AE 678 – Aeroelasticity

Assignment 2

Static Aeroelastic Analysis of XB-47 Bomber with Swept wing

BY

Name: Rishi Raj

Roll Number: 22M0033

Program: M.Tech (TA)

Specialization: Aerospace Structures

Structure case – I

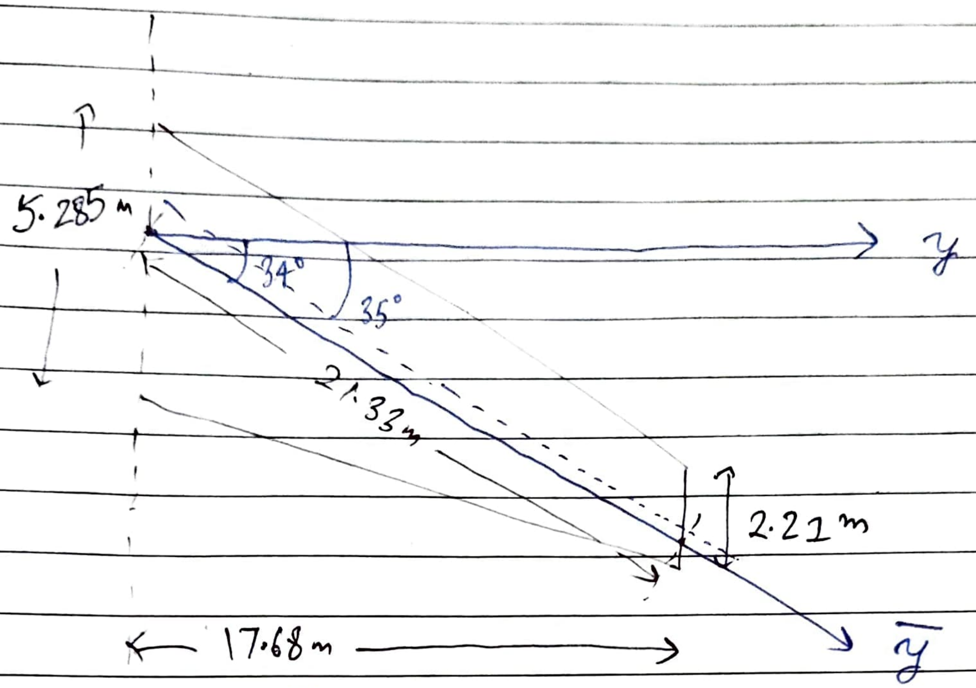


Fig-1:

Mach no = 0.8 , line of CG = 0.35c

altitude = 6000m

CL (2d-uncorrected) = 6.28 per radian

Cmac = −0.015

3g-pullout , W = 66000 kg , ~~A.R = 9.43~~

@ 6000m altitude:

density = 0.54 kg/m3

Temperature = 251.7K

$\therefore P_{dyn} = \frac{1}{2} \rho_\infty u_\infty^2$

$= \cancel{\frac{1}{2} \times 0.54 \times (0.8 \times 330)^2}$

$u_\infty = \sqrt{\gamma R T} = \sqrt{1.4 \times \cancel{8.314} \times 251.7}$ 287

Page No.:
Date: / /

Ref: $U_\infty = 318\ m/sec$.

$q_{dyn} = \frac{1}{2} \times 0.54 \times (318)^2 = 27305.93\ N/m^2$

Variation of stiffness, geometric and mass distribution along $\bar{y}$ direction.

1. Variation of EI:

$$EI(y) = EI_{root} - \left(\frac{-EI_{tip} + EI_{root}}{21.33\ m}\right)\bar{y}$$

$$= 4\times10^8 - \frac{3.8}{21.33}\bar{y} \times 10^8$$

$$= (4 - 0.178\,\bar{y})\times10^8\ Nm^2$$

2. Variation of GJ:

$$GJ(y) = GJ_{root} - \left(\frac{-GJ_{tip} + GJ_{root}}{21.33}\right)\bar{y}$$

$$= (2.5 - 0.109\,\bar{y})\times10^8\ Nm^2$$

3. Variation of mass:

$$M = m_{root} - \left(\frac{m_{root} - m_{tip}}{21.33}\right)\bar{y}$$

$$m = 800 - \left(\frac{800 - 200}{21.33}\right)\bar{y}$$

$$m(y) = (800 - 28.13\,\bar{y})\ kg/m$$

4. <u>Variation of chord</u>:

$$C(y) = C_{root} - \left(\frac{C_{root} - C_{tip}}{21.33}\right)\bar{y}$$

$$= 5.285 - \left(\frac{5.285 - 2.21}{21.33}\right)\bar{y}$$

$$= (5.285 - 0.1442\,\bar{y})\ m$$

5. <u>Variation of ec</u>: (w.r.t reference axis)

$$ec(y) = e \times C(y)$$

Now, <u>calculation of e</u>:

$$(ec)_{root} = 2.008 - \frac{5.285}{4} = 0.6875$$

$$e = \frac{0.6875}{C_{root}} = \frac{0.6875}{5.285} = 0.13$$

$$\therefore\ ec(y) = 0.6875 - 0.0185\,\bar{y}$$

PANKAJ
Page No.:
Date : / /

6. Variation of CG line (w.r.t reference axis)

$y_{cg}(y) = 0.35\,c(y)$
$= 0.35(5.285 - 0.1442\,\bar{y})$

$y_{cg}(y) = 2.008 - 0.35\,c(y)$
$= 2.008 - (1.85 - 0.05\,\bar{y})$
$= 0.1013\,\bar{y} - 0.158$

~~Lift slope curve correction:~~

~~Using Oswald correction factor:~~

$$\cancel{e = \frac{1}{\left[1 + \left(\frac{b^2}{a^2} \times AR \times C_{L\alpha}\right)\right]}}$$

$$\cancel{= \frac{1}{\left[1 + \frac{(35.35)^2}{\left(\frac{5.285 + 2.21}{2}\right)^2} \times 9.43 \times 6.28\right]}}$$

$$\cancel{e = \frac{1}{1 + \left(\frac{b^2}{a^2 AR C_{L\alpha}}\right)}}$$

Correction for lift slope curve:

$$C_{L\alpha\,corr} = \frac{\pi AR \cdot B}{\left[\left(\sqrt{1+B^2}\right)+B\right]} \quad ; \quad B = \frac{C_{L\alpha=0} \times \cos\Omega}{\pi AR\sqrt{1-M^2\cos^2\Omega}}$$

$$B = \frac{6.28 \times \cos 34}{\pi \times 9.43 \times \sqrt{1-0.64 \times \cos^2 34}} = 0.21$$

$$C_{L\alpha\,corrected} = \frac{\pi \times 9.43 \times 0.21}{\left(\sqrt{1+(.21)^2}+.21\right)} = 5.05$$ per radian.

F1 formulation:

From the figure -1,

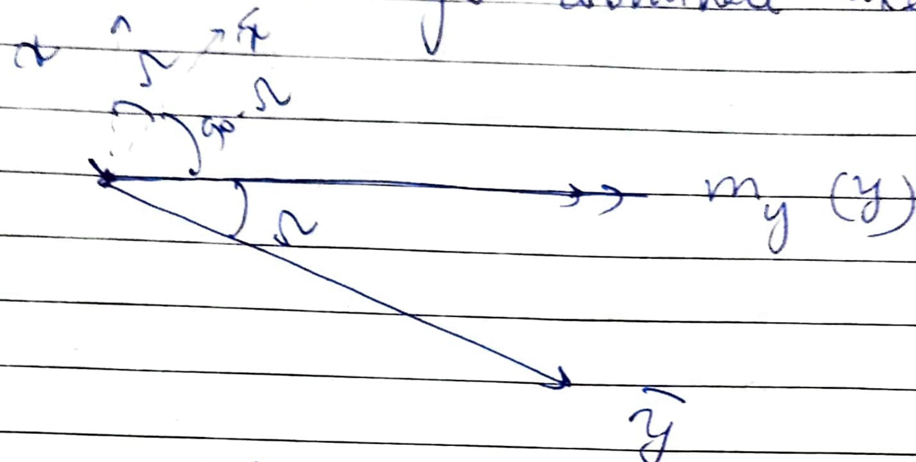
$$l(y) = \frac{1}{2}\rho V^2 \times c(y) \times C_{L\alpha} \times \alpha.$$

$$; \alpha = \alpha_{rigid} + \theta(y) - \underbrace{\phi\frac{\partial\phi}{\partial y}}_{\text{due to bending.}}$$

force per unit length $P(y) = \ell(y) - N \cdot m(y) \cdot g$

$m_y(y) = \ell(y)\,(ec)\,y + \rho_{air}\,[c(y)]^2\, C_{mac} - N\, m(y)\, g \times y_{cg}(y) \cdot c(y)$

Transforming running lift and running moment in the $\bar{y}$ coordinate axes.



i.e $m_y(y) \rightarrow m_{\bar{y}}(\bar{y})$

$\rightarrow m_{\bar{x}}(\bar{y})$

$P(y) \rightarrow P(\bar{y})$

is the transformation required.

So we need two transformation,

i.e $m_{\bar{y}}(\bar{y}) = m_y(y) \cos\Omega \times \sin\Omega$ — (1)

$m_{\bar{x}}(\bar{y}) = m_y(y) \cos\Omega \cos\Omega$ — (2)

for $P(y) \rightarrow P(\bar{y})$

only one transformation is required,

$P(\bar{y}) = P(y) \cos\Omega$ — (3)

writing down governing differential equations for Bending and torsion respectively. ~~($\bar{y}$ is referred to as y here)~~

$$R_1 : \frac{d^2}{d\bar{y}^2}\left(EI(\bar{y})\frac{d^2 w(\bar{y})}{d\bar{y}^2}\right) + \frac{d^2}{d\bar{y}^2}\left(m_{\bar{y}}(\bar{y})\right) + p(\bar{y}) = 0 \quad \text{--- (4)}$$

&

$$R_2 : \frac{d}{d\bar{y}}\left(GJ(\bar{y})\frac{d\theta}{d\bar{y}}\right) + m_x(\bar{y}) = 0 \quad \text{--- (5)}$$

Solution → S2B : Weighted error minimization → least squares with integrated error.

→ R_1 & R_2 are the two residues after ~~putting~~ substituting approximate solutions.

→ $\phi(\bar{y}) = a_1 + a_2\bar{y} + a_3\bar{y}^2 + \dots + a_n\bar{y}^{n-1}$

$\psi(\bar{y}) = b_1 + b_2\bar{y} + b_3\bar{y}^2 + \dots + b_m\bar{y}^{m-1}$

BCs for flexible wing case :

$w(0) = 0$, $w'(0) = 0$ → slope @ root.

deflection @ root

Bending moment @ tip : $EI\frac{d^2 w(21.33)}{dy^2} = 0$

Page No.:
Date: / /

Shear force @ tip : $\frac{d}{d\bar{y}}\left(EI \frac{d^2 w(21\cdot33)}{dy^2}\right) = 0$

For torsion case:

twist @ root = $\bar{\sigma}_y$ $\theta(0) = 0$

torque @ tip = $GJ \frac{d\theta(21\cdot33)}{d\bar{y}} = 0$

Applying B.C on the trial function: (generalized)

$w(0) = 0$

$\Rightarrow a_1 = 0$

$w'(0) = 0 \Rightarrow a_2 = 0$

$EI \frac{d^2 w(L)}{d\bar{y}^2} = 0$

$\Rightarrow EI(y) \frac{d^2 w(L)}{d\bar{y}^2} = 2a_3 + 6a_4 \cdot L$ —— (6)

$~~+ 12a_5 \ldots~~$

$+ 12a_5 L^2 + \cdots$

$= 0$

$EI \frac{d^3 w(L)}{d\bar{y}^3} = 6a_4 + 24a_5 L + \cdots$ —— (7)

$= 0$

~~from ⑦; $6a_4 L + 24 a_5 L$~~

$$w(\bar{y}) = \sum_{i=1}^{N} a_i \phi_i$$

here ϕ_1 will be first trial function for $w(\bar{y})$;

taking 4 terms, for ϕ_1

$$\phi_1 = a_1 \bar{y} + a_2 \bar{y} + a_3 \bar{y}^2 + a_4 \bar{y}^3 + a_5$$

Now applying BC on this,

$w(0) = 0 \Rightarrow a_1 = 0$

$w'(0) = 0 \Rightarrow a_2 = 0$

$EI(y)\, w''(0) = 0 \Rightarrow 2a_3 + 6a_4 L + 12 a_5 L^2$

$EI(y)\quad w'''(0) = 0 \Rightarrow 6a_4 + 24 a_5 L = 0$

Date : / /

we have,

$$a_3 + 3a_4 L + 6a_5 L^2 = 0$$

$$a_4 + 4a_5 L = 0$$

$\Rightarrow$ $a_3 = -12a_4 L$.

$$a_4 = -4a_5 L$$

$$a_3 = 6a_5 L^2$$

$$\therefore \phi_1 = (6a_5 L^2)\bar{y}^2 + -(4a_5 L)\bar{y}^3 + a_5 L^4$$

$$\phi_1 = a_5(6L^2\bar{y}^2 - 4L\bar{y}^3 + L^4)$$

Now for choosing other ϕ_i's we can choose different powers of $\bar{y}$, as we can simply use 'ϕ' for various i's.

