ASEN 3802 Lab 3 Part 1

Andrew Patella, Niko Pappas, Dane Shedd, Lorien Hoshall

April 21, 2025

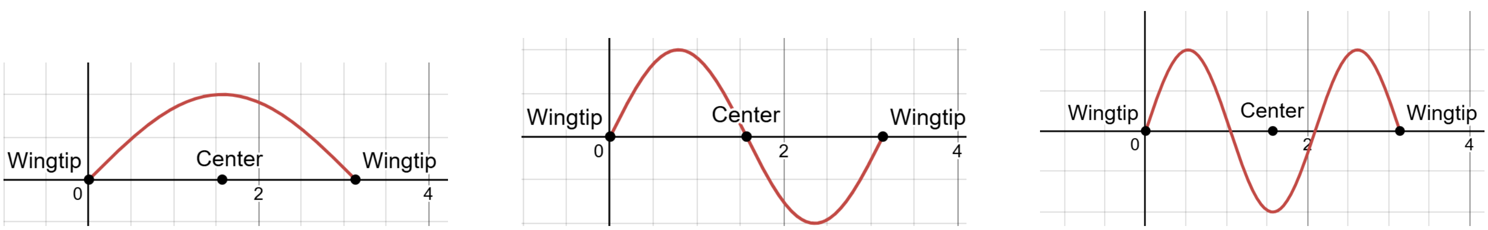
1 Task 1: Creating a PLLT Function

```
function [e,c_L,c_Di] = PLLT(b,a0_t,a0_r,c_t,c_r,aero_t,aero_r,geo_t,geo_r,N)

% INPUTS:
% b - span
% a0_t - lift slope at the wing tip
% a0_r - lift slope at the wing root
% c_t - chord length at the tip
% c_r - chord length at root
% aero_t - induced AoA at wing tip
% aero_r - induced AoA at wing root
% geo_t - geometric AoA at wing tip
% geo_r - geometric AoA at wing root
% N - number of Fourier Coefficients (terms)


% OUTPUTS:
% e - span efficiency factor
% c_L - coefficient of lift
% c_Di - induced drag coefficient


%% Calculating the geometric parameters as functions to use later
S = 2*(0.5*(c_t + c_r)*b/2);
AR = b^2/S;
c = @(y) (c_t-c_r)/(-b/2)*y + c_r; % Assuming linear change
a0 = @(y) (a0_t-a0_r)/(-b/2)*y + a0_r;
aero = @(y) (aero_t-aero_r)/(-b/2)*y + aero_r;
geo = @(y) (geo_t-geo_r)/(-b/2)*y + geo_r;

% Control points
theta0 = zeros(N,1);
for i = 1:N
    theta0(i) = i*pi/(2*N);
end

y0 = -b/2*cos(theta0);


%% Solving system of equations to find the Fourier Coefficients
M = zeros(N,N); %Matrix for linear system of equations

% Assigning the entries of B
for i = 1:N
```

```
    for j = 1:N
        M(i,j) = 4*b/(a0(y0(i))*c(y0(i)))*sin((2*j-1)*theta0(i)) + (2*j-1)*sin((2*j-1)*theta0(i))/
            sin(theta0(i));
    end
end

% Assigning the entries of b
rhs = zeros(N,1);
for i = 1:N
    rhs(i) = -aero(y0(i)) + geo(y0(i));
end

% Solving for the Fourier Coefficients
%A = M\rhs;
A = inv(M)*rhs;
% A1 should be 0.01261

%% Solving for the desired quantities
c_L = A(1)*pi*AR;
delta = 0;

for i = 2:N
    delta = (2*i-1)*(A(i)/A(1))^2 + delta;
end
e = 1/(1+delta);
c_Di = c_L^2/(pi*e*AR);

end
```

Note that in this lab, only the odd terms are utilized in the series expansion for circulation in Prandtl's Lifting Line Theory. Why is this the case? When would both the odd and even terms be required?

Since the domain we are working with is $\theta_0 \in [0, \pi]$, we want to use only the odd terms to create a symmetric lift distribution. When n is odd, $\sin(n\theta_0)$ is symmetric in $[0, \pi]$ across $\theta_0 = \pi/2$. When n is even, $\sin(n\theta_0)$ is asymmetric across $\theta_0 = \pi/2$. We want to neglect these terms because we are assuming for a steady, level flight, there will be no asymmetric lift distribution in order to have no moment about the aircraft's body x axis. If we were modeling an aircraft in a banked turn or analyzing the lateral dynamics of an aircraft, or even a nonuniform lift distribution due to unsteady conditions, we would want to use the odd and even terms in order to capture this behavior in the Fourier Series.



