*AR*e));
        LD_seg2 = CL_seg1/CD_seg2;
        Thrust_reqseg2 = W_takeoff/LD_seg2;
        Grad_2 = ((type_engineqty-1)*Thrust_avblseg1-Thrust_reqseg2)*100/W_takeoff;
        % 3rd Segment
        CLmax_clean = fit_CLmax_clean(TC,Sweep);
        V_seg3 = 1.2*sqrt((296*WS_takeoff)/(.925*CLmax_clean));
        M_seg3 = V_seg3/659;
        CL_seg3 = CLmax_clean/(1.2^2);
        CD_seg3 = CD_overall + ((CL_seg3^2)/(pi*AR*e));
```

```
            LD_seg3 = CL_seg3/CD_seg3;
            Thrust_reqseg3 = W_takeoff/LD_seg3;
            Thrust_avblseg3 = (Thrust_engine/Thrust_sls) * fit_JT9D_climbthrust(M_seg3);
            Grad_3 = ((type_engineqty-1)*Thrust_avblseg3-Thrust_reqseg3)*100/W_takeoff;
            % Approach
            CL_approach = CLmax_takeoff/(1.3^2);
            CL_approach_CLmax_landing = CL_approach/CLmax_takeoff;
            CDp_approach = fit_CDp_takeoff(CL_approach_CLmax_landing);
            CD_approach = CD_overall + CDp_approach + (CL_approach^2)/(pi*AR*e);
            LD_approach = CL_approach/CD_approach;
            W_landing = WS_landing * S;
            Thrust_reqapproach = W_landing/LD_approach;
            V_approach_grad = sqrt((296*WS_landing)/(.953*CL_approach));
            M_approach = V_approach_grad/659;
            Thrust_avblapproach = (Thrust_engine/Thrust_sls) * fit_JT9D_climbthrust(M_approach);
            Grad_approach = ((type_engineqty-1)*Thrust_avblapproach-Thrust_reqapproach)*100/W_landing;
            %Landing
            CL_landing = CLmax_landing/1.3^2;
            CDp_landing = fit_CDp_landing(1/1.3^2);
            CD_landing = CD_overall + CDp_landing + CD_gear + (CL_landing^2)/(pi*AR*e);
            LD_landing = CL_landing/CD_landing;
            Thrust_reqlanding = W_landing/LD_landing;
            V_landing = sqrt((296*WS_landing)/(0.953*CL_landing));
            M_landing = V_landing/659;
            Thrust_avbllanding = (Thrust_engine/Thrust_sls) * fit_JT9D_takeoffthrust(M_landing);
            Grad_landing = ((type_engineqty*Thrust_avbllanding) - Thrust_reqlanding) * 100 / W_landing;
            % Climb Requirements (pg. 39)
            if type_engineqty == 2
                Grad_1_min = 0;
                Grad_2_min = 2.4;
                Grad_3_min = 1.2;
                Grad_approach_min = 2.1;
                Grad_landing_min = 3.2;
            elseif type_engineqty == 3
                Grad_1_min = 0.3;
                Grad_2_min = 2.7;
                Grad_3_min = 1.5;
                Grad_approach_min = 2.4;
                Grad_landing_min = 3.2;
            elseif type_engineqty == 4
                Grad_1_min = 0.5;
                Grad_2_min = 3.0;
                Grad_3_min = 1.7;
                Grad_approach_min = 2.7;
                Grad_landing_min = 3.2;
            end

            if Thrust_reqtop_JT9D > 10000 || Grad_1 < Grad_1_min || Grad_2 < Grad_2_min || Grad_3 <
Grad_3_min || Grad_approach < Grad_approach_min || Grad_landing < Grad_landing_min
                Weightincrement = Weightincrement - 0.1;
                disp('Gradient Failed')
            end
        end

    %% DOC
        D = Range_scheduled * 1.15;
        %Time_gm = (D / (11.866+0.040669*D))/60;
        Time_gm = 0.25;
        Time_cl_hr = Time_cl/60;
        Time_d = 0;
        Time_am = 0.10;
        if D <= 1400
```

