AAE 334: Aerodynamics

HW7: Wing Stability Analysis

Dr. Blaisdell

Purdue University

School of Aeronautical and Astronautical Engineering

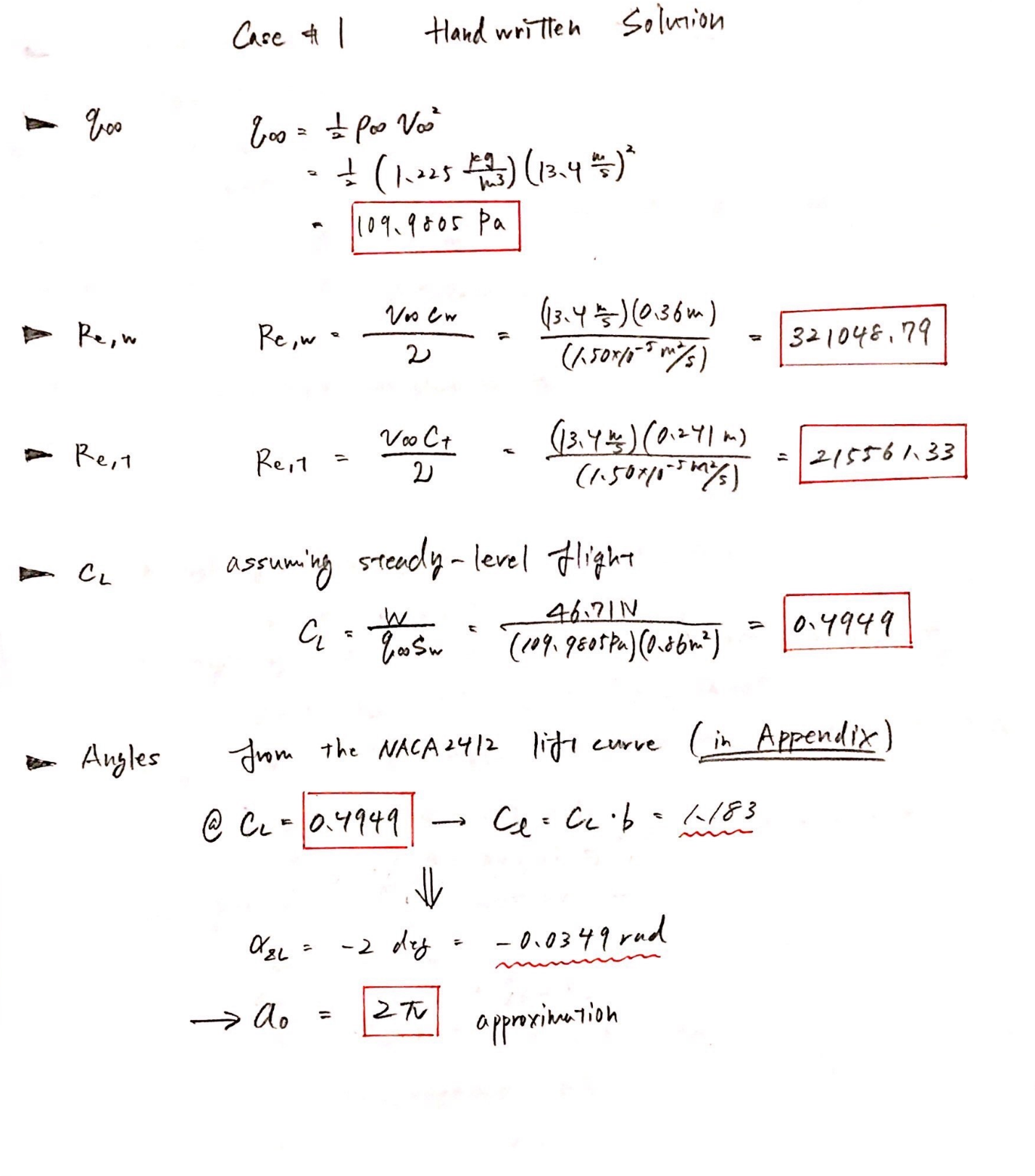
Tomoki Koike

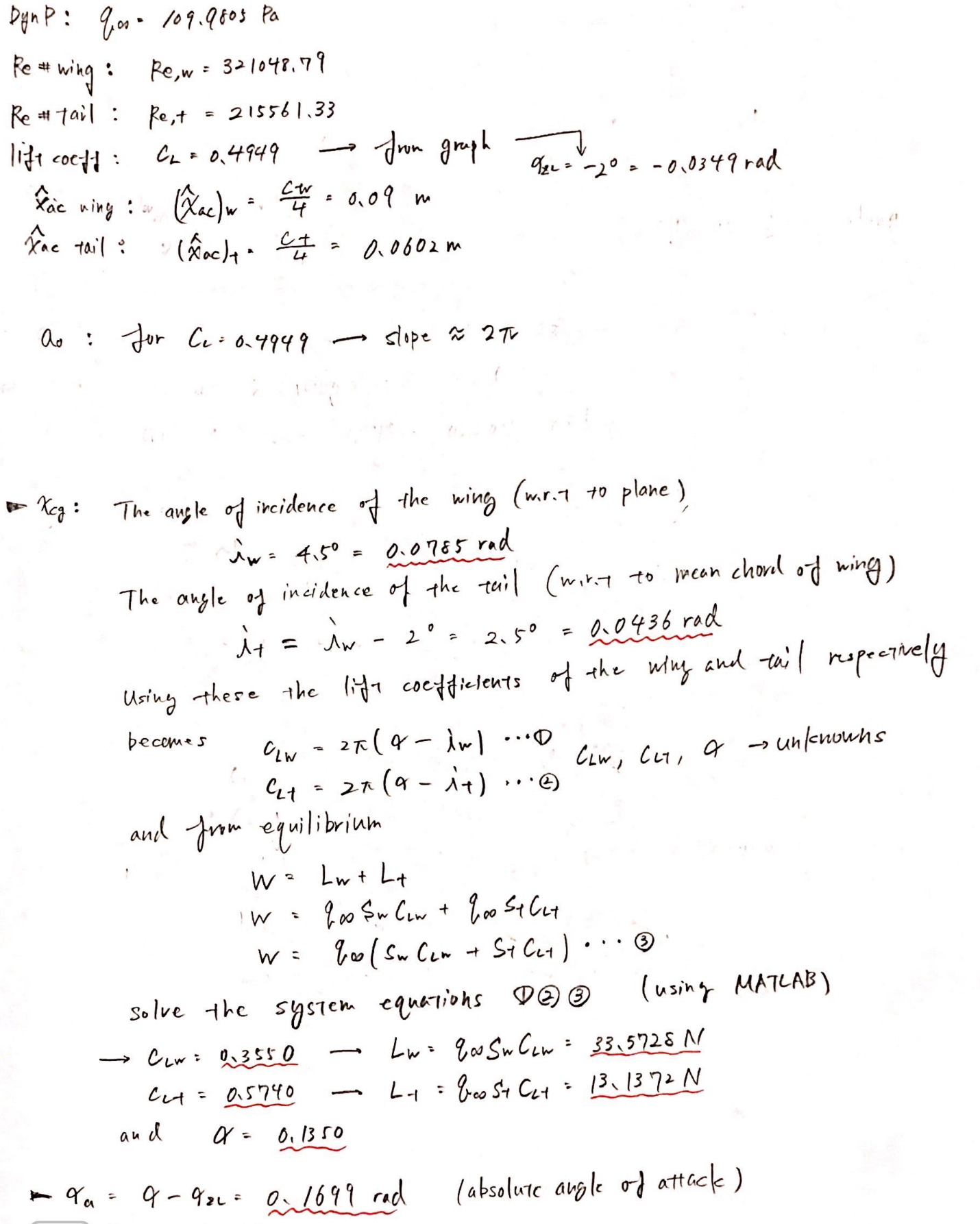
Friday March 13th 2020