This ensures different powers of $\bar{y}$ for ϕ_i's, $w\bar{y} \leq O(\bar{y})$

$\therefore$ Similarly applying BC to the second solution (ψ_i);

$$\psi_1 = b_1 + b_2\bar{y} + b_3\bar{y}^2$$

This is the first term for $O\bar{y})$

$\theta(0) = 0$

$\Rightarrow \psi_1(0) = 0$

$\Rightarrow b_1 = 0$

$\theta'(L) = 0$

$\Rightarrow \psi_1'(L) = 0$

$\Rightarrow b_2 + 2b_3 L = 0$

$\Rightarrow b_2 = -2b_3 L$

$\therefore$ ~~$\psi_1 = -3b_3 L^2 \bar{y} + b_3 \bar{y}$~~

$\psi_1 = (-2b_3 L)\bar{y} + b_3 \bar{y}^2$

$= (-2L\bar{y} + \bar{y}^2) b_3$

Simillarly,

other $\psi_i = \psi_1^i$

& $\phi_i = \phi_1^i$

$\therefore \omega(y) = a_1 \phi_1 + a_2 \phi_1^2 + a_3 \phi_1^3 + \ldots$

& $\theta(y) = b_1 \psi_1 + a_2 \psi_1^2 + a_3 \psi_1^3 + \ldots$

a_i's and b_i's are redefined from here on

where, $\phi_1 = 6L^2\bar{y}^2 + L^4 - 4L\bar{y}^3$

& $\psi_1 = -2L\bar{y} + \bar{y}^2$

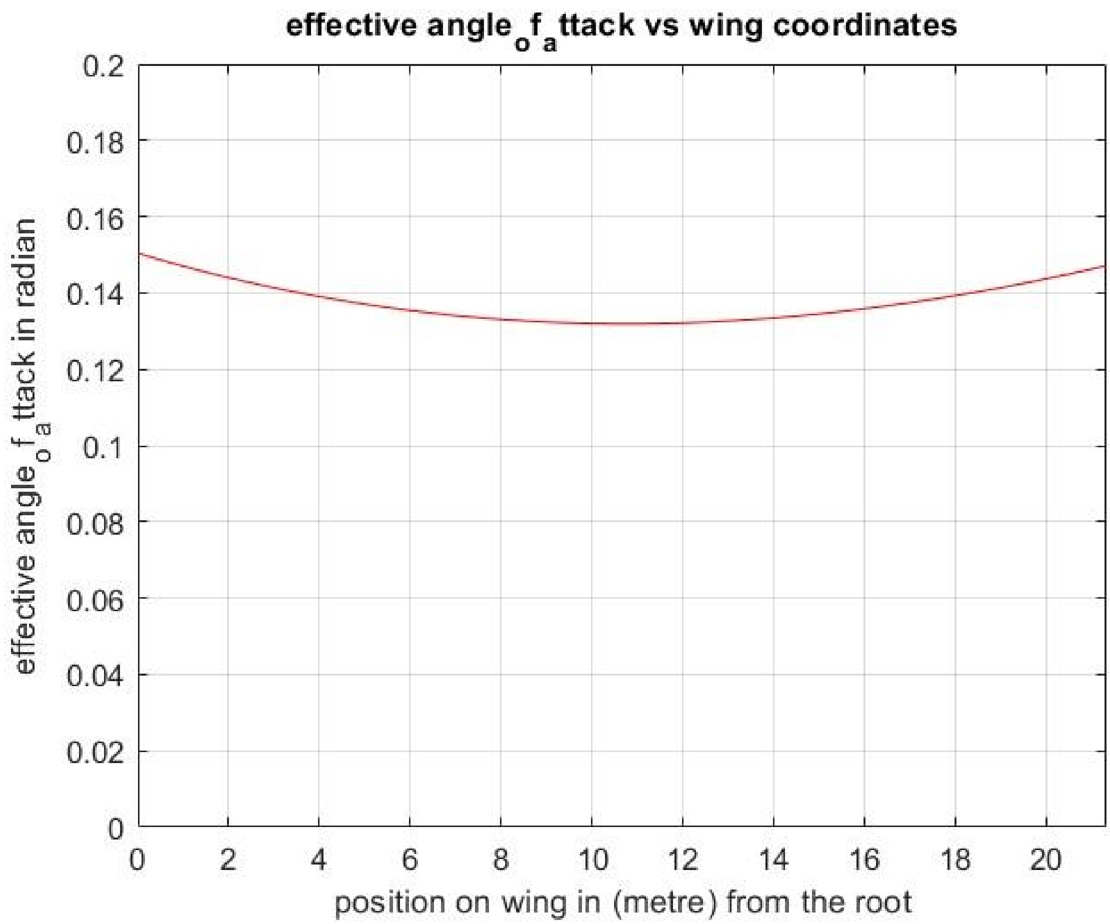
As per solution method,

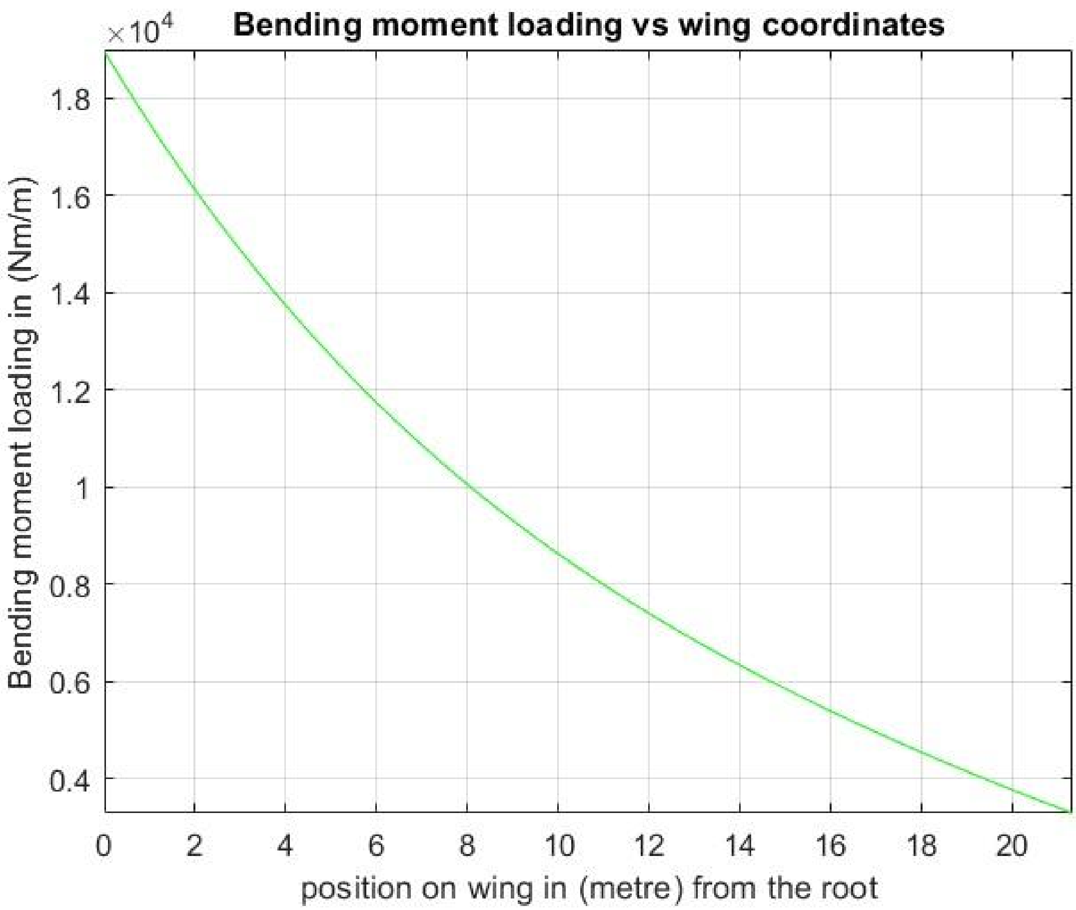
$$\frac{d}{da_i}\int R_1^2 \, d\Omega = 0 \quad \& \quad \frac{d}{db_i}\int R_2^2 \, d\Omega = 0$$

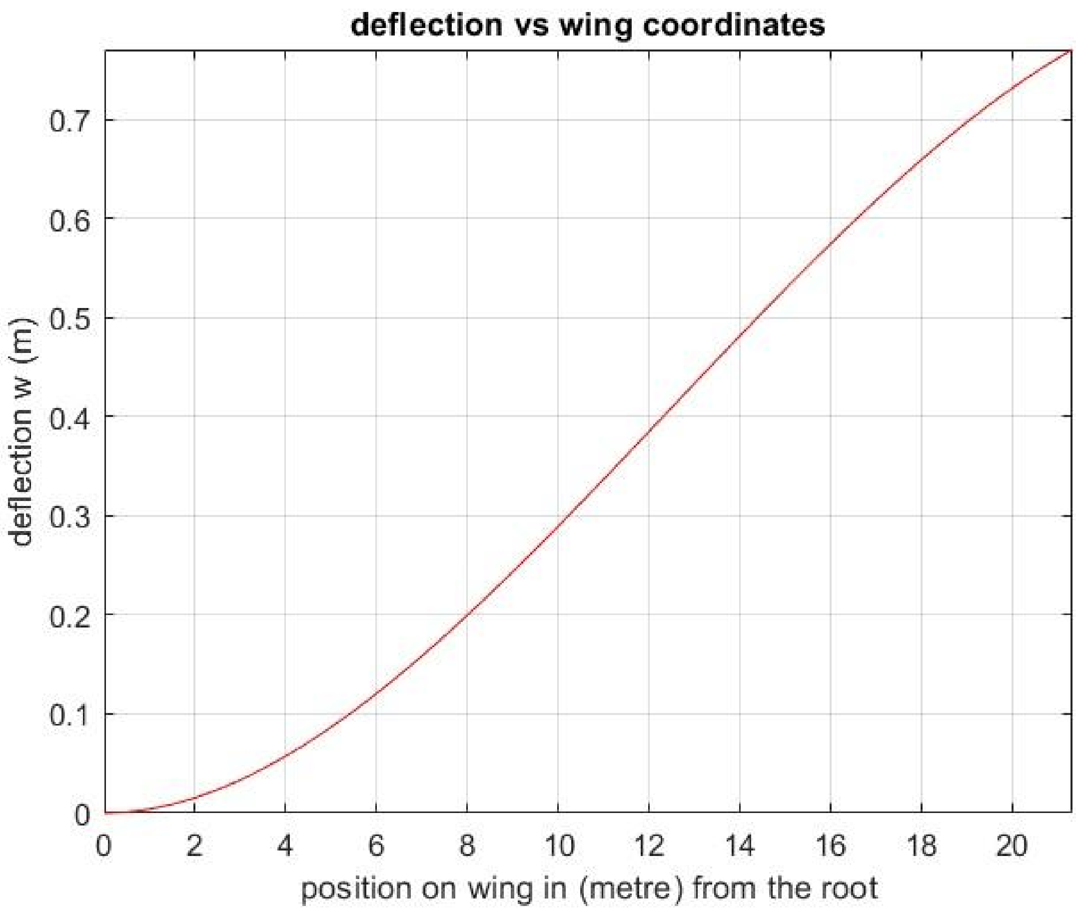
Plots for elastic wing case

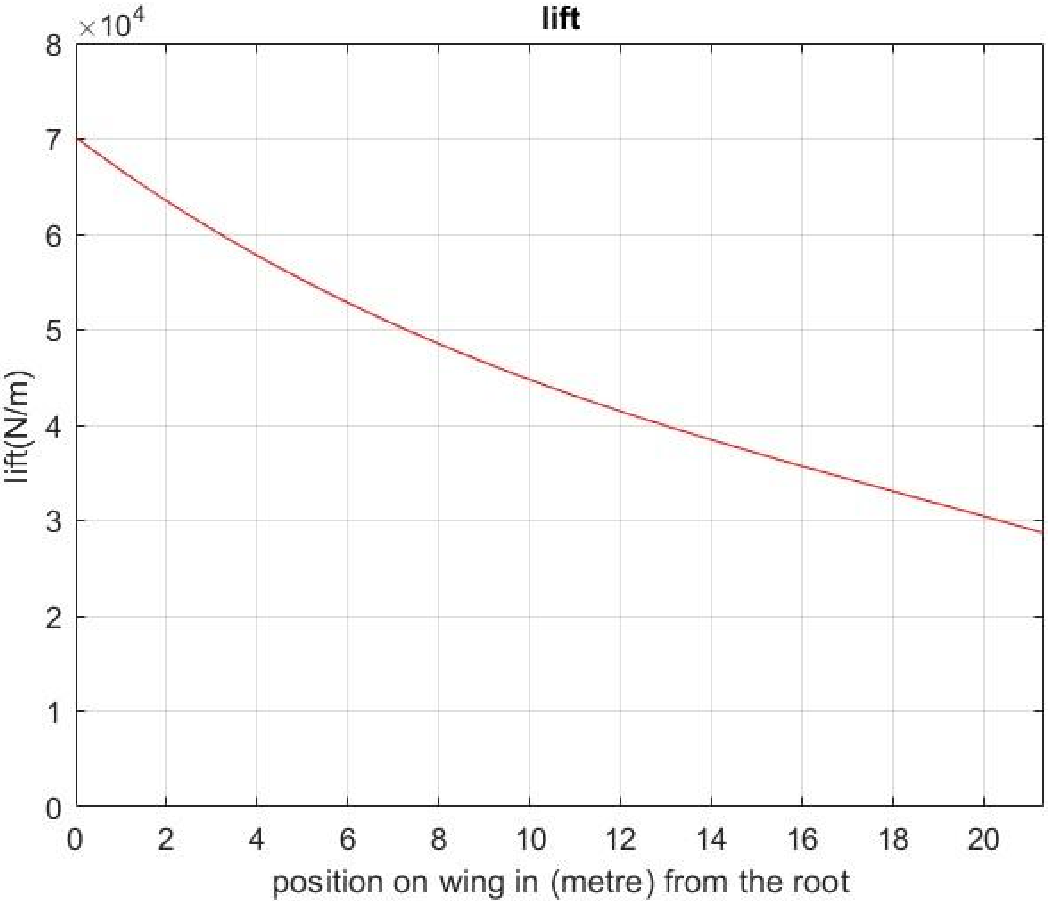
Plots taken:

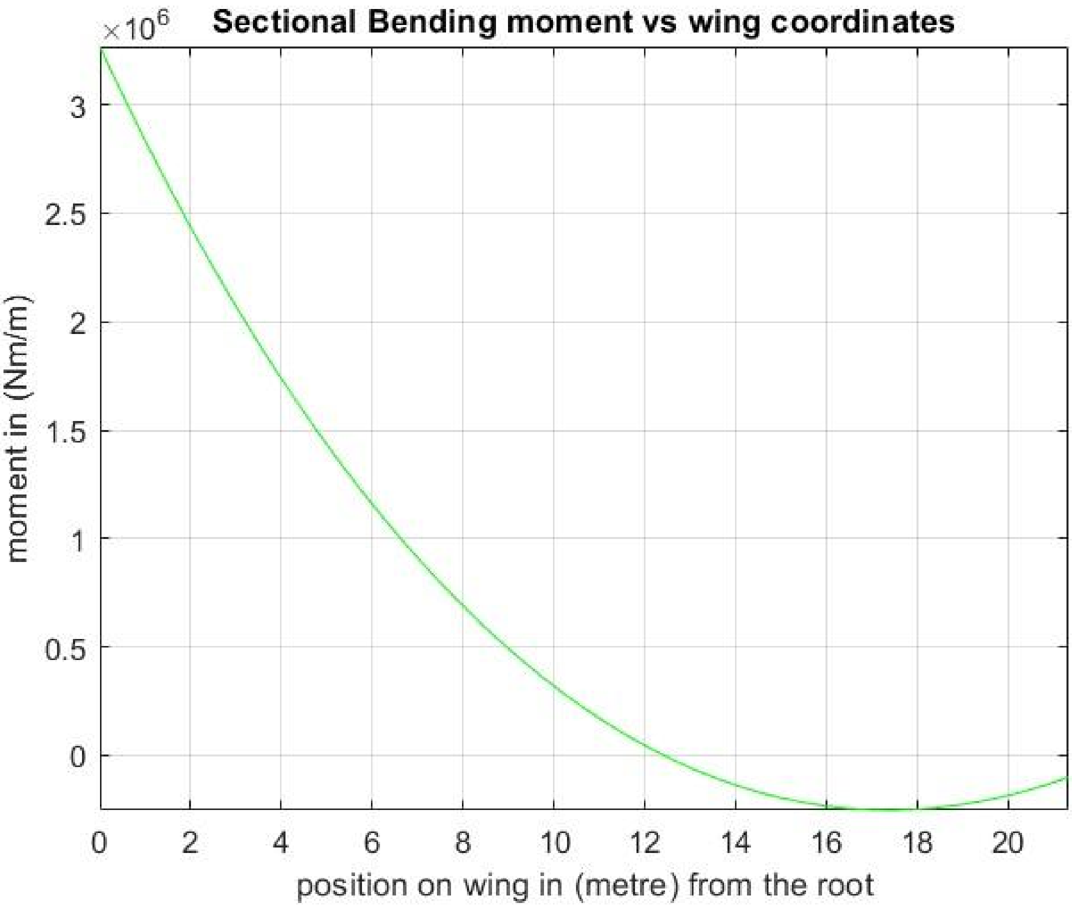
1. Effective angle of attack
2. Bending moment loading
3. Deflection
4. Lift
5. Sectional bending moment
6. Sectional shear force
7. Slope
8. Torsional moment loading
9. Sectional torque
10. Twist


effective angle$_o$f$_a$ttack vs wing coordinates
0.2
0.18
0.16
0.14
0.12
0.1
0.08
0.06
0.04
0.02
0
effective angle$_o$f$_a$ttack in radian
0
2
4
6
8
10
12
14
16
18
20
position on wing in (metre) from the root

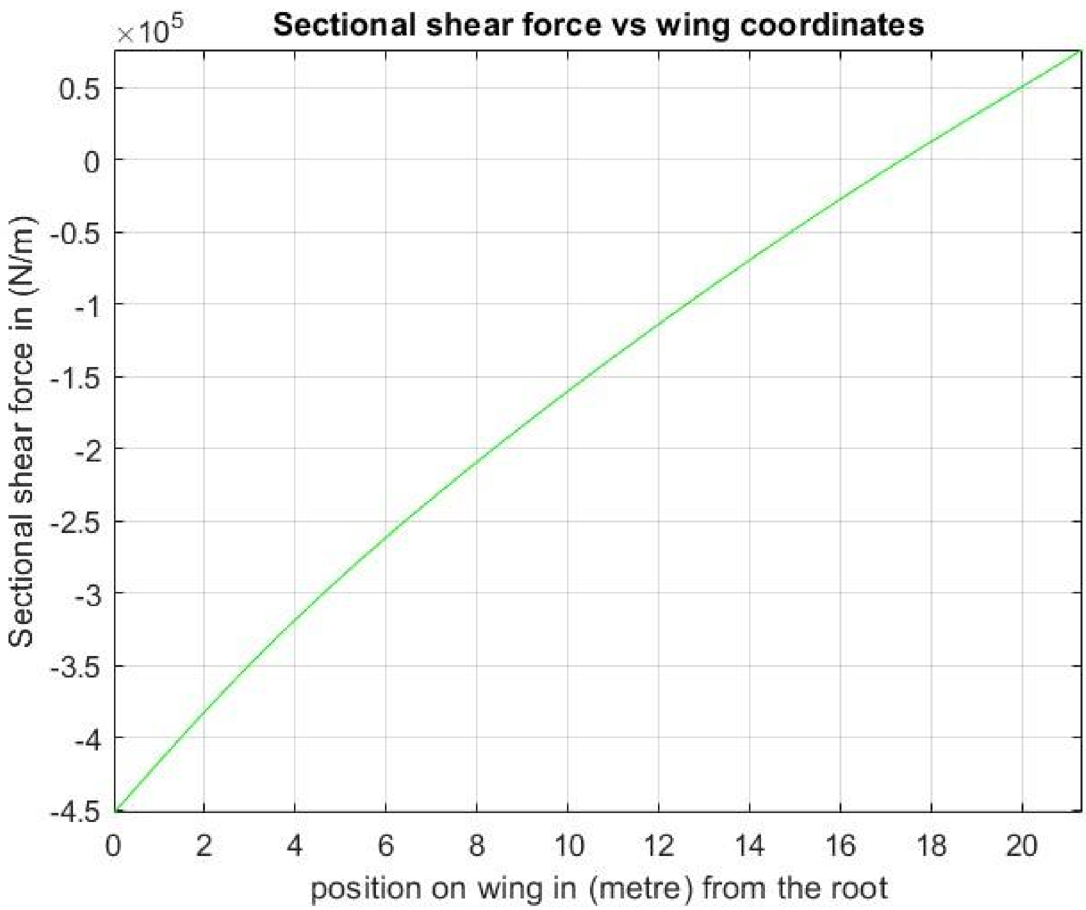

Bending moment loading vs wing coordinates
×10^4
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
Bending moment loading in (Nm/m)
0
2
4
6
8
10
12
14
16
18
20
position on wing in (metre) from the root


deflection vs wing coordinates
deflection w (m)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0
2
4
6
8
10
12
14
16
18
20
position on wing in (metre) from the root

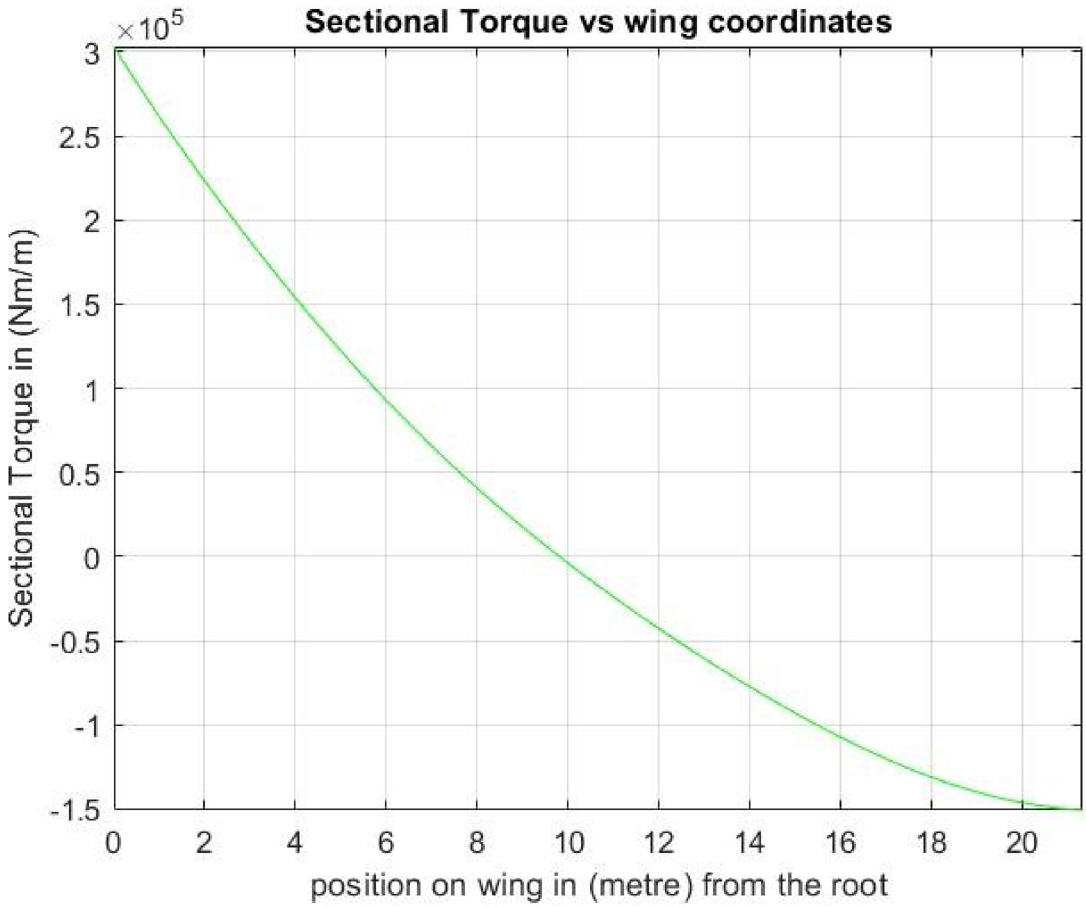

lift
×10^4
8
7
6
5
4
3
2
1
0
lift(N/m)
0
2
4
6
8
10
12
14
16
18
20
position on wing in (metre) from the root