```
    k_a = (7+0.015*D);
elseif D > 1400
    k_a = 0.02*D;
end
D_d = 0;
Time_cr = ((D+k_a+20)-(Range_cl*1.15+D_d))/(V_cruise*1.15);
V_B = D / (Time_gm + Time_cl_hr + Time_d + Time_cr + Time_am);
% Block Time
Time_b = Time_gm + Time_cl_hr + Time_d + Time_cr + Time_am;
% Block Fuel
F_cr_am = Thrust_reqcr * SFC_cr * (Time_cr+Time_am);
F_cl = W_fcl;
F_b = F_cr_am + F_cl;
% Flying Operations Cost
% a. Flight Crew
P = W_pl / 2000;
if Number_flightcrew == 2
    CostPerBlockHr = 17.849 * (V_cruise*W_takeoff/1e5)^0.3 + 40.83;
else
    CostPerBlockHr = 24.261 * (V_cruise*W_takeoff/1e5)^0.3 + 57.62;
end
C_TM_FlightCrew = CostPerBlockHr / (V_B * P);
% b. Fuel and Oil
if type_flight == 1
    C_ft = 0.0438;
elseif type_flight == 2
    C_ft = 0.0625;
end
C_ot = 2.15;
C_TM_FuelOil = ( 1.02*F_b*C_ft + type_engineqty*C_ot*Time_b*0.135 ) / (D*P);
% c. Hull Insurance Costs
W_a = W_takeoff * (1-FF) - W_pl - (W_pp*W_takeoff);
C_a = 2.4e6 + (87.5*W_a);
C_e = 590000 + (16*Thrust_engine);
if type_inputs == 2
    C_e = 1314942;
end
C_T = (type_engineqty*C_e) + C_a;
IR_A = 0.01;
U = 630 + 4000/( 1 + 1/(Time_b + 0.5)); % block hrs/yr
C_TM_HullInsurance = IR_A * C_T / (U * V_B * P);
% Direct Maintenance
% a. Airframe-Labor
K_FHa = 4.9169 * log10(W_a/1000) - 6.425;
K_FCa = 0.21256 * log10(W_a/1000)^3.7375;
T_F = Time_b - Time_gm;
R_L = 8.6;
C_TM_a_AirframeLabor = (K_FHa * T_F + K_FCa) * R_L * (1 + 0.29*(M-1))^1.5 / (V_B * Time_b * P);
% b. Airframe Material
C_FHa = ((1.5994*C_a)/1e6) + 3.4263;
C_FCa = ((1.9229*C_a)/1e6) + 2.2504;
C_TM_b_AirframeMaterial = (C_FHa*T_F+C_FCa) / (V_B*Time_b*P);
% c. Engine Labor
K_FHe = (type_engineqty*(Thrust_engine/1e3))/(0.82715*(Thrust_engine/1e3)+ 13.639);
K_FCe = 0.2 * type_engineqty;
C_TM_c_EngineLabor = ((K_FHe*T_F+K_FCe)*R_L) / (V_B*Time_b*P);
% d. Engine Material
C_FHe = (((28.2353*C_e)/1e6) - 6.5176) * type_engineqty;
C_FCe = (((3.6698*C_e)/1e6) + 1.3685) * type_engineqty;
C_TM_d_EngineMaterial = (C_FHe*T_F+C_FCe) / (V_B*Time_b*P);
if type_engine == 1
    C_TM_d_EngineMaterial = C_TM_d_EngineMaterial * 1.1;
```

```
        end
        % e. Total Maintenance - Burdened
        C_TM_e_Maintenance = (C_TM_a_AirframeLabor + C_TM_b_AirframeMaterial + C_TM_c_EngineLabor + C_TM_d_EngineMaterial)*2;
        % Depreciation
        D_a = 14;
        C_TM_Depreciation = (1/(V_B*P)) * (C_T + 0.06*(C_T-type_engineqty*C_e)+0.3*type_engineqty*C_e) / (D_a*U);
        % Total DOC
        DOC_perTonMile = C_TM_FlightCrew + C_TM_FuelOil + C_TM_HullInsurance + C_TM_e_Maintenance + C_TM_Depreciation;
        DOC_perPAXMile = DOC_perTonMile * P/PAX;

    %% Data Collection
        % Plots
        table_engineqty_AR(i,j) = AR;
        table_engineqty_EngineQty(i,j) = type_engineqty;
        table_engineqty_DOC(i,j) = DOC_perPAXMile;
        table_engineqty_Weight(i,j) = W_takeoff;
        table_engineqty_EngineMaterialCost(i,j) = C_TM_d_EngineMaterial;
        table_engineqty_EngineLabor(i,j) = C_TM_c_EngineLabor;

        iteration = iteration + 1;
        j = j+1;
    end
    i = i+1;
end

figure
hold on
for i = [1:1:(type_engineqty-1)]
    plot(table_engineqty_AR(i,:),table_engineqty_DOC(i,:))
end
legend('2','3','4')
title(legend,'Engine Quantity')
title('DOC wrt Engine Quantity')
xlabel('Aspect Ratio')
ylabel('DOC ($)')
hold off

figure
hold on
for i = [1:1:(type_engineqty-1)]
    plot(table_engineqty_AR(i,:),table_engineqty_EngineMaterialCost(i,:))
end
legend('2','3','4')
title(legend,'Engine Quantity')
title('Engine Material Costs wrt Engine Quantity')
xlabel('Aspect Ratio')
ylabel('Material Costs ($)')
hold off

figure
hold on
for i = [1:1:(type_engineqty-1)]
    plot(table_engineqty_AR(i,:),table_engineqty_EngineLabor(i,:))
end
legend('2','3','4')
title(legend,'Engine Quantity')
title('Engine Labor Costs wrt Engine Quantity')
xlabel('Aspect Ratio')
ylabel('Labor Costs ($)')
```