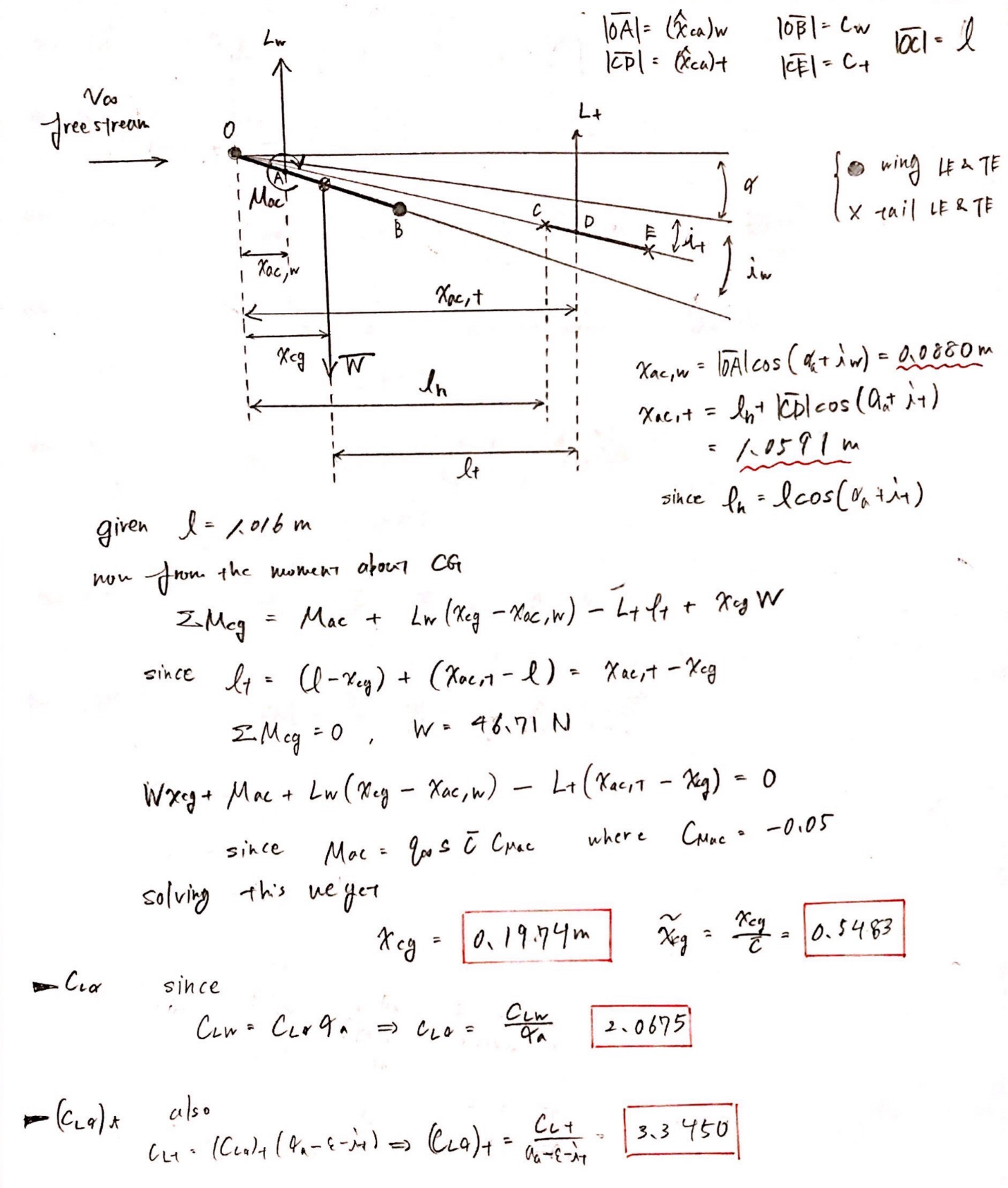
Download the files in the HW7 folder on Blackboard. To review use of XFLR5 for stability analysis see the video ‘XFLR5 Demo for AAE 334 Stability Lecture’ by Professor Sullivan, which is available on the AAE 334 Course Content page on Blackboard. After performing the calculations or running the analyses described below, enter the results in the appropriate column in the Excel file “AAE334 Homework 7 Data and Answer Sheet”. Print a copy of the completed spreadsheet to turn in with your homework.