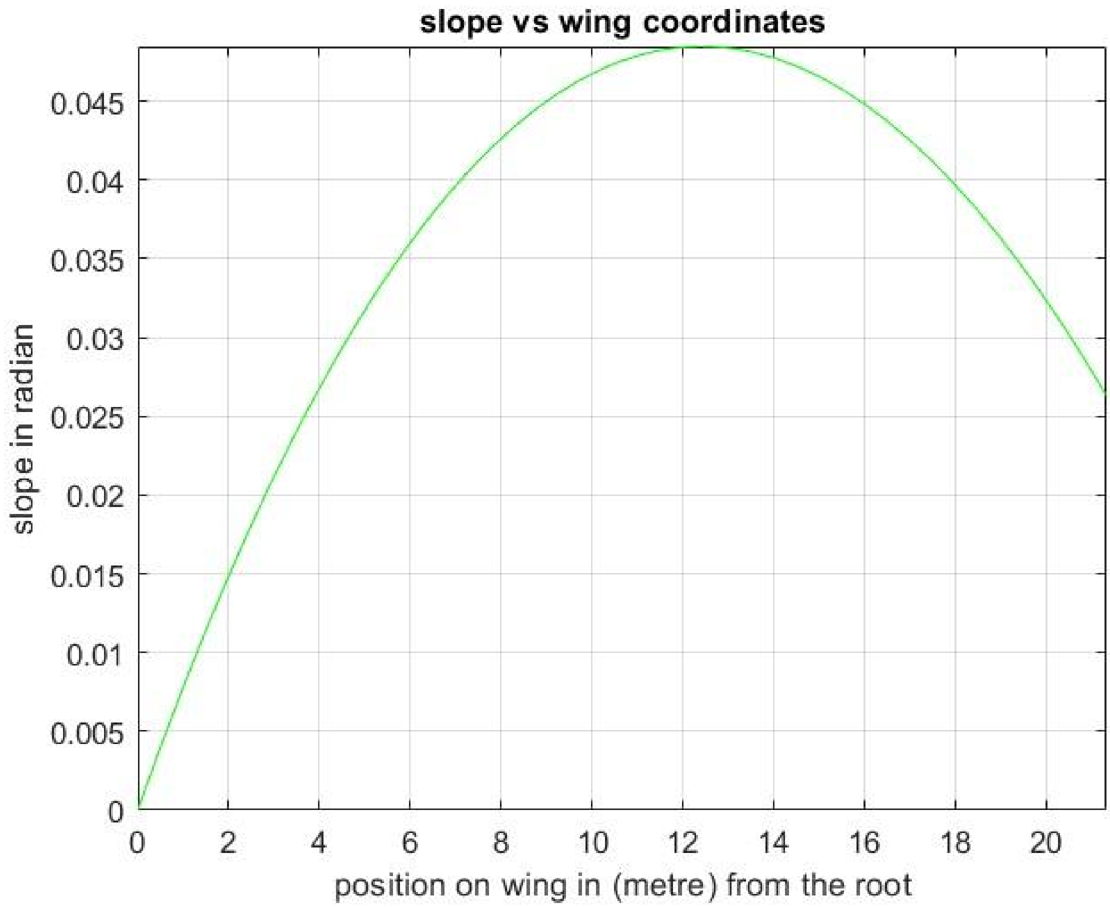


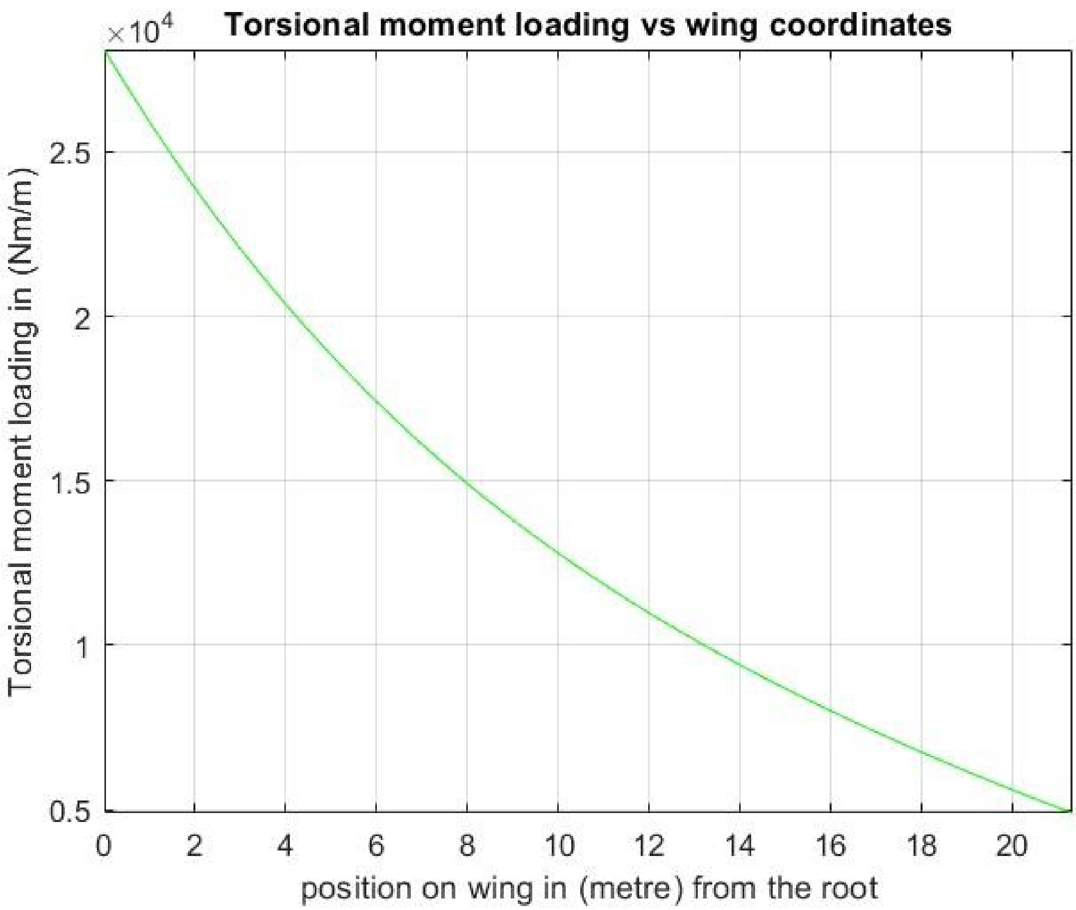
Sectional Bending moment vs wing coordinates
×10^6
3
2.5
2
1.5
1
0.5
0
moment in (Nm/m)
0
2
4
6
8
10
12
14
16
18
20
position on wing in (metre) from the root



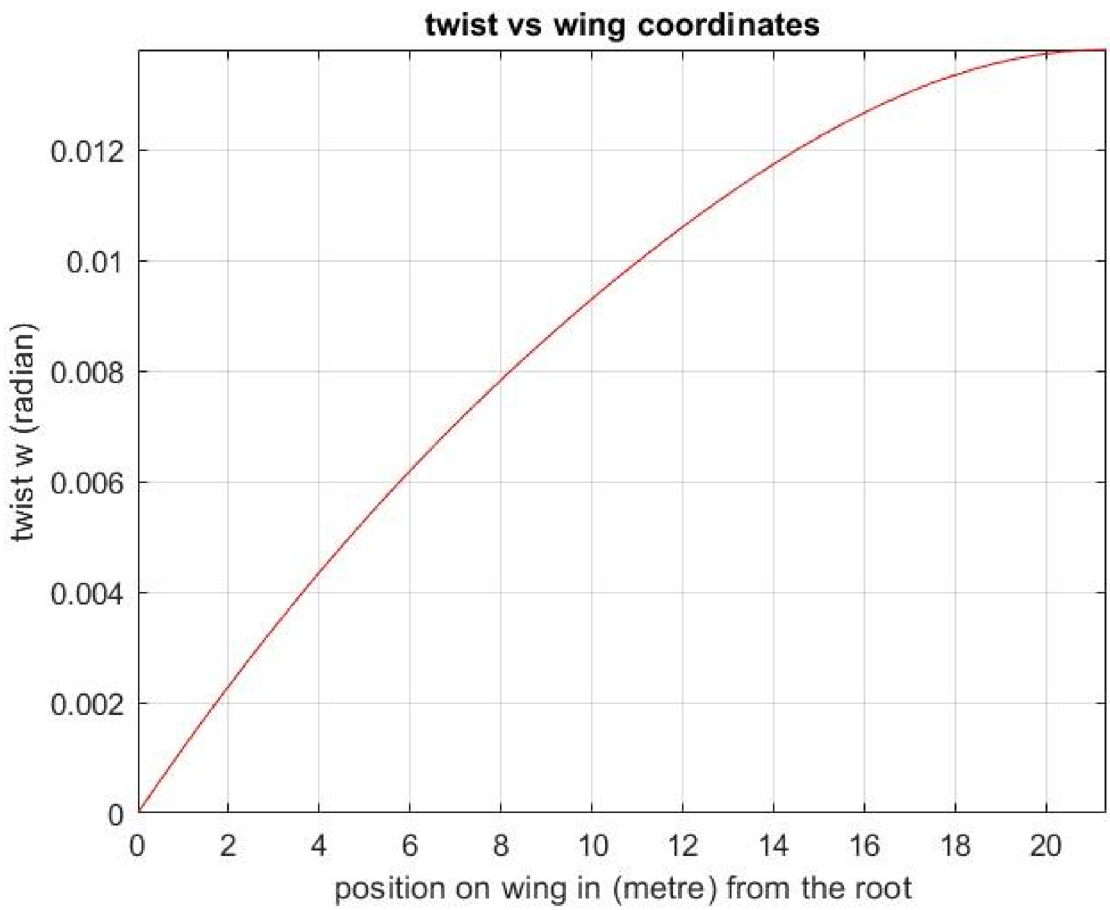
Sectional shear force vs wing coordinates
×10^5
0.5
0
-0.5
-1
-1.5
-2
-2.5
-3
-3.5
-4
-4.5
Sectional shear force in (N/m)
0
2
4
6
8
10
12
14
16
18
20
position on wing in (metre) from the root


Sectional Torque vs wing coordinates
×10^5
3
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
Sectional Torque in (Nm/m)
0
2
4
6
8
10
12
14
16
18
20
position on wing in (metre) from the root


slope vs wing coordinates
slope in radian
position on wing in (metre) from the root
0.045
0.04
0.035
0.03
0.025
0.02
0.015
0.01
0.005
0
0
2
4
6
8
10
12
14
16
18
20



Torsional moment loading vs wing coordinates
×10^4
2.5
2
1.5
1
0.5
0
2
4
6
8
10
12
14
16
18
20
Torsional moment loading in (Nm/m)
position on wing in (metre) from the root