```
hold off

%% Configuration Loop: Airfoil Shape
iteration = 1;
type_engineqty = 2;
i = 1;
for type_airfoil = 1:2
    if type_airfoil == 1
    % Figure 1a: Effect of Wing Sweep and Thickness on Mdiv (Conventional)
        x1 = Table(1:165,1);
        y1 = Table(1:165,3);
        z1 = Table(1:165,2);
        [fit_TC,info_TC] = fit([x1,y1],z1,'poly23');

    % Figure 1b: Effect of Wing Sweep and Thickness on Mdiv (Supercritical)
    elseif type_airfoil == 2
        x2 = Table(1:211,4);
        y2 = Table(1:211,6);
        z2 = Table(1:211,5);
        [fit_TC,info_TC] = fit([x2,y2],z2,'poly35');
    end
    j = 1;
    for AR = range_AR
        Weightincrement = 0;
        Thrust_reqtop_JT9D = 10001;
        while Thrust_reqtop_JT9D > 10000
    %% Range Loop
            Range = 0;
            Range_ao = 500000;
            Fuelincrement = 0;
            while abs(Range - Range_ao) > 2
    %% CL Loop
                CL_guess = 0.5;
                CL_ic = 0;
                while abs(CL_ic - CL_guess) > 0.001
                    Mdiv_delta = fit_MDIVdelta(CL_guess);
                    Mdiv = (M + .004) - Mdiv_delta;
                    TC = fit_TC(Mdiv,Sweep);
                    CLmax_landing = fit_CLmax_landing((((cosd(Sweep))^2)*((TC)^2)*AR));
                    CLmax_takeoff = fit_CLmax_takeoff((((cosd(Sweep))^2)*((TC)^2)*AR));%5
                    WS_landing = ((V_approach/1.3)^2) * (sigma * CLmax_landing) / 296;%6
                    V_cruise = M * a;
                    Range_ao = Range_scheduled + 200 + (V_cruise*(3/4));%7
                    FF_jt8d = fit_FF(Range_ao); %8
                    if type_inputs == 1
                        FF_jt9d = (FF_jt8d * (SFC_jt9d/.78)); %9
                    elseif type_inputs == 2
                        FF_jt9d = (FF_jt8d * (SFC_jt9d/.78)); %9
                    elseif type_inputs == 3
                        FF_jt9d = (FF_jt8d * (SFC_jt9d/.78)); %9
                    end
                    if type_engine == 1 || type_engine == 2
                        FF = FF_jt9d + Fuelincrement;
                    elseif type_engine == 3
                        FF = FF_jt8d + Fuelincrement;
                    end
                    WS_takeoff = WS_landing/(1-(FF_max*FF));
                    WS_ic = .965 * WS_takeoff;
                    CL_ic = WS_ic / (1481 * delta * M^2);
                    if abs(CL_ic - CL_guess) > 0.001
                        CL_guess = (CL_guess + CL_ic)/2;
                    end
```

```
                end
    %% TOFL
                if type_engineqty == 2
                    k_1 = fit_k_2ENG((TOFL/1000));
                elseif type_engineqty == 3
                    k_1 = fit_k_3ENG((TOFL/1000));
                elseif type_engineqty == 4
                    k_1 = fit_k_4ENG((TOFL/1000));
                end
                WT_7V_lo = (k_1/WS_takeoff) * sigma * CLmax_takeoff;
                V_lo = 1.2 *sqrt((296*WS_takeoff)/(sigma*CLmax_takeoff));
                M_lo = V_lo/(661/sqrt(sigma));
                Thrust_sls = fit_JT9D_takeoffthrust(0);
                Thrust_M = fit_JT9D_takeoffthrust(M_lo*0.7);
                WT = (WT_7V_lo * Thrust_M/Thrust_sls) + Weightincrement;
    %% Weight
                eta = 1.5*2.5;
                % Wing
                TC_bar = TC + 0.03;
                W_w = (.00945 * (AR^0.8) * ((1+Taper)^0.25) * K_w * (eta^0.5))/((TC_bar^0.4) * cosd(Sweep) ↙
(WS_takeoff^0.695)); %times W_takeoff^1.195
                % Fuselage
                K_f = 11.5;
                L_f = (3.76 * (PAX/Abreast) + 33.2);
                D_f = ((1.75*Abreast) + (1.58*Aisles) + 1);
                if type_flight == 2
                    L_f = 1.1 * L_f;
                    D_f = 1.1 * D_f;
                end
                W_fus = 0.6727 * K_f * (L_f^0.6) * (D_f^0.72) * (eta^0.3);%times W_takeoff^0.235
                % Landing Gear
                W_lg = 0.04; %times W_takeoff
                % Nacelle & Pylon
                W_np = (0.0555/WT); %times W_takeoff
                % Tail Surface
                W_ts = K_ts * W_w; %times W_takeoff^1.195
                % Power Plant
                W_pp = 1/(3.58*WT); %times W_takeoff
                if type_engine == 1
                    W_pp = W_pp * 1.1;
                end
                % Fuel
                W_f = 1.0275 * (FF);
                % Payload
                W_pl = 215*PAX + Weight_cargo;
                % Fixed Equipment
                W_fe = (132*PAX) + (300*type_engineqty) + (260*Number_flightcrew) + (170*Number_stewards);
                W_fe_wto = (0.035); %times W_takeoff
                if type_structure == 2
                    W_w = 0.7 * W_w;
                    W_ts = 0.7 * W_ts;
                    W_np = 0.8 * W_np;
                elseif type_structure == 3
                    W_w = 0.7 * W_w;
                    W_ts = 0.7 * W_ts;
                    W_fus = 0.85 * W_fus;
                    W_fe = 0.9 * W_fe;
                    W_fe_wto = 0.9 * W_fe_wto;
                    W_np = 0.8 * W_np;
                end
                a_W = W_w + W_ts;
                b_W = W_fus;
```