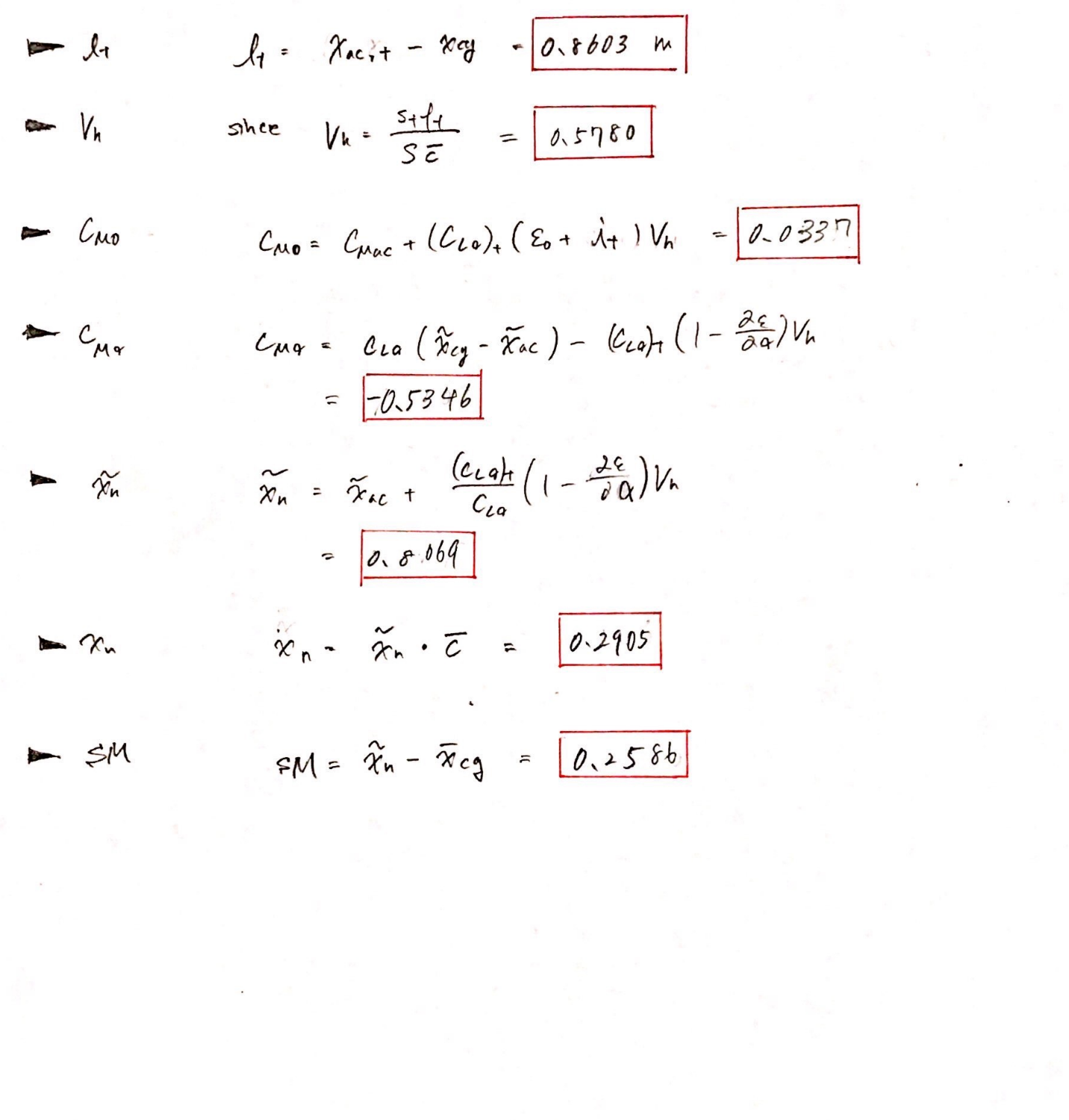
1. Problem #1
   1. [25 pts] Using the formulas in the notes, calculate the stability quantities on the “Homework sheet” tab of the Excel file “AAE334 Homework 7 Data and Answer Sheet”. Show your handwritten work on separate sheets of paper.
   2. [25 pts] Run XFLR5 using the file “AAE 334 telemaster stability Homework #7.xfl” and compare your answers (i.e., calculate the percent difference). (Note that the xfl file already has the geometry of the airplane included. In addition, the analysis of the wing alone has already been done. However, you will have to run the necessary analysis of the airplane.)
2. Problem #2 - Rerun xFLR5 with a velocity V=20 m/sec.
   1. [25 pts] Trim the aircraft by moving the CG so the aircraft is stable at the proper trim point.
   2. [25 pts] Trim the aircraft by changing the tail angle so the aircraft is stable at the proper trim point. (Use the original CG location.)