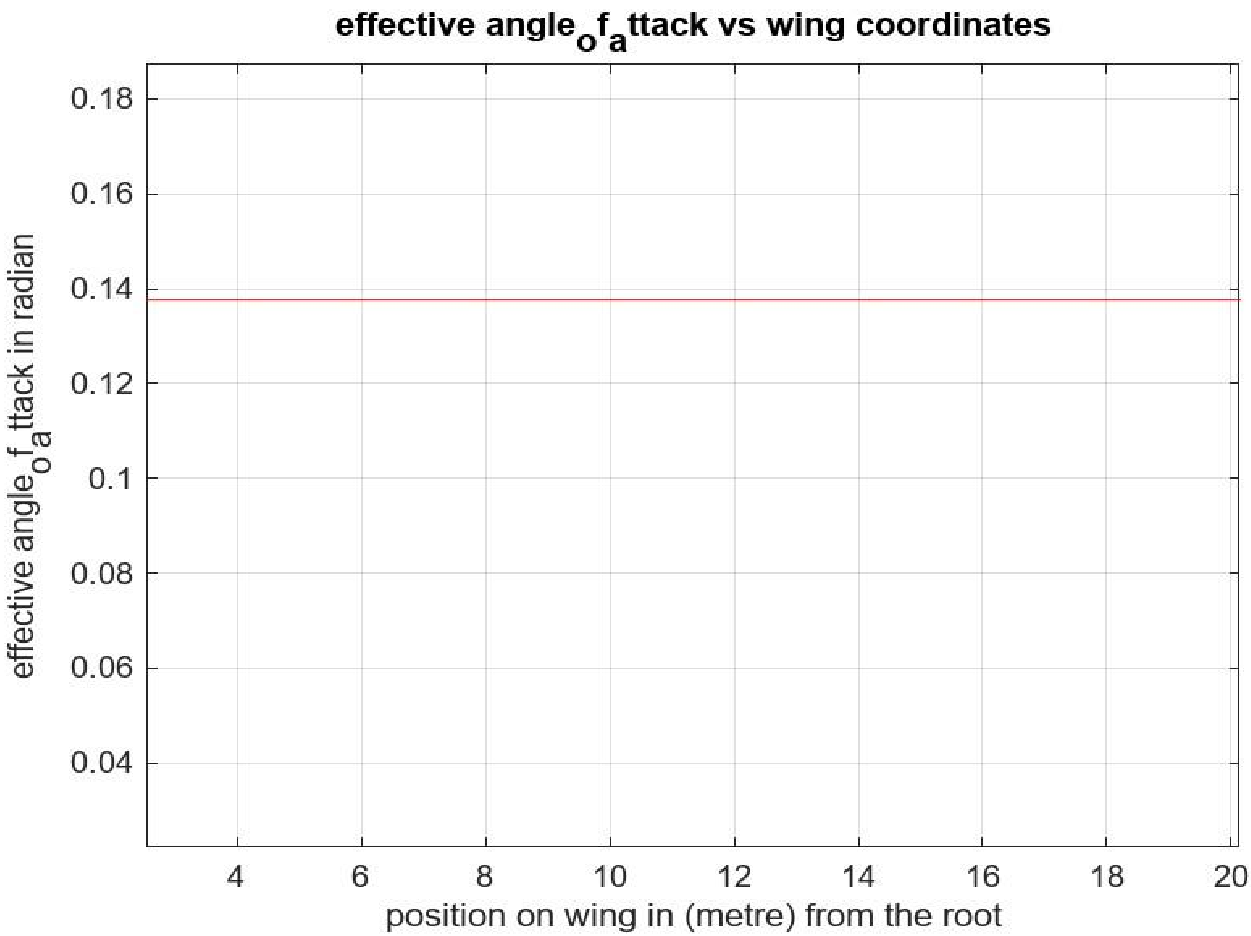


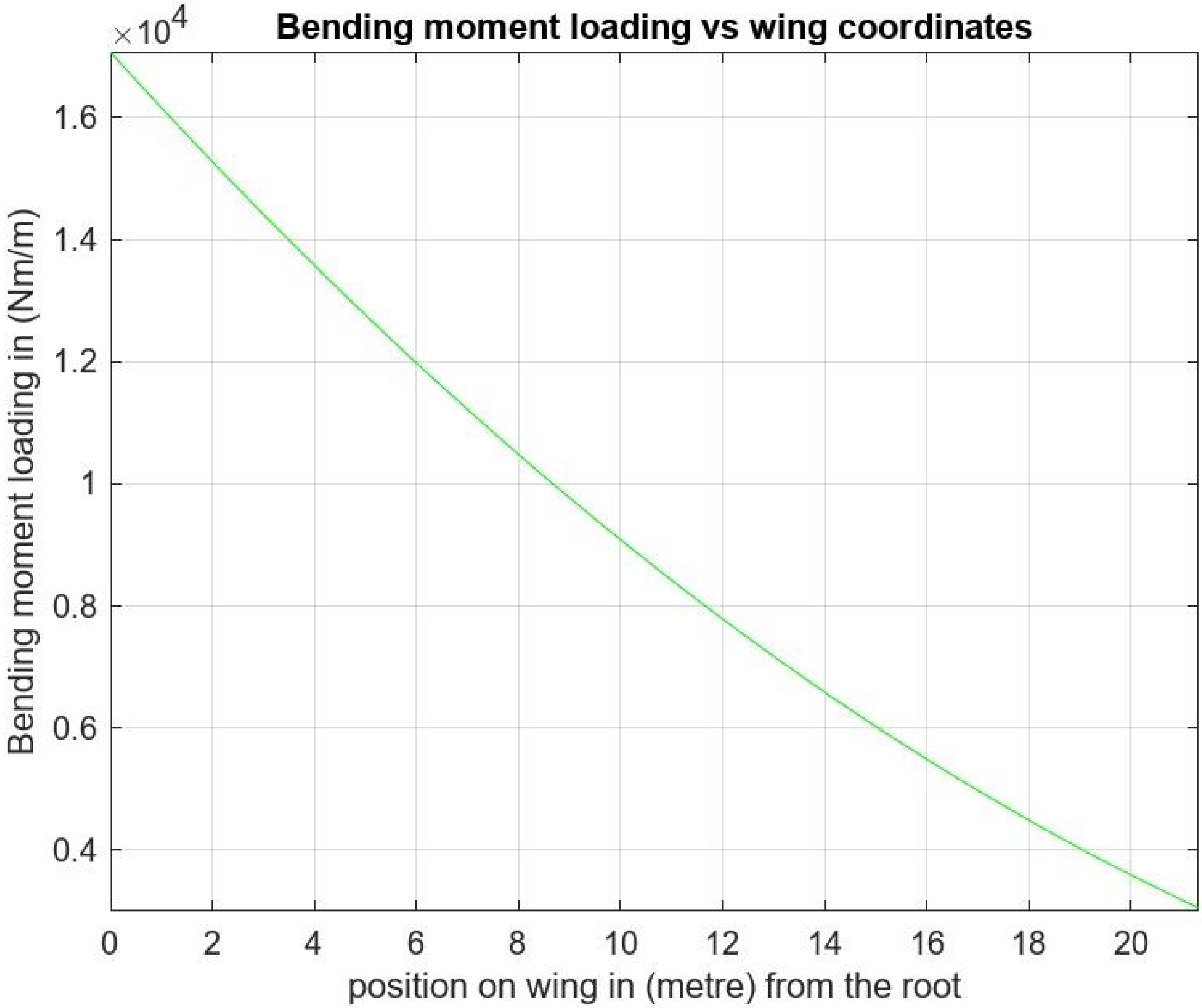
twist vs wing coordinates
twist w (radian)
0.012
0.01
0.008
0.006
0.004
0.002
0
0
2
4
6
8
10
12
14
16
18
20
position on wing in (metre) from the root

Plots for rigid wing case

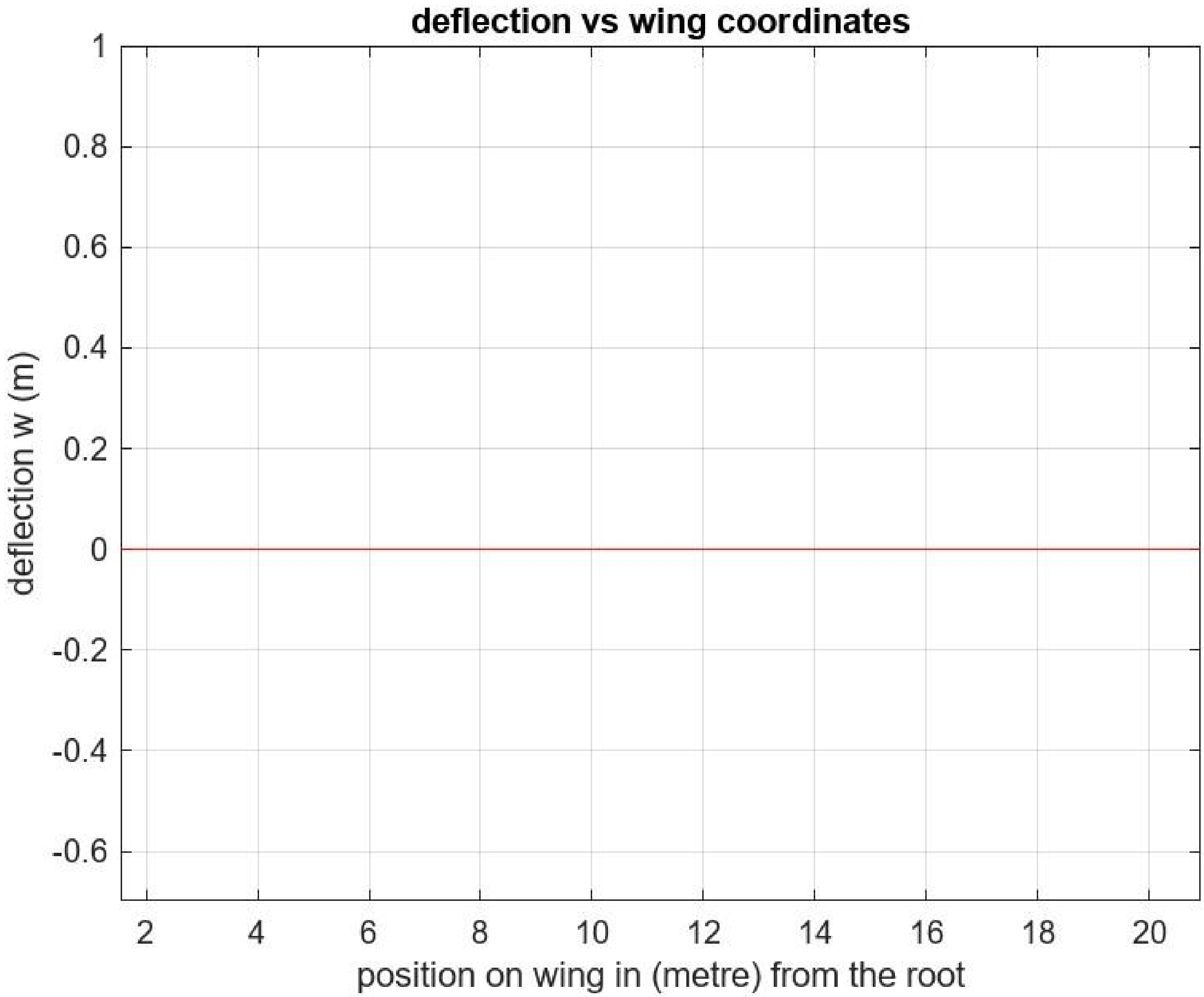
Plots taken:

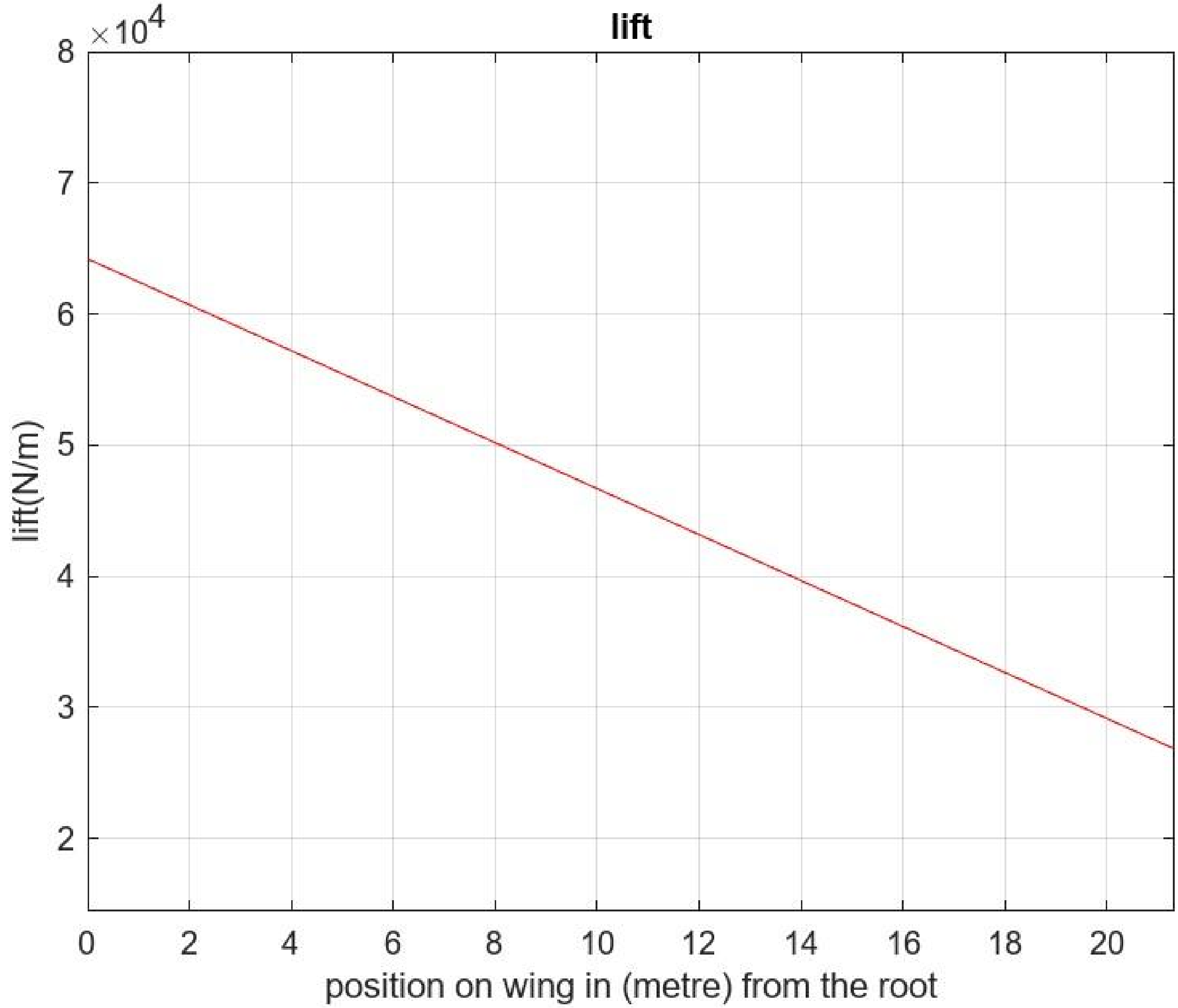
1. Effective angle of attack
2. Bending moment loading
3. Deflection
4. Lift
5. Sectional bending moment
6. Sectional shear force
7. Slope
8. Torsional moment loading
9. Sectional torque
10. Twist

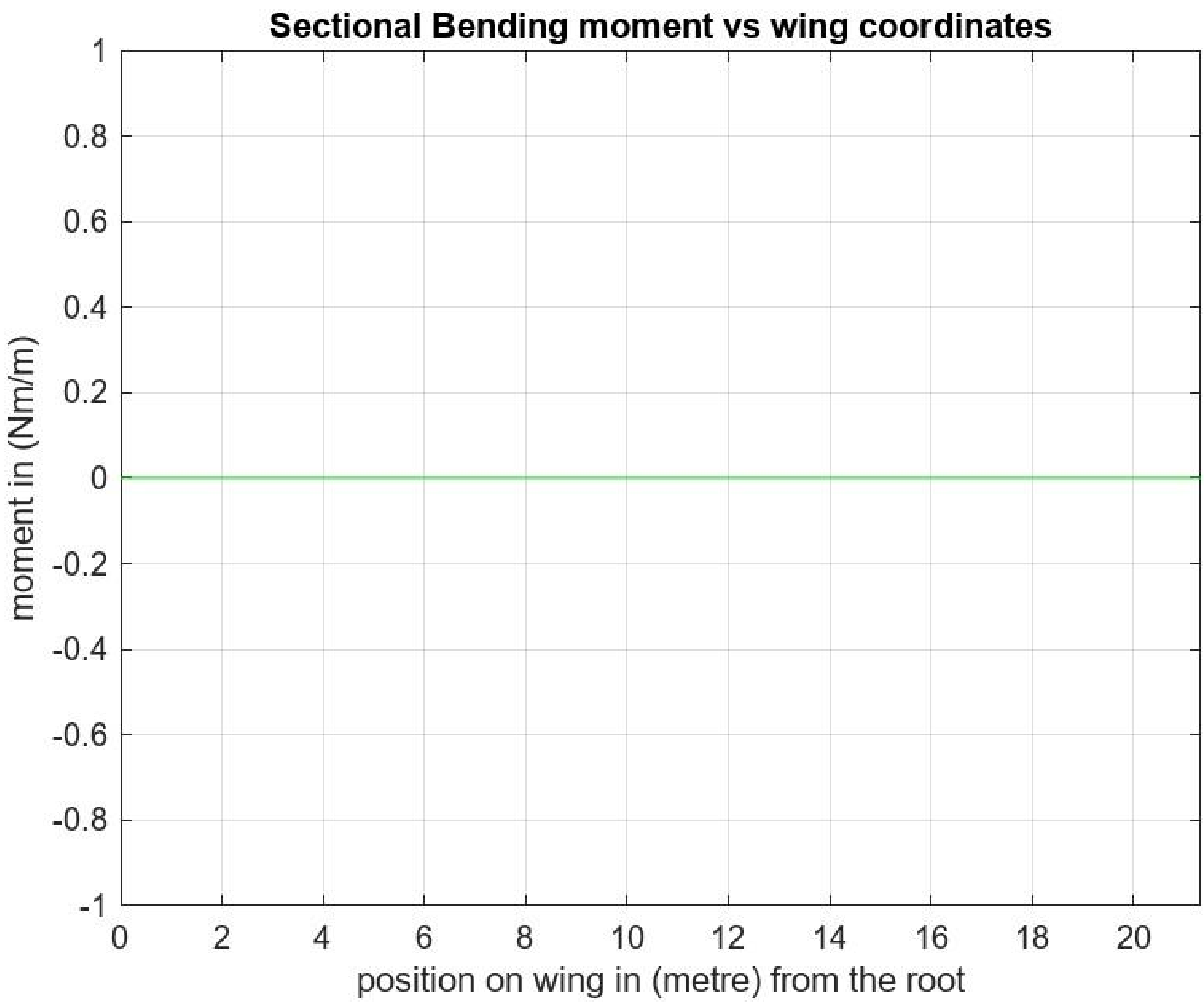

effective angle_of_attack vs wing coordinates
effective angle_of_attack in radian
0.18
0.16
0.14
0.12
0.1
0.08
0.06
0.04
4
6
8
10
12
14
16
18
20
position on wing in (metre) from the root