(a) $n = 1$ (b) $n = 2$ (c) $n = 3$

Figure 1: Symmetry across $\theta_0 = \pi/2$ as n increases

2 Task 2: Creating Figure 5.20 from Anderson

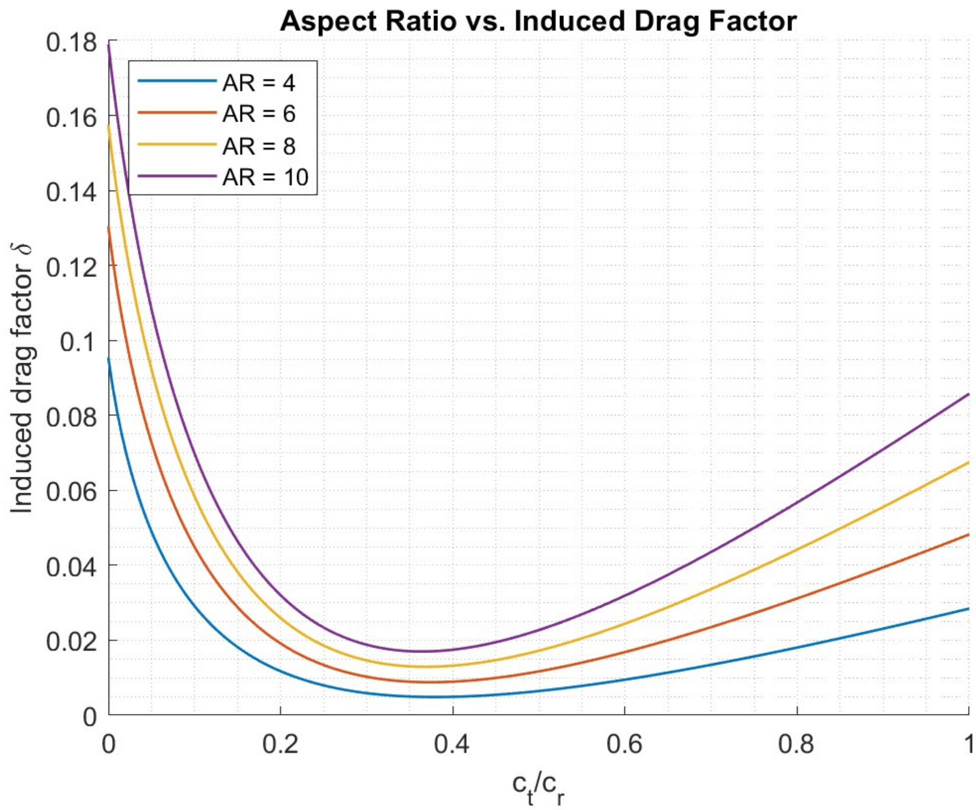


Figure 2: Aspect Ratio and Taper Ratio influence on δ

What does your reproduction of Figure 5.20 imply are the most efficient tapered wing planform geometries? Consider both the impact of taper ratio, c_t/c_r, and Aspect Ratio, AR, in your discussion.
This graph suggests the ideal wing would have a taper ratio of $c_t/c_r \approx 0.3$ and an aspect ratio of $AR = 4$. This minimizes δ, which should then theoretically minimize induced drag and increase spanwise efficiency. However, when fully considering the equations that model lift and drag, this is not the ideal value for AR. To actually pick the best value of AR, we need to look at these equations.

$$C_{Di} = \frac{C_L^2}{\pi e AR} = \frac{C_L^2}{\pi AR}(1 + \delta) \tag{1}$$

Looking at equation 1, by minimizing AR to minimize δ, we can decrease the induced drag by at most 0.08 (when c_t/c_r is not optimized, it is about 0.01 when it is). But by minimizing the aspect ratio, we are dividing by a smaller number. For the aspect values considered, we could use an $AR = 10$ instead of $AR = 4$ to divide drag by 10 instead of 4. So we can minimize δ with $AR = 4$, but we can minimize C_{Di} with a maximized AR.