\*The spreadsheet with the final answers is at the end









PLOT

A close up of a map

Description automatically generated

MATLAB CODE

**AAE334 HW7 MATLAB CODE**

clear all; close all; clc;

fdir = 'C:\Users\Tomo\Desktop\studies\2020-Spring\AAE334\matlab\outputs\HW7';

set(groot, 'defaulttextinterpreter',"latex");

set(groot, 'defaultAxesTickLabelInterpreter',"latex");

set(groot, 'defaultLegendInterpreter',"latex");

**problem 1-a**

% Defining necessary properties

v\_inf = 13.4; % freestream velocity [m/s]

rho\_inf = 1.225; % freestream air density [kg/m3]

q\_inf = 0.5\*rho\_inf\*v\_inf^2; % freestream dynamic pressure [Pa]

S\_w = 0.86; % wing area [m2]

S\_t = 0.208; % tail area [m2]

ch\_w = 0.36; % chord length of wing [m]

ch\_t = 0.241; % chord length of tail [m]

l = 1.016; % distance from the LE of the wing to the LE of the tail [m]

ch\_mean = ch\_w; % mean chord length of the wing [m]

epsilon = 0;

e\_a = 0.4; % D(epsilon)/Da

W = 46.71; % weight of the plane [N]

% Calculating the x\_cg (center of gravity of the plane)

% Necessary angles [rad]

a\_zl = deg2rad(-2.1) % zero lift angle

i\_w = deg2rad(4.5) % incident angle of the wing

i\_t = i\_w - deg2rad(2) % incident angle of the tail wrt the plane

% find a, C\_Lw, and C\_Lt

[C\_Lw, C\_Lt, a] = CLw\_CLt\_AoA(W, i\_w, i\_t, q\_inf, S\_w, S\_t)

a\_a = a - a\_zl % absolute angle of attack

% the lifts for the wing and tail

L\_w = q\_inf\*S\_w\*C\_Lw % wing [N]

L\_t = q\_inf\*S\_t\*C\_Lt % tail [N]

% aerodynamic center positions (wrt the LE of the wing)

xhat\_ac\_w = ch\_w/4 % wing [m]

xhat\_ac\_t = ch\_t/4 % tail [m]

lh = l\*cos(a\_a + i\_t);

x\_ac\_w = xhat\_ac\_w\*cos(a\_a + i\_w)

x\_ac\_t = lh + xhat\_ac\_t\*cos(a\_a + i\_t)

x\_ac\_til = 0.25;