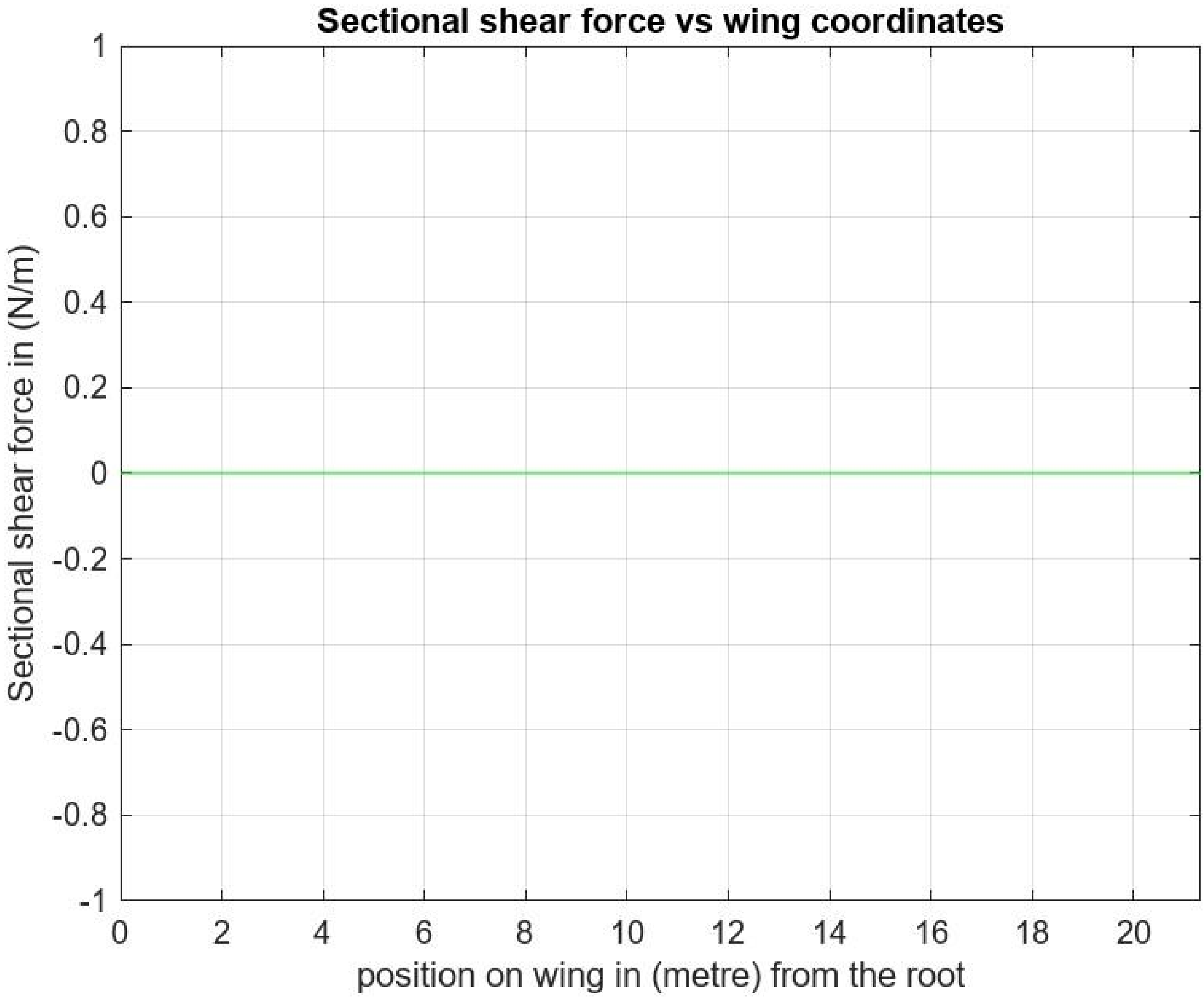


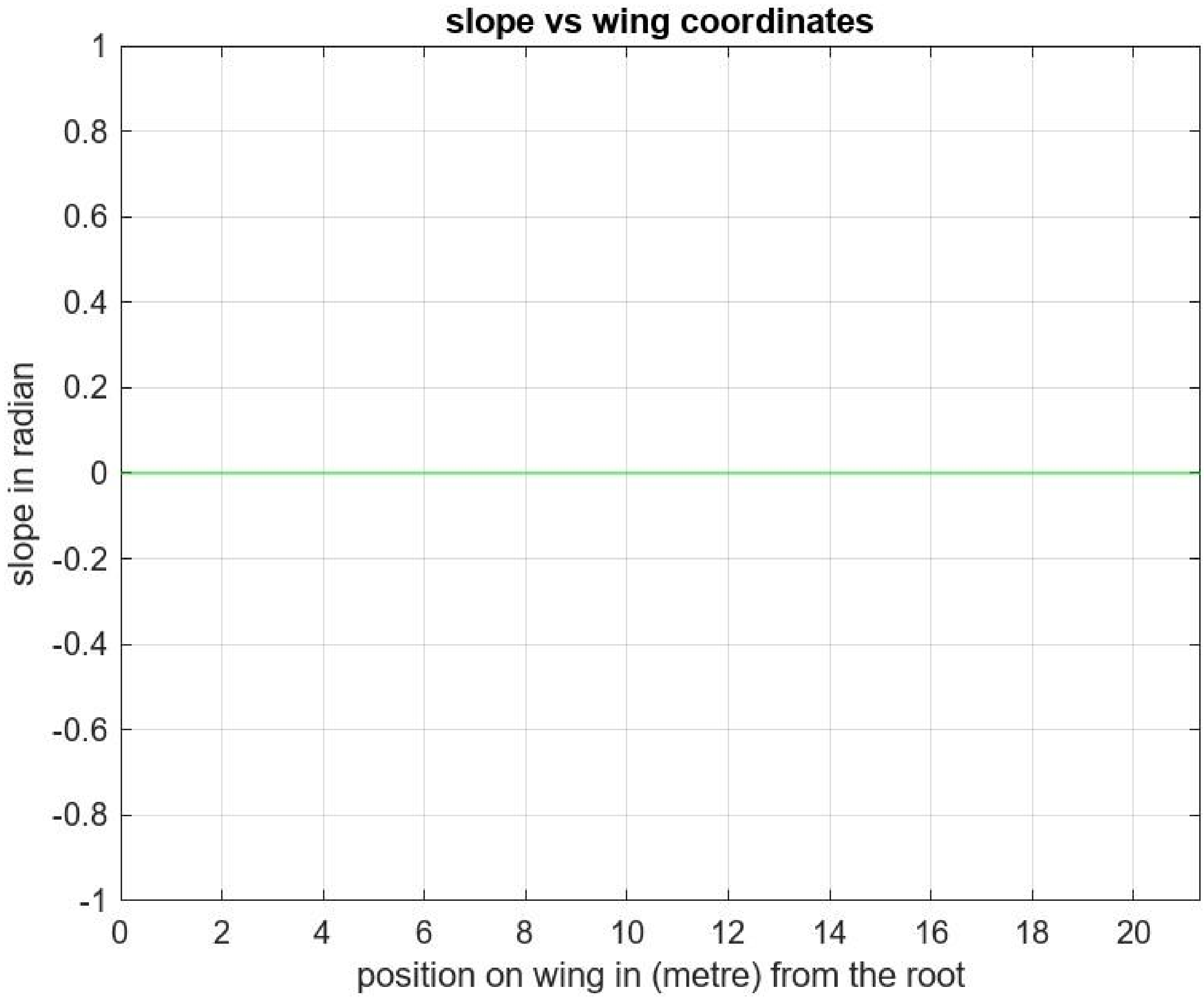
Bending moment loading vs wing coordinates
×10^4
Bending moment loading in (Nm/m)
position on wing in (metre) from the root
1.6
1.4
1.2
1
0.8
0.6
0.4
0
2
4
6
8
10
12
14
16
18
20

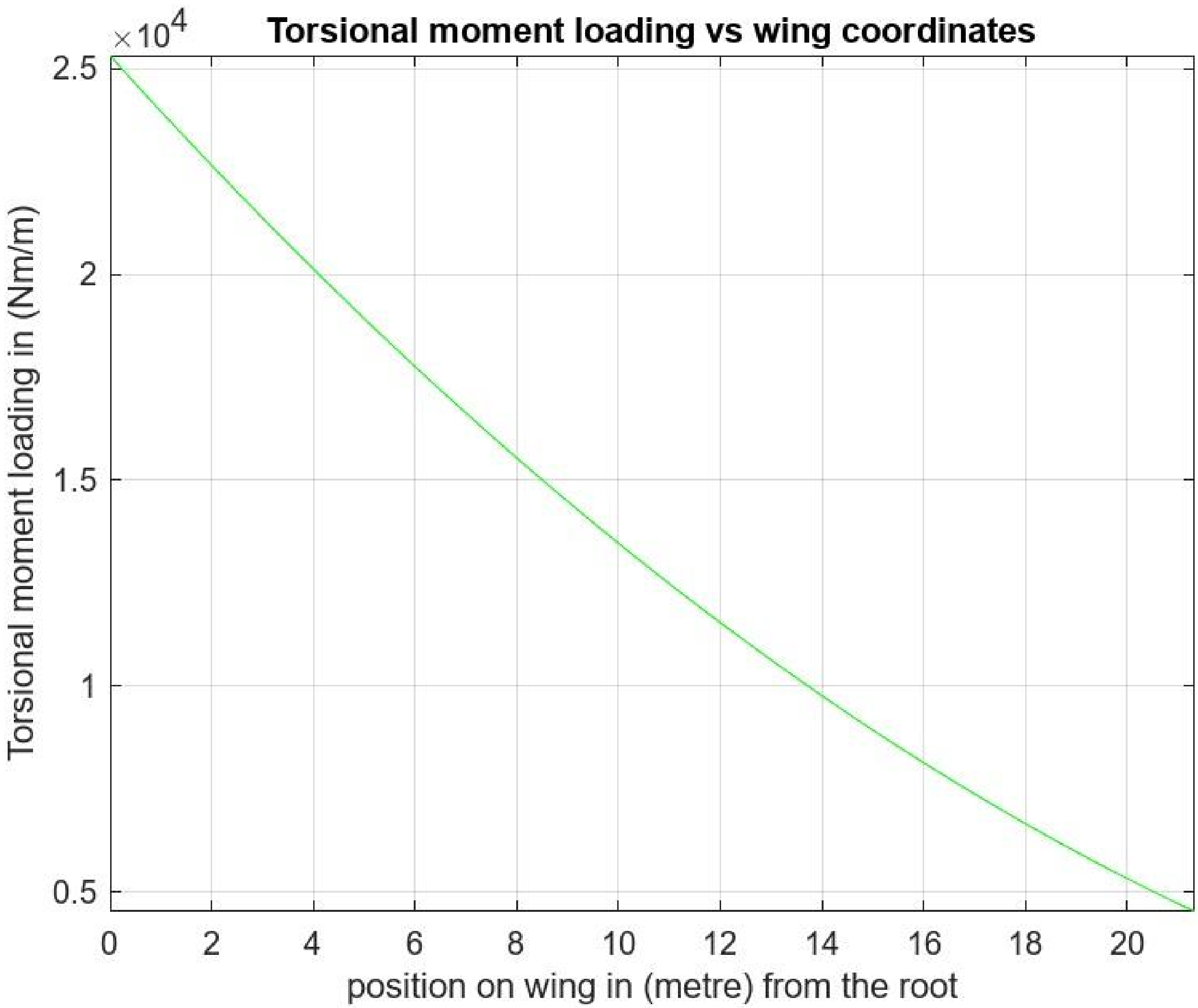

deflection vs wing coordinates
deflection w (m)
position on wing in (metre) from the root
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
2
4
6
8
10
12
14
16
18
20


lift
×10^4
lift(N/m)
position on wing in (metre) from the root
8
7
6
5
4
3
2
0
2
4
6
8
10
12
14
16
18
20