```
                c_W = W_lg + W_np + W_pp + W_f + W_fe_wto - 1;
                d_W = W_pl + W_fe;
                W_takeoff = 300000;
                difference_1 = W_takeoff;
                while abs(difference_1) > 0.1
                    if abs(difference_1) > 0.5
                        difference_1 = a_W*(W_takeoff^1.195) + b_W*(W_takeoff^0.235) + c_W*W_takeoff + d_W;
                        W_takeoff = W_takeoff + 4*(difference_1);
                    elseif abs(difference_1) <= 0.5
                        difference_1 = a_W*(W_takeoff^1.195) + b_W*(W_takeoff^0.235) + c_W*W_takeoff + d_W;
                        W_takeoff = W_takeoff + (difference_1);
                    end
                end
                W_fuel = W_f * W_takeoff;
    %% Wing Sizing
                S = W_takeoff/WS_takeoff;
                b = sqrt(AR*S);
                c_bar = S/b;
                Thrust_total = W_takeoff/WT;
                Thrust_engine = Thrust_total/type_engineqty;
    %% Drag
                Re_l = 2.852e6 * 0.5; % 0.5 * rho * v / mu
                % Wing
                Re_w = Re_l * c_bar;
                Cf_w = fit_Cf(Re_w);
                Swet_w = 2 * S * 1.02;
                z_w = ((2-M^2)*cosd(Sweep))/sqrt(1-((M^2).*(cosd(Sweep))));
                k_w = 1+(z_w*TC)+100*(TC^4);
                f_w = k_w*Cf_w*Swet_w;
                % Fuselage
                Swet_fus = 0.9*pi*D_f*L_f;
                Re_fus = Re_l * L_f;
                Cf_fus = fit_Cf(Re_fus);
                fineness_fus = L_f/D_f;
                k_fus = fit_k(fineness_fus);
                f_fus = k_fus*Cf_fus*Swet_fus;
                % Tail
                f_t = 0.35*f_w;
                % Nacelles
                Re_n = Re_l * c_bar;
                Swet_n = 2.1 * sqrt(Thrust_engine) * type_engineqty;
                Cf_n = fit_Cf(Re_n);
                k_n = 1.25;
                f_n = k_n*Cf_n*Swet_n;
                % Pylons
                f_p = 0.2*f_n;
                %Total
                f = f_w + f_fus + f_t + f_n + f_p;
                CD_overall = f/S;
                e = 1/(1.035 + (.38*CD_overall*pi*AR));
                CD_gear = CD_overall;
    %% Climb
                W_avgcl = (1.965) * W_takeoff / 2;
                h_avgcl = (20/35) * height;
                sigma_cl = .5702;
                V_ldmax = (12.9/(f*e)^(1/4)) * sqrt(W_avgcl/(sigma_cl*b));
                V_cl = 1.3 * V_ldmax;
                M_cl = V_cl/(973.1/1.68781);
                Thrust_reqcl = (sigma_cl*f*(V_cl^2)/296) + ((94.1/(sigma_cl*e))*((W_avgcl/b)^2)*(1/
(V_cl^2)));
                Thrust_cl15 = fit_T_avbl15(M_cl) * 1000;
                Thrust_cl25 = fit_T_avbl25(M_cl) * 1000;
```