3 Appendix

3.1 PLLT Function

```
    function [e,c_L,c_Di] = PLLT(b,a0_t,a0_r,c_t,c_r,aero_t,aero_r,geo_t,geo_r,N)

% INPUTS:
% b - span
% a0_t - lift slope at the wing tip
% a0_r - lift slope at the wing root
% c_t - chord length at the tip
% c_r - chord length at root
% aero_t - induced AoA at wing tip
% aero_r - induced AoA at wing root
% geo_t - geometric AoA at wing tip
% geo_r - geometric AoA at wing root
% N - number of Fourier Coefficients (terms)


% OUTPUTS:
% e - span efficiency factor
% c_L - coefficient of lift
% c_Di - induced drag coefficient


%% Calculating the geometric parameters as functions to use later
S = 2*(0.5*(c_t + c_r)*b/2);
AR = b^2/S;
c = @(y) (c_t-c_r)/(-b/2)*y + c_r; % Assuming linear change
a0 = @(y) (a0_t-a0_r)/(-b/2)*y + a0_r;
aero = @(y) (aero_t-aero_r)/(-b/2)*y + aero_r;
geo = @(y) (geo_t-geo_r)/(-b/2)*y + geo_r;

% Control points
theta0 = zeros(N,1);
for i = 1:N
    theta0(i) = i*pi/(2*N);
end

y0 = -b/2*cos(theta0);


%% Solving system of equations to find the Fourier Coefficients
M = zeros(N,N); %Matrix for linear system of equations

% Assigning the entries of B
for i = 1:N
    for j = 1:N
        M(i,j) = 4*b/(a0(y0(i))*c(y0(i)))*sin((2*j-1)*theta0(i)) + (2*j-1)*sin((2*j-1)*theta0(i))/
            sin(theta0(i));
    end
end

% Assigning the entries of b
rhs = zeros(N,1);
for i = 1:N
    rhs(i) = -aero(y0(i)) + geo(y0(i));
end
```

```

% Solving for the Fourier Coefficients
%A = M\rhs;
A = inv(M)*rhs;
% A1 should be 0.01261

%% Solving for the desired quantities
c_L = A(1)*pi*AR;
delta = 0;

for i = 2:N
    delta = (2*i-1)*(A(i)/A(1))^2 + delta;
end
e = 1/(1+delta);
c_Di = c_L^2/(pi*e*AR);

end
```

3.2 Main code

```
 close all; clear; clc;

%% Constants

%
b = 40;
a0_t = 2*pi;
a0_r = 2*pi;
c_t = 5;
c_r = 5;
aero_t = 0;
aero_r = 0;
geo_t = 5*pi/180;
geo_r = 5*pi/180;

N = 2;

[e,c_L,c_Di] = PLLT(b,a0_t,a0_r,c_t,c_r,aero_t,aero_r,geo_t,geo_r,N);

%% PART 2
close all; clear; clc;
a0_t = 2*pi;
a0_r = 2*pi;
c_r = 15;
aero_t = 0;
aero_r = 0;
geo_t = 2*pi/180;
geo_r = 2*pi/180;
N = 10;

ctcr = linspace(0,1,200);
AR = [4,6,8,10];


delta = zeros(length(ctcr),length(AR));
for j = 1:length(AR)
    for i = 1:length(ctcr)
```

```
        b = 0.5*AR(j)*(c_r*ctcr(i)+c_r);
        %[e(i,j),~,~,delta(i,j)] = PLLT2(b,a0_t,a0_r,c_r*ctcr(i),c_r,aero_t,aero_r,geo_t,geo_r,N);
        [e(i,j),~,~] = PLLT(b,a0_t,a0_r,c_r*ctcr(i),c_r,aero_t,aero_r,geo_t,geo_r,N);
        delta(i,j) = 1/(e(i,j)) - 1;
    end
end

figure()
hold on
for j = 1:length(AR)
    plot(ctcr,delta(:,j),'Linewidth',1)
end
grid minor
legend("AR = 4","AR = 6","AR = 8","AR = 10",'location','northwest')
xlabel("c_t/c_r")
ylabel("Induced drag factor \delta")
ylim([0,0.18])
title("Aspect Ratio vs. Induced Drag Factor")

figure()
hold on
for j = 1:length(AR)
    plot(ctcr,e(:,j),'Linewidth',1)
end
grid minor
legend("AR = 4","AR = 6","AR = 8","AR = 10",'location','northwest')
xlabel("c_t/c_r")
ylabel("e")
title("Aspect Ratio influence on Spanwise Efficiency ")
```