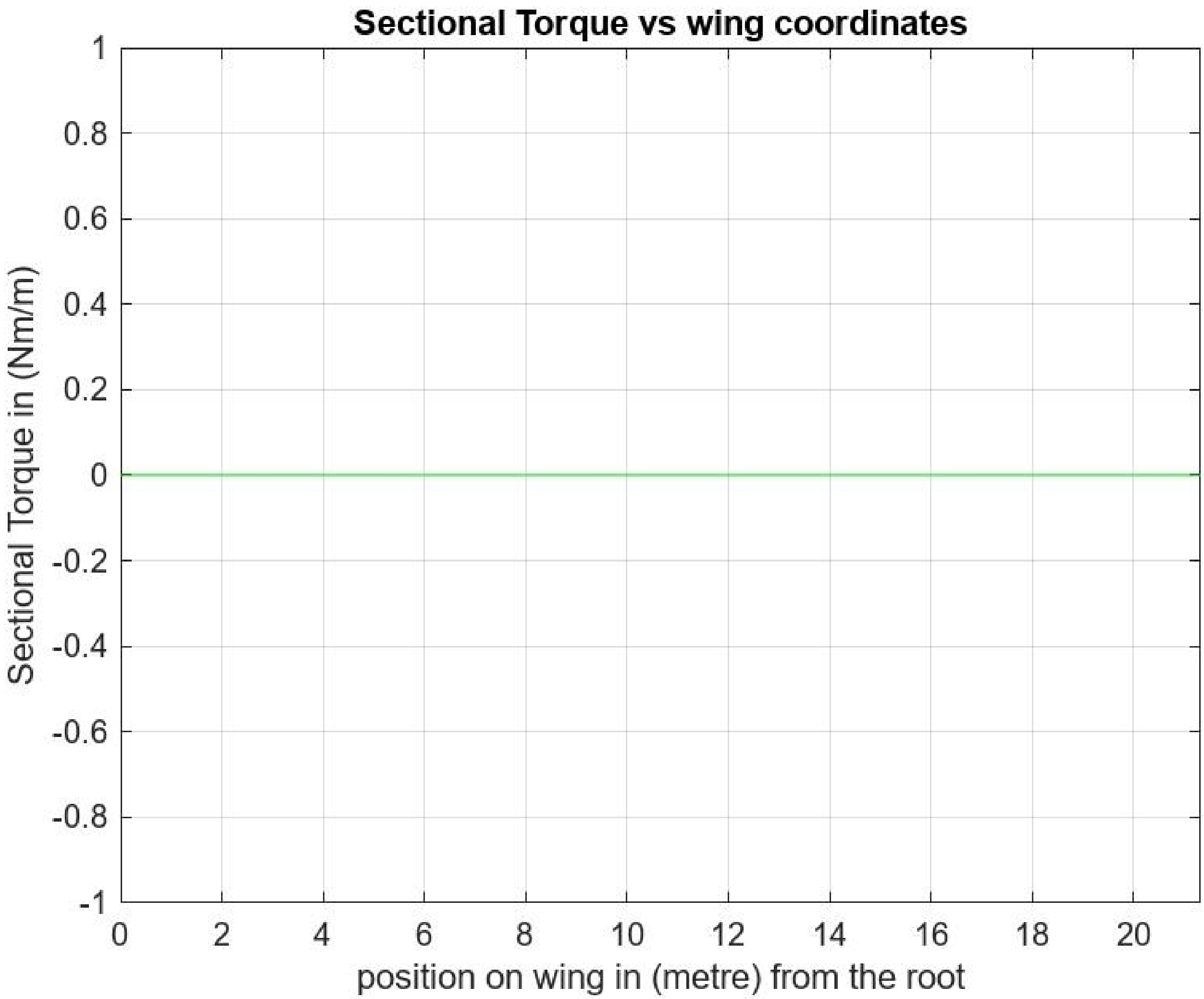

Sectional Bending moment vs wing coordinates
moment in (Nm/m)
position on wing in (metre) from the root
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
2
4
6
8
10
12
14
16
18
20


Sectional shear force vs wing coordinates
Sectional shear force in (N/m)
position on wing in (metre) from the root
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
2
4
6
8
10
12
14
16
18
20

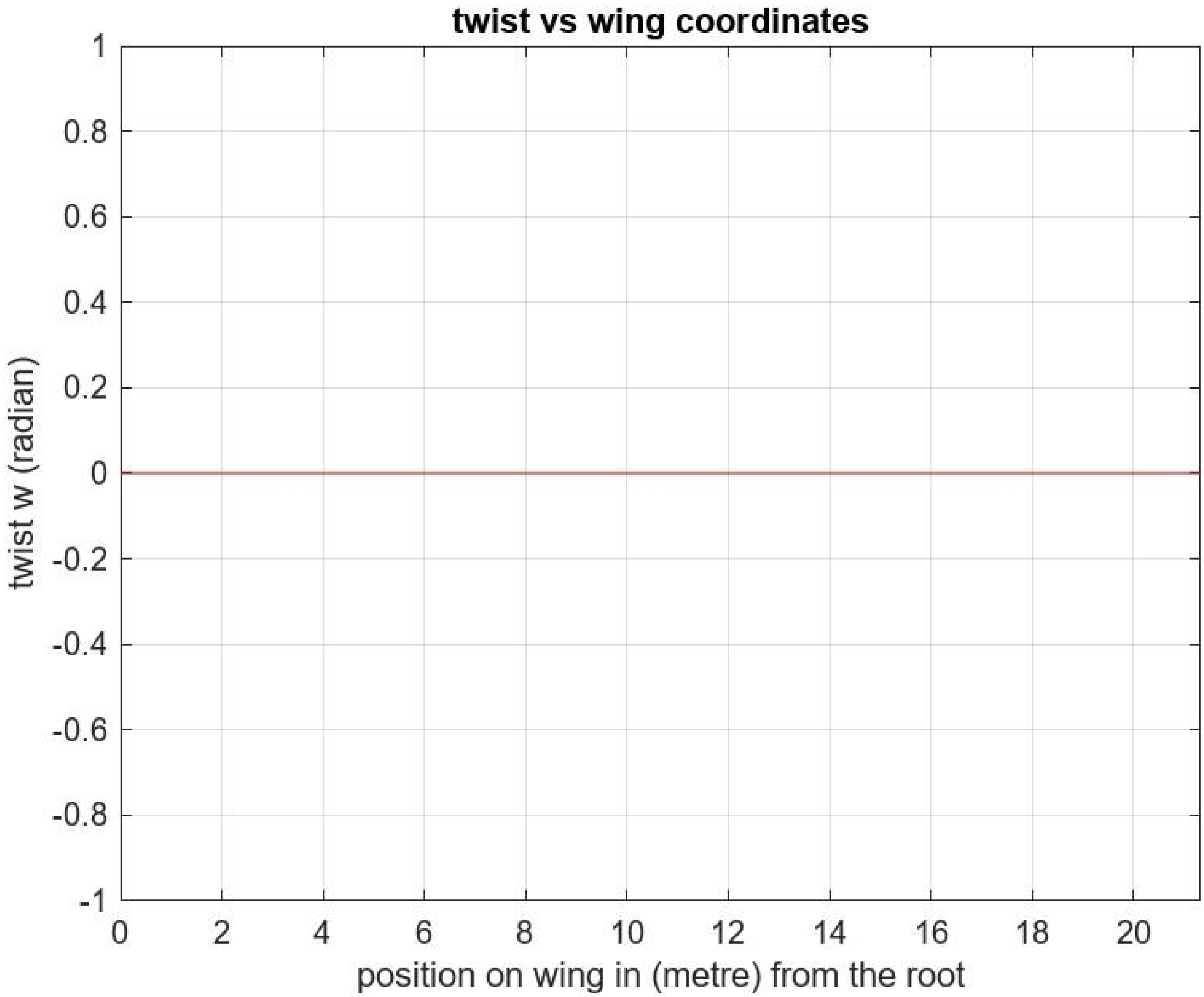

slope vs wing coordinates
slope in radian
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
2
4
6
8
10
12
14
16
18
20
position on wing in (metre) from the root



Torsional moment loading vs wing coordinates
$\times 10^4$
2.5
2
1.5
1
0.5
Torsional moment loading in (Nm/m)
0
2
4
6
8
10
12
14
16
18
20
position on wing in (metre) from the root



Sectional Torque vs wing coordinates
Sectional Torque in (Nm/m)
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
2
4
6
8
10
12
14
16
18
20
position on wing in (metre) from the root


twist vs wing coordinates
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
twist w (radian)
0
2
4
6
8
10
12
14
16
18
20
position on wing in (metre) from the root

Divergence dynamic pressure = 129184 Pa
Divergence Velocity = 941.2948 m/s
Divergence Mach No. = 2.96
Rigid angle of attack (flexible case): 0.15 radians
Rigid angle of attack (rigid case): 0.137 radians

Matlab code for flexible wing case

```
clc
clear all
syms GJ a b y w(y) t(Y) alpha alphae(y) M(y) Pdyn l(y) m(y) P(y) c(y) w(y)
theta(y) b1 b2 b3 b4 b5 a1 a2 a3 a4 a5 y
[Ma,Cla,Clac,Cmac,sweep,s,N,g,W,elevation,rho,T]=deal(0.8,6.28,5.05,-0.015,34*p
i/180,21.33,3,9.81,66000,6000,0.54,251.7);
Pdyn=(sqrt(1.4*287*T)*Ma)^2/2*.54; %N/m^2
%LINEARLY VARYING TERMS
[EI(y),GJ(y),c(y),M(y),ec(y),x_cg(y)]=deal(4e8-.178e8*y,2.5e8-.109e8*y,5.285-.1
442*y,800-28.13*y,.6875-0.0185*y,-0.106+.00288*y);
% no of terms
[n, m, w(y), theta(y)] = deal(5, 5, 0,0);
[a,b]=deal(sym('a',[1 n]),sym('b',[1 m]));
for i=1:n
w(y)=w(y)+a(i)*(y^4+6*s^2*y^2-4*y^3*s)^i;
end
for i=1:m
theta(y)=theta(y)+ b(i)*(-2*s*y+y^2)^i;
end
syms alpha
alphae(y)= alpha + theta(y)*cos(sweep)-diff(w(y),y)*sin(sweep);
syms l(y) m(y)
l(y)= Pdyn*Clac*c(y)*alphae(y);
P(y)= (l-N*M(y)*g)*cos(sweep);
m(y)=(l(y)*ec(y)+Pdyn*((c(y))^2)*Cmac-N*M(y)*g*x_cg(y))*(cos(sweep))^2;
D=EI(y)*diff(diff(w(y),y),y);
%residues
R1(y)=diff(diff(D,y),y)-P(y)-diff((l(y)*ec(y)+Pdyn*((c(y))^2)*Cmac-N*M(y)*g*x_c
g(y)),y)*sin(sweep)*cos(sweep);
R2(y)=diff((GJ(y)*diff(theta(y),y)),y)+m(y);
% squaring residues
[Q1,Q2] = deal(R1(y)*R1(y),R2(y)*R2(y));
% integrating square of the residues
[I1,I2] = deal(int(Q1, 0, s),int(Q2, 0, s));
```

```matlab
eq = cell(10,1);
% differentiating integrated square of the residues wrt coefficients
eq{1}=diff(I1 , a1);
eq{2}=diff(I1 , a2);
eq{3}=diff(I1 , a3);
eq{4}=diff(I1 , a4);
eq{5}=diff(I2 , b1);
eq{6}=diff(I2 , b2);
eq{7}=diff(I2 , b3);
eq{8}=diff(I2 , b4);
eq{9}=diff(I1 , a5);
eq{10}=diff(I2 , b5);
eq1=diff(I1 , a1);
eq2=diff(I1 , a2);
eq3=diff(I1 , a3);
eq4=diff(I1 , a4);
eq5=diff(I2 , b1);
eq6=diff(I2 , b2);
eq7=diff(I2 , b3);
eq8=diff(I2 , b4);
eq9=diff(I1 , a5);
eq10=diff(I2 , b5);
% divergence matrix
a= [a1;a2;a3;a4;b1;b2;b3;b4;a5;b5];
[a1, a2, a3, a4, b1, b2, b3, b4,a5,b5] =
solve([eq1,eq2,eq3,eq4,eq5,eq6,eq7,eq8,eq9,eq10],[a1, a2, a3, a4, b1, b2, b3,
b4,a5,b5]);
matrix = cell(10,1);
for i=1:10
matrix{i} = fliplr(coeffs(eq{i},a));
end
%calculation of rigid angle of attack
L = vpa(2*int(l(y),[0,s]),10);
eqalpha = subs(L-N*W*g);
alpha = vpa(solve(eqalpha),10);
%--------------------------------------------------------------------------
-----%
my(y)=(l(y)*ec(y)+Pdyn*((c(y))^2)*Cmac)*(cos(sweep))^2;
mx(y)=(l(y)*ec(y)+Pdyn*((c(y))^2)*Cmac)*sin(sweep)*cos(sweep);
%deflection
w(y)=vpa(subs(subs(w(y))),7);
%Twist
theta(y)=vpa(subs(subs(theta(y))),7);
%effective angle_of_attack
effectiveangle_of_attack(y)=subs(subs(alphae));
%lift
lift(y)= subs(subs(l(y)));
%slope
slope(y)=diff(w(y),y);
```

```
% moment
bendingmoment(y)=EI(y)*diff(slope(y),y);
%shear
shear(y)=diff(bendingmoment(y),y);
%torque
torque(y)=diff(GJ(y)*theta(y),y);
%Running torsional momemt
running_torsional_moment(y)=subs(subs(my(y)));
%Running Bending momemt
running_bending_moment(y)=subs(subs(mx(y)));
figure;
fplot(w(y),[0,21.33],'r')
grid on;
hold on
title('deflection vs wing coordinates');xlabel('position on wing in (metre)
from the root');ylabel('deflection w (m)');
%-----------------------------------------------%
figure;
fplot(theta(y),[0,21.33],'r')
grid on;hold on
title('twist vs wing coordinates');xlabel('position on wing in (metre) from the
root');ylabel('twist w (radian)');
%-----------------------------------------------%
figure;
fplot(effectiveangle_of_attack(y),[0,21.33],'r')
grid on;hold on;ylim([0 .2])
title('effective angle_of_attack vs wing coordinates');xlabel('position on wing
in (metre) from the root');
ylabel('effective angle_of_attack in radian');
%-----------------------------------------------%
figure;
fplot(lift(y),[0,21.33],'r')
grid on;hold on; ylim([0 8*10^4])
title('lift');xlabel('position on wing in (metre) from the
root');ylabel('lift(N/m)');
%-----------------------------------------------%
figure;
fplot(slope(y),[0,21.33],'g')
grid on;hold on
title('slope vs wing coordinates');xlabel('position on wing in (metre) from the
root');ylabel('slope in radian');
%-----------------------------------------------%
figure;
fplot(bendingmoment(y),[0,21.33],'g')
grid on;hold on
title('Sectional Bending moment vs wing coordinates');xlabel('position on wing
in (metre) from the root');
ylabel('moment in (Nm/m)');
%-----------------------------------------------%
```

```
figure;
fplot(shear(y),[0,21.33],'g')
grid on;hold on
title('Sectional shear force vs wing coordinates');xlabel('position on wing in
(metre) from the root');
ylabel('Sectional shear force in (N/m)');
%-----------------------------------------------%
figure;
fplot(torque(y),[0,21.33],'g')
grid on;hold on
title('Sectional Torque vs wing coordinates');xlabel('position on wing in
(metre) from the root');
ylabel('Sectional Torque in (Nm/m)');
%-----------------------------------------------%
figure;
fplot(running_torsional_moment(y),[0,21.33],'g')
grid on;hold on
title('Torsional moment loading vs wing coordinates');xlabel('position on wing
in (metre) from the root');
ylabel('Torsional moment loading in (Nm/m)');
%-----------------------------------------------%
figure;
fplot(running_bending_moment,[0,21.33],'g')
grid on;hold on
title('Bending moment loading vs wing coordinates');xlabel('position on wing in
(metre) from the root');
ylabel('Bending moment loading in (Nm/m)');
fprintf('\n rigid angle of attack = %0.4f radian', alpha);
```