```
            SFC_15 = fit_SFC_15(M_cl);
            SFC_25 = fit_SFC_25(M_cl);
            Thrust_cl = (Thrust_cl15 + Thrust_cl25)/2;
            SFC_cl = (SFC_15 + SFC_25)/2;
            if type_engine == 1
                SFC_cl = SFC_cl * 0.9;
            end
            Thrust_avbl = (Thrust_engine/Thrust_sls) * Thrust_cl;
            RC = (101*((type_engineqty*Thrust_avbl)-Thrust_reqcl)*V_cl)/W_avgcl;
            Time_cl = height/RC;
            Range_cl = V_cl * (Time_cl/60);
            W_fcl = (type_engineqty*Thrust_avbl) * SFC_cl * (Time_cl/60);
    %% Range
            W_0 = W_takeoff-W_fcl;
            W_I = (1-FF) * W_takeoff;
            CL_avg = ((W_0+W_I)/(2*S)) / (1481*delta*(M^2));
            CD_i = (CL_avg^2)/(pi*AR*e);
            CD_c = 0.001;
            CD = CD_overall + CD_i + CD_c;
            LD = CL_avg/CD;
            Thrust_reqcr = ((W_0 + W_I)/(2*LD));
            Thrust_reqcrjt9d = Thrust_reqcr * (Thrust_sls/Thrust_engine);
            Thrust_reqcreng = Thrust_reqcrjt9d / type_engineqty;
            SFC_cr = fit_SFC_35(Thrust_reqcreng);
            if type_engine == 1
                SFC_cr = SFC_cr * 0.9;
            end
            Range_cr = (V_cruise/SFC_cr) * LD * log(W_0/W_I);
            Range = Range_cl + Range_cr;
            difference_2 = Range-Range_ao;
            if     difference_2 <  -200 %200 or less
                Fuelincrement = Fuelincrement + 0.01;
            elseif difference_2 >= 200 %200 or more
                Fuelincrement = Fuelincrement - 0.01;
            elseif difference_2 <  -18 && (Range-Range_ao) >= -200 %Between -20 and -200
                Fuelincrement = Fuelincrement + 0.001;
            elseif difference_2 >  18  && (Range-Range_ao) <= 200 %Between 20 and 200
                Fuelincrement = Fuelincrement - 0.001;
            elseif difference_2 <  -3    && (Range-Range_ao) >= -18 %Between 0 and -20
                Fuelincrement = Fuelincrement + 0.0001;
            elseif difference_2 >= 3    && (Range-Range_ao) <= 18 %Between 0 and 20
                Fuelincrement = Fuelincrement - 0.0001;
            elseif difference_2 <  0    && (Range-Range_ao) >= -3 %Between 0 and -20
                Fuelincrement = Fuelincrement + 0.00001;
            elseif difference_2 >= 0    && (Range-Range_ao) <= 3 %Between 0 and 20
                Fuelincrement = Fuelincrement - 0.00001;
            end
        end
    %% Top of Climb
        CL_ic_top = (W_0/S)/(1481*delta*M^2);
        CDi_top = (CL_ic_top^2) / (pi*AR*e);
        CD_top = CD_overall + CDi_top + 0.001;
        LD_top = CL_ic_top / CD_top;
        if type_inputs == 2
            LD_top = 15.05; % SAMPLE
        end
        Thrust_reqtop = W_0 / LD_top;
        Thrust_reqtop_engine = Thrust_reqtop / type_engineqty;
        Thrust_reqtop_JT9D = (Thrust_sls/Thrust_engine) * Thrust_reqtop_engine;