% moment coefficient about the aerodynamic center of the wing

C\_Mac\_w = -0.05;

% moment about the aerodynamic center of the wing [N-m]

Mac\_w = q\_inf\*S\_w\*ch\_mean\*C\_Mac\_w

% solve

syms var1

eqn1 = Mac\_w + L\_w\*(var1 - x\_ac\_w) - L\_t\*(x\_ac\_t - var1) + W\*var1 == 0;

x\_cg = double(solve(eqn1, var1))

x\_cg\_til = x\_cg/ch\_mean

% C\_La

C\_La\_w = C\_Lw/a\_a

% C\_La\_t

C\_La\_t = C\_Lt/(a\_a)

% l\_t distance from the x\_cg to the aerodynamic center of the tail

l\_t = x\_ac\_t - x\_cg

% Vh

Vh = S\_t\*l\_t/S\_w/ch\_mean

% C\_Mo

C\_Mo = C\_Mac\_w + C\_La\_t\*(epsilon + i\_t)\*Vh

% C\_Ma

C\_Ma = C\_La\_w\*(x\_cg\_til - x\_ac\_til) - C\_La\_t\*(1 - e\_a)\*Vh

% x\_n

x\_n\_til = x\_ac\_til + C\_La\_t/C\_La\_w\*(1 - e\_a)\*Vh

x\_n = x\_n\_til\*ch\_mean

% SM

SM = x\_n\_til - x\_cg\_til

**problem 1-b**

% importing data

data\_1b = readmatrix("inputs\hw7\hw7\_Cm\_vs\_Cl\_case1\_2\_2.csv");

% Calculating the static margin

SM\_1b = calc\_SM\_from\_xflr(data\_1b)

% Importing another data

data\_1b2 = readmatrix("inputs\hw7\hw7\_Cm\_vs\_alpha\_case1\_2.csv");

% Calculating the C\_M0 (\*for the angle of attack add 2.1 degrees to adjust

% to the absolute angle of attack)

[C\_Mo\_1b, C\_Ma\_1b] = calc\_CMo\_and\_CMa\_from\_xlfr(data\_1b2, -2.1, rad2deg(i\_w))

**problem 2-a**

% importing data

data\_2a = readmatrix("inputs\hw7\hw7\_Cm\_vs\_Cl\_case2a.csv");

% Calculating the static margin

SM\_2a = calc\_SM\_from\_xflr(data\_2a)

% Importing another data

data\_2a2 = readmatrix("inputs\hw7\hw7\_Cm\_vs\_alpha\_case2a.csv");

% Calculating the C\_M0 (\*for the angle of attack add 2.1 degrees to adjust

% to the absolute angle of attack)

[C\_Mo\_2a, C\_Ma\_2a] = calc\_CMo\_and\_CMa\_from\_xlfr(data\_2a2, -2.1, rad2deg(i\_w))

% find a, C\_Lw, and C\_Lt

q\_inf\_2a = 245;

[C\_Lw\_2a, C\_Lt\_2a, a\_2a] = CLw\_CLt\_AoA(W, i\_w, i\_t, q\_inf\_2a, S\_w, S\_t)

a\_a\_2a = a\_2a - a\_zl % absolute angle of attack

% the lifts for the wing and tail

L\_w = q\_inf\_2a\*S\_w\*C\_Lw\_2a % wing [N]

L\_t = q\_inf\_2a\*S\_t\*C\_Lt\_2a % tail [N]

**problem 2-b**

% importing data

data\_2b = readmatrix("inputs\hw7\hw7\_Cm\_vs\_Cl\_case2b.csv");

% Calculating the static margin

SM\_2b = calc\_SM\_from\_xflr(data\_2b)