    %% Climb Gradients
        % 1st Segment
```

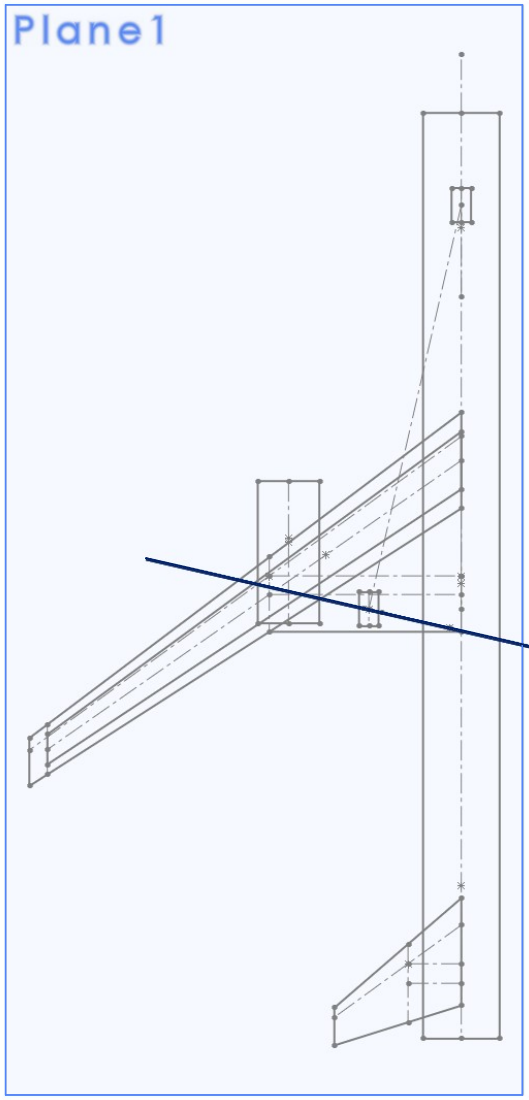
```
CL_seg1 = CLmax_takeoff/(1.2^2);
CL_to_CLmax_to = CL_seg1/CLmax_takeoff;
CDp_seg1 = fit_CDp_takeoff(CL_to_CLmax_to);
CD_seg1 = CD_overall + CDp_seg1 + CD_gear + ((CL_seg1^2)/(pi*AR*e));
LD_seg1 = CL_seg1/CD_seg1;
Thrust_reqseg1 = W_takeoff/LD_seg1;
Thrust_avblseg1 = (Thrust_engine/Thrust_sls) * fit_JT9D_takeoffthrust(M_lo);
Grad_1 = ((type_engineqty-1)*Thrust_avblseg1-Thrust_reqseg1)*100/W_takeoff;
% 2nd Segment
CD_seg2 = CD_overall + CDp_seg1 + ((CL_seg1^2)/(pi*AR*e));
LD_seg2 = CL_seg1/CD_seg2;
Thrust_reqseg2 = W_takeoff/LD_seg2;
Grad_2 = ((type_engineqty-1)*Thrust_avblseg1-Thrust_reqseg2)*100/W_takeoff;
% 3rd Segment
CLmax_clean = fit_CLmax_clean(TC,Sweep);
V_seg3 = 1.2*sqrt((296*WS_takeoff)/(.925*CLmax_clean));
M_seg3 = V_seg3/659;
CL_seg3 = CLmax_clean/(1.2^2);
CD_seg3 = CD_overall + ((CL_seg3^2)/(pi*AR*e));
LD_seg3 = CL_seg3/CD_seg3;
Thrust_reqseg3 = W_takeoff/LD_seg3;
Thrust_avblseg3 = (Thrust_engine/Thrust_sls) * fit_JT9D_climbthrust(M_seg3);
Grad_3 = ((type_engineqty-1)*Thrust_avblseg3-Thrust_reqseg3)*100/W_takeoff;
% Approach
CL_approach = CLmax_takeoff/(1.3^2);
CL_approach_CLmax_landing = CL_approach/CLmax_takeoff;
CDp_approach = fit_CDp_takeoff(CL_approach_CLmax_landing);
CD_approach = CD_overall + CDp_approach + (CL_approach^2)/(pi*AR*e);
LD_approach = CL_approach/CD_approach;
W_landing = WS_landing * S;
Thrust_reqapproach = W_landing/LD_approach;
V_approach_grad = sqrt((296*WS_landing)/(.953*CL_approach));
M_approach = V_approach_grad/659;
Thrust_avblapproach = (Thrust_engine/Thrust_sls) * fit_JT9D_climbthrust(M_approach);
Grad_approach = ((type_engineqty-1)*Thrust_avblapproach-Thrust_reqapproach)*100/W_landing;
%Landing
CL_landing = CLmax_landing/1.3^2;
CDp_landing = fit_CDp_landing(1/1.3^2);
CD_landing = CD_overall + CDp_landing + CD_gear + (CL_landing^2)/(pi*AR*e);
LD_landing = CL_landing/CD_landing;
Thrust_reqlanding = W_landing/LD_landing;
V_landing = sqrt((296*WS_landing)/(0.953*CL_landing));
M_landing = V_landing/659;
Thrust_avbllanding = (Thrust_engine/Thrust_sls) * fit_JT9D_takeoffthrust(M_landing);
Grad_landing = ((type_engineqty*Thrust_avbllanding) - Thrust_reqlanding) * 100 / W_landing;
% Climb Requirements (pg. 39)
if type_engineqty == 2
    Grad_1_min = 0;
    Grad_2_min = 2.4;
    Grad_3_min = 1.2;
    Grad_approach_min = 2.1;
    Grad_landing_min = 3.2;
elseif type_engineqty == 3
    Grad_1_min = 0.3;
    Grad_2_min = 2.7;
    Grad_3_min = 1.5;
    Grad_approach_min = 2.4;
    Grad_landing_min = 3.2;
elseif type_engineqty == 4
    Grad_1_min = 0.5;
    Grad_2_min = 3.0;
    Grad_3_min = 1.7;
```

```
                Grad_approach_min = 2.7;
                Grad_landing_min = 3.2;
            end

            if Thrust_reqtop_JT9D > 10000 || Grad_1 < Grad_1_min || Grad_2 < Grad_2_min || Grad_3 <
Grad_3_min || Grad_approach < Grad_approach_min || Grad_landing < Grad_landing_min
                Weightincrement = Weightincrement - 0.1;
                disp('Gradient Failed')
            end
        end

    %% DOC
        D = Range_scheduled * 1.15;
        Time_gm = (D / (11.866+0.040669*D))/60;
        Time_cl_hr = Time_cl/60;
        Time_d = 0;
        Time_am = 0.10;
        if D <= 1400
            k_a = (7+0.015*D);
        elseif D > 1400
            k_a = 0.02*D;
        end
        D_d = 0;
        Time_cr = ((D+k_a+20)-(Range_cl*1.15+D_d))/(V_cruise*1.15);
        V_B = D / (Time_gm + Time_cl_hr + Time_d + Time_cr + Time_am);
        % Block Time
        Time_b = Time_gm + Time_cl_hr + Time_d + Time_cr + Time_am;
        % Block Fuel
        F_cr_am = Thrust_reqcr * SFC_cr * (Time_cr+Time_am);
        F_cl = W_fcl;
        F_b = F_cr_am + F_cl;
        % Flying Operations Cost
        % a. Flight Crew
        P = W_pl / 2000;
        if Number_flightcrew == 2
            CostPerBlockHr = 17.849 * (V_cruise*W_takeoff/1e5)^0.3 + 40.83;
        else
            CostPerBlockHr = 24.261 * (V_cruise*W_takeoff/1e5)^0.3 + 57.62;
        end
        C_TM_FlightCrew = CostPerBlockHr / (V_B * P);
        % b. Fuel and Oil
        if type_flight == 1
            C_ft = 0.0438;
        elseif type_flight == 2
            C_ft = 0.0625;
        end
        C_ot = 2.15;
        C_TM_FuelOil = ( 1.02*F_b*C_ft + type_engineqty*C_ot*Time_b*0.135 ) / (D*P);
        % c. Hull Insurance Costs
        W_a = W_takeoff * (1-FF) - W_pl - (W_pp*W_takeoff);
        C_a = 2.4e6 + (87.5*W_a);
        C_e = 590000 + (16*Thrust_engine);
        if type_inputs == 2
            C_e = 1314942;
        end
        C_T = (type_engineqty*C_e) + C_a;
        IR_A = 0.01;
        U = 630 + 4000/( 1 + 1/(Time_b + 0.5)); % block hrs/yr
        C_TM_HullInsurance = IR_A * C_T / (U * V_B * P);
        % Direct Maintenance
        % a. Airframe-Labor
        K_FHa = 4.9169 * log10(W_a/1000) - 6.425;
```

```
        K_FCa = 0.21256 * log10(W_a/1000)^3.7375;
        T_F = Time_b - Time_gm;
        R_L = 8.6;
        C_TM_a_AirframeLabor = (K_FHa * T_F + K_FCa) * R_L * (1 + 0.29*(M-1))^1.5 / (V_B * Time_b * P);
        % b. Airframe Material
        C_FHa = ((1.5994*C_a)/1e6) + 3.4263;
        C_FCa = ((1.9229*C_a)/1e6) + 2.2504;
        C_TM_b_AirframeMaterial = (C_FHa*T_F+C_FCa) / (V_B*Time_b*P);
        % c. Engine Labor
        K_FHe = (type_engineqty*(Thrust_engine/1e3))/(0.82715*(Thrust_engine/1e3)+ 13.639);
        K_FCe = 0.2 * type_engineqty;
        C_TM_c_EngineLabor = ((K_FHe*T_F+K_FCe)*R_L) / (V_B*Time_b*P);
        % d. Engine Material
        C_FHe = (((28.2353*C_e)/1e6) - 6.5176) * type_engineqty;
        C_FCe = (((3.6698*C_e)/1e6) + 1.3685) * type_engineqty;
        C_TM_d_EngineMaterial = (C_FHe*T_F+C_FCe) / (V_B*Time_b*P);
        if type_engine == 1
            C_TM_d_EngineMaterial = C_TM_d_EngineMaterial * 1.1;
        end
        % e. Total Maintenance - Burdened
        C_TM_e_Maintenance = (C_TM_a_AirframeLabor + C_TM_b_AirframeMaterial + C_TM_c_EngineLabor +
C_TM_d_EngineMaterial)*2;
        % Depreciation
        D_a = 14;
        C_TM_Depreciation = (1/(V_B*P)) * (C_T + 0.06*(C_T-type_engineqty*C_e)+0.3*type_engineqty*C_e) /
(D_a*U);
        % Total DOC
        DOC_perTonMile = C_TM_FlightCrew + C_TM_FuelOil + C_TM_HullInsurance + C_TM_e_Maintenance +
C_TM_Depreciation;
        DOC_perPAXMile = DOC_perTonMile * P/PAX;

    %% Data Collection
        % Plots
        table_airfoil_AR(i,j) = AR;
        table_airfoil_AirfoilType(i,j) = type_airfoil;
        table_airfoil_DOC(i,j) = DOC_perPAXMile;
        table_airfoil_Weight(i,j) = W_takeoff;