% Importing another data

data\_2b2 = readmatrix("inputs\hw7\hw7\_Cm\_vs\_alpha\_case2b.csv");

% Calculating the C\_M0 (\*for the angle of attack add 2.1 degrees to adjust

% to the absolute angle of attack)

[C\_Mo\_2b, C\_Ma\_2b] = calc\_CMo\_and\_CMa\_from\_xlfr(data\_2b2, -2.1, rad2deg(i\_w))

% find a, C\_Lw, and C\_Lt

q\_inf\_2b = 245;

[C\_Lw\_2b, C\_Lt\_2b, a\_2b] = CLw\_CLt\_AoA(W, i\_w, i\_t, q\_inf\_2b, S\_w, S\_t)

a\_a\_2b = a\_2b - a\_zl % absolute angle of attack

% the lifts for the wing and tail

L\_w = q\_inf\_2b\*S\_w\*C\_Lw\_2b % wing [N]

L\_t = q\_inf\_2b\*S\_t\*C\_Lt\_2b % tail [N]

**Plotting**

fig1 = figure("Renderer","painters");

plot(data\_1b(:,1),data\_1b(:,2))

title({'$C\_m$ vs $C\_L$ Plots for All Cases - Tomoki Koike'})

xlabel('$C\_L$')

ylabel('$C\_m$')

hold on

plot(data\_2a(:,1),data\_2a(:,2))

plot(data\_2b(:,1),data\_2b(:,2))

plot(linspace(-0.2,0.7,100),zeros([1,length(linspace(-0.2,0.7,100))]),'--k',"LineWidth",1.0)

plot(zeros([1,length(linspace(-0.15,0.15,100))]),linspace(-0.15,0.15,100),'--k',"LineWidth",1.0)

hold off

grid on; grid minor; box on;

legend('Case 1','Case 2a','Case 2b')

saveas(fig1, fullfile(fdir,'Cm\_vs\_Cl.png'))

**FUNCTIONS**

function SM = calc\_SM\_from\_xflr(data)

% data: data points

CL\_1b = data(:,1);

Cm\_1b = data(:,2);

% SM

if Cm\_1b(CL\_1b == min(abs(CL\_1b)))

Cm\_1b\_0 = Cm\_1b(CL\_1b == min(abs(CL\_1b)));

else

Cm\_1b\_0 = Cm\_1b(CL\_1b == -(min(abs(CL\_1b))));

end

if CL\_1b(Cm\_1b == -(min(abs(Cm\_1b))))

CL\_1b\_0 = CL\_1b(Cm\_1b == -(min(abs(Cm\_1b))));

else

CL\_1b\_0 = CL\_1b(Cm\_1b == (min(abs(Cm\_1b))));

end

SM = Cm\_1b\_0/CL\_1b\_0;

end

function [C\_Mo, C\_Ma] = calc\_CMo\_and\_CMa\_from\_xlfr(data, a\_zl, i\_w)

% data: data points

% a\_zl: zero lift angle of attack

% i\_w: wing incidence angle

Cm = data(:,2);

a = data(:,1) - a\_zl + i\_w;

P = polyfit(a,Cm,1);

C\_Ma = P(1);

C\_Mo = P(2);

end

function [C\_Lw, C\_Lt, a] = CLw\_CLt\_AoA(W, i\_w, i\_t, q\_inf, S\_w, S\_t)

% system equations to find a, C\_Lw, and C\_Lt

syms x1 x2 x3

eqn1 = x1 == 2\*pi\*(x3 - i\_w);

eqn2 = x2 == 2\*pi\*(x3 - i\_t);

eqn3 = W == q\_inf\*(S\_w\*x1 + S\_t\*x2);

res = solve([eqn1 eqn2 eqn3],[x1 x2 x3]);

% the lift coefficients for the wing and tail respectively

C\_Lw = double(res.x1);

C\_Lt = double(res.x2);

% angle of attack

a = double(res.x3);

end