        iteration = iteration + 1
        j = j+1;
    end
    i = i+1
end

figure
hold on
for i = [1:1:type_airfoil]
    plot(table_airfoil_AR(i,:),table_airfoil_DOC(i,:))
end
legend('Conventional','Supercritical')
title(legend,'Airfoil Type')
title('DOC wrt Airfoil Type')
xlabel('Aspect Ratio')
ylabel('DOC ($)')
hold off
```

3D Model

Sketch of Dimensions



Tables

Wing Sizing	
Aspect Ratio	12
Sweep	36
MAC	147.72
b	1773.891
S	262224
Fuselage Length	1902
Fuselage Diameter	157.2
Taper	0.5
Thickness/Chord	0.136
Engine Qty	2
Abreast	6
Aisles	1
C_root	197.099
C_tip	98.54948
MAC	153.2992
y_MAC	394.1979

H Tail Sizing	
AR	3.5
Taper	0.35
Volume Coeff	1.25
Tail Arm	646.9453
S	77670.26
b	521.3884
C_root	220.6935
C_tip	77.24273
Sweep	36
MAC	160.4796

V Tail Sizing	
AR	1.6
Taper	0.5
Volume Coeff	0.079
Tail Arm	646.9453
S	56801.37
b	150.7334
C_root	502.4445
C_tip	251.2223
Sweep	36
MAC	390.7902

Engine Sizing	
Thrust Max	72790
Bypass Ratio	5
Fan Diameter	115
Engine Length	185
Nacelle Diameter	126.5
Nacelle Inlet Length	80.5
Nacelle Total Length	292.05
y_Engine	354.7781
Engine Offset	126.5

Wing Parameters	
t/c	0.1366
Taper	0.5
Root Chord	197.099
Weight of Fuel	81619
Fuel (Gallons)	12163.65
Fuel (in^3)	2810090
Fuel Per Tank	1405045
Tank Height	0.9
Front Spar Location	0.2
Rear Spar Location	0.8
Fuel Tip Chord	102.6545

Rectangular Frustum Parameter		
Base Height	24.23135	
Base Width	118.2594	
A_1	2865.584	
Tip Height	12.62035	
Tip Width	61.59271	
A_2	777.3213	
Length of Tank	850	Difference
Volume of Tank	1455024	-49978.9
Centroid Location	338.5885	(negative is good)
Front Spar from Leading Edge	30.65984	

Fuselage	
Fuselage Length	1902
Fuselage Diameter	157.2
Nose Length	235.8
Afterbody Length	314.4
Wing 1/4 MAC Position	951

W_takeoff	288610	
Component	Coefficient	Weight
Wing	0.0075	25126.5786
Fuselage	1530	29367.3395
Landing Gear	0.04	11544.4
Nacelle + Pylon	0.0224	6464.864
Tails	0.0013	375.193
Power Plant	0.155	44734.55
Fixed Equipment	0.035	10101.35
Fixed Equipment		28200
Fuel	0.2828	81618.908
Passenger		43000
Cargo		4000
Total		284533.1831
(to a reference pt 120in in front of nose)	120	

Configurations	Cargo	Fuel	PAX
Design			
Empty			
Ferry			
Landing			

	Design			Empty			Ferry			Landing		
Component CG	Weight (lbs)	Length (in)	Moment	Weight (lbs)	Length (in)	Moment	Weight (lbs)	Length (in)	Moment	Weight (lbs)	Length (in)	Moment
1/4 MAC (Airplane Desired CG)		1071										
Wing	25126.5786	1109.324798	27873537	25126.5786	1109.324798	27873537	25126.5786	1071	26910566	25126.5786	1109.324798	27873537
Fuselage	29367.3395	1071	31452421	29367.3395	1071	31452421	29367.3395	1071	31452421	29367.3395	1071	31452421
Landing Gear	11544.4	1086.329919	12541027	11544.4	1086.329919	12541027	11544.4	1086.329919	12541027	11544.4	1086.329919	12541027
Nacelle + Pylon	6464.864	963.2997978	6227602	6464.864	963.2997978	6227602	6464.864	963.2997978	6227602	6464.864	963.2997978	6227602
Tails	375.193	1796.390019	673993	375.193	1796.390019	673993	375.193	1796.390019	673993	375.193	1796.390019	673993
Power Plant	44734.55	963.2997978	43092783	44734.55	963.2997978	43092783	44734.55	963.2997978	43092783	44734.55	963.2997978	43092783
Fixed Equipment	10101.35	1191	12030708	10101.35	1191	12030708	10101.35	1191	12030708	10101.35	1191	12030708
Fixed Equipment	28200	1191	33586200	28200	1191	33586200	28200	1191	33586200	28200	1191	33586200
Fuel	81618.908	1027.810789	83888794	0	1027.810789	0	81618.908	1027.810789	83888794	0	1027.810789	0
Passenger	43000	1071	46053000	0	1071	0	0	1071	0	43000	1071	46053000
Cargo	4000	1071	4284000	0	1071	0	0	1071	0	4000	1071	4284000
Sum	284533.1831		3.02E+08	155914.2751		1.67E+08	237533.1831		2.5E+08	202914.2751		2.18E+08
	Delta 1/4 MAC		-10.6524									
			If negative, move stuff back									

CG Design	1060.348	
CG Empty	1074.169	
CG Ferry	1054.186	
CG Landing	1073.435	
Forward	1052.604	
Aft	1086.33	

Landing Gear		Diameter
W_takeoff	288610	
Main Gear per Wheel	36076.25	55.8
Nose Gear per Wheel	21645.75	39.2

Length	Weight	Moment
0	288610	0
-15.3299191	288610	-4424368
153.299191	28861	4424368
		0

Passengers	200		
First Class Seats	16		
Abreast	4		
Rows	4		
LAV	2	38in x 40in	
Galley Carts	4.8	12in x 34in	
Seat Pitch	40in		
Depth/Recline	28in/36in		
Bulkhead to seat nose	24in		
Aisle Width	20in		
Economy Seats	184		186
Abreast	6		
Rows	30.66667		
LAV	3.68		
Galley Carts	14.72		
Seat Pitch	32in		
Depth/Recline	28in/36in		
Bulkhead to seat nose	18in		
Aisle Width	18in		
LDW Volume	84		
Cargo Weight	4000		
Cargo Density	10		
Cargo Volume	400		
Containers Required	4.761905		