Matlab code for rigid wing case

```
clc
clear all
syms GJ_root GJ_tip GJ b y w(y) t(Y) alpha alphae(y) M(y) Pdyn l(y) m(y) P(y)
c(y) w(y) theta(y) b1 b2 b3 b4 b5 a1 a2 a3 a4 a5 y z
[Ma,Cla,Clac,Cmac,sweep,s,N,g,W,elevation,rho,T]=deal(0.8,6.28,5.05,-0.015,34*p
i/180,21.33,3,9.81,66000,6000,0.54,251.7);
Pdyn=(sqrt(1.4*287*T)*Ma)^2/2*.54; %N/m^2
%LINEARLY VARYING TERMS
[EI(y),GJ(y),c(y),M(y),ec(y),x_cg(y)]=deal(4e8-.178e8*y,2.5e8-.109e8*y,5.285-.1
442*y,800-28.13*y,.6875-0.0185*y,-0.106+.00288*y);
% no of terms
[n, m, w(y), theta(y)] = deal(5, 5, 0,0);
[a,b]=deal(sym('a',[1 n]),sym('b',[1 m]));
syms alpha
alphae(y)= alpha + theta(y)*cos(sweep)-diff(w(y),y)*sin(sweep);
syms l(y) m(y)
l(y)= Pdyn*Clac*c(y)*alphae(y);
```

```
P(y)= (l-N*M(y)*g)*cos(sweep);
m(y)=(l(y)*ec(y)+Pdyn*((c(y))^2)*Cmac-N*M(y)*g*x_cg(y))*(cos(sweep))^2;
D=EI(y)*diff(diff(w(y),y),y);
%residues
R1(y)=diff(diff(D,y),y)-P(y)-diff((l(y)*ec(y)+Pdyn*((c(y))^2)*Cmac-N*M(y)*g*x_c
g(y)),y)*sin(sweep)*cos(sweep);
R2(y)=diff((GJ(y)*diff(theta(y),y)),y)+m(y);
% squaring residues
[Q1,Q2] = deal(R1(y)*R1(y),R2(y)*R2(y));
% integrating square of the residues
[I1,I2] = deal(int(Q1, 0, s),int(Q2, 0, s));
eq = cell(10,1);
% differentiating integrated square of the residues wrt coefficients
eq{1}=diff(I1 , a1);
eq{2}=diff(I1 , a2);
eq{3}=diff(I1 , a3);
eq{4}=diff(I1 , a4);
eq{5}=diff(I2 , b1);
eq{6}=diff(I2 , b2);
eq{7}=diff(I2 , b3);
eq{8}=diff(I2 , b4);
eq{9}=diff(I1 , a5);
eq{10}=diff(I2 , b5);
eq1=diff(I1 , a1);
eq2=diff(I1 , a2);
eq3=diff(I1 , a3);
eq4=diff(I1 , a4);
eq5=diff(I2 , b1);
eq6=diff(I2 , b2);
eq7=diff(I2 , b3);
eq8=diff(I2 , b4);
eq9=diff(I1 , a5);
eq10=diff(I2 , b5);
% divergence matrix
a= [a1;a2;a3;a4;b1;b2;b3;b4;a5;b5];
[a1, a2, a3, a4, b1, b2, b3, b4,a5,b5] =
solve([eq1,eq2,eq3,eq4,eq5,eq6,eq7,eq8,eq9,eq10],[a1, a2, a3, a4, b1, b2, b3,
b4,a5,b5]);
matrix = cell(10,1);
for i=1:10
matrix{i} = fliplr(coeffs(eq{i},a));
end
%calculation of rigid angle of attack
L = vpa(2*int(l(y),[0,s]),10);
eqalpha = subs(L-N*W*g);
alpha = vpa(solve(eqalpha),10);
%---------------------------------------------------------------------------
-----%
my(y)=(l(y)*ec(y)+Pdyn*((c(y))^2)*Cmac)*(cos(sweep))^2;
```

```
mx(y)=(l(y)*ec(y)+Pdyn*((c(y))^2)*Cmac)*sin(sweep)*cos(sweep);
%deflection
w(y)=vpa(subs(subs(w(y))),7);
%Twist
theta(y)=vpa(subs(subs(theta(y))),7);
%effective angle_of_attack
effectiveangle_of_attack(y)=subs(subs(alphae));
%lift
lift(y)= subs(subs(l(y)));
%slope
slope(y)=diff(w(y),y);
% moment
bendingmoment(y)=EI(y)*diff(slope(y),y);
%shear
shear(y)=diff(bendingmoment(y),y);
%torque
torque(y)=diff(GJ(y)*theta(y),y);
%Running torsional momemt
running_torsional_moment(y)=subs(subs(my(y)));
%Running Bending momemt
running_bending_moment(y)=subs(subs(mx(y)));
figure;
fplot(w(y),[0,21.33],'r')
grid on;
hold on
title('deflection vs wing coordinates');xlabel('position on wing in (metre)
from the root');ylabel('deflection w (m)');
%-----------------------------------------------%
figure;
fplot(theta(y),[0,21.33],'r')
grid on;hold on
title('twist vs wing coordinates');xlabel('position on wing in (metre) from the
root');ylabel('twist w (radian)');
%-----------------------------------------------%
figure;
fplot(effectiveangle_of_attack(y),[0,21.33],'r')
grid on;hold on;ylim([0 .2])
title('effective angle_of_attack vs wing coordinates');xlabel('position on wing
in (metre) from the root');
ylabel('effective angle_of_attack in radian');
%-----------------------------------------------%
figure;
fplot(lift(y),[0,21.33],'r')
grid on;hold on; ylim([0 8*10^4])
title('lift');xlabel('position on wing in (metre) from the
root');ylabel('lift(N/m)');
%--------------------------------------------------%
figure;
fplot(slope(y),[0,21.33],'g')
```

```
grid on;hold on
title('slope vs wing coordinates');xlabel('position on wing in (metre) from the
root');ylabel('slope in radian');
%-----------------------------------------------%
figure;
fplot(bendingmoment(y),[0,21.33],'g')
grid on;hold on
title('Sectional Bending moment vs wing coordinates');xlabel('position on wing
in (metre) from the root');
ylabel('moment in (Nm/m)');
%-----------------------------------------------%
figure;
fplot(shear(y),[0,21.33],'g')
grid on;hold on
title('Sectional shear force vs wing coordinates');xlabel('position on wing in
(metre) from the root');
ylabel('Sectional shear force in (N/m)');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure;
fplot(torque(y),[0,21.33],'g')
grid on;hold on
title('Sectional Torque vs wing coordinates');xlabel('position on wing in
(metre) from the root');
ylabel('Sectional Torque in (Nm/m)');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure;
fplot(running_torsional_moment(y),[0,21.33],'g')
grid on;hold on
title('Torsional moment loading vs wing coordinates');xlabel('position on wing
in (metre) from the root');
ylabel('Torsional moment loading in (Nm/m)');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure;
fplot(running_bending_moment,[0,21.33],'g')
grid on;hold on
title('Bending moment loading vs wing coordinates');xlabel('position on wing in
(metre) from the root');
ylabel('Bending moment loading in (Nm/m)');
fprintf('\n rigid angle of attack = %0.4f radian', alpha);
```

Matlab code for divergence analysis

```
clc
clear all
syms GJ_root GJ_tip GJ b y w(y)  t(Y) alphae(y) M(y) Pdyn l(y) P(y) c(y) w(y)
theta(y) b1 b2 b3 b4 b5 a1 a2 a3 a4 a5 y z
[Mc,Cla,Clac,Cmac,sweep,s,N,g,W,elevation,rho,T]=deal(0.8,6.28,5.05,-0.015,34*p
i/180,21.33,3,9.81,66000,6000,0.54,251.7);
%Pdyn=(sqrt(1.4*287*T)*Mc)^2/2*.66; %N/m^2
%LINEARLY VARYING TERMS
[EI(y),GJ(y),c(y),M(y),ec(y),x_cg(y)]=deal(4e8-.1782e8*y,2.5e8-.109e8*y,5.285-.
144253*y,800-28.13*y,.68705-0.018753*y,-0.106+.00288*y);
% no of terms
[n, m, w(y), theta(y)] = deal(5, 5, 0,0);
[a,b]=deal(sym('a',[1 n]),sym('b',[1 m]));
for i=1:n
w(y)=w(y)+a(i)*(y^4+6*s^2*y^2-4*y^3*s)^i;
end
for i=1:m
theta(y)=theta(y)+ b(i)*(-2*s*y+y^2)^i;
end
syms alpha
alphae(y)= alpha + theta(y)*cos(sweep)-diff(w(y),y)*sin(sweep);
syms l(y) m(y)
l(y)= Pdyn*Clac*c(y)*alphae(y);
P(y)= (l-N*M(y)*g)*cos(sweep);
m(y)=(l(y)*ec(y)+Pdyn*((c(y))^2)*Cmac-N*M(y)*g*x_cg(y))*(cos(sweep))^2;
D=EI(y)*diff(diff(w(y),y),y);
%residuals
R1(y)=diff(diff(D,y),y)-P(y)-diff((l(y)*ec(y)+Pdyn*((c(y))^2)*Cmac-N*M(y)*g*x_c
g(y)),y)*sin(sweep)*cos(sweep);
R2(y)=diff((GJ(y)*diff(theta(y),y)),y)+m(y);
[Q1,Q2] = deal(R1(y)*R1(y),R2(y)*R2(y));
% using weighted integral residual -least square error
[I1,I2] = deal(int(Q1, 0, s),int(Q2, 0, s));
eq = cell(10,1);
eq{1}=diff(I1 , a1);
eq{2}=diff(I1 , a2);
eq{3}=diff(I1 , a3);
eq{4}=diff(I1 , a4);
eq{5}=diff(I2 , b1);
eq{6}=diff(I2 , b2);
eq{7}=diff(I2 , b3);
eq{8}=diff(I2 , b4);
eq{9}=diff(I1 , a5);
eq{10}=diff(I2 , b5);
% divergence matrix formation
a= [a1;a2;a3;a4;b1;b2;b3;b4;a5;b5];
```

```
matrix = cell(10,1);
for i=1:10
   matrix{i} = fliplr(coeffs(eq{i},a));
end
%calculation of rigid angle of attack
L = vpa(2*int(l(y),[0,s]),7);
alpha_val = subs(L-N*W*g);
alpha = vpa(solve(alpha_val),7);
D= subs(matrix);
coeff = D(:,1:10);
% Solving for divergence dynamic pressure
syms Pdyn
eqn = det(coeff) == 0;
PdynD = solve(eqn, Pdyn);
% Converting symbolic expression to decimal
PdynD = vpa(subs(PdynD), 5);
% Removing imaginary and negative values
PdynD = PdynD((imag(PdynD)==0) & (PdynD>=0));
% minimum positive value for divergence pressure
PdynD = min(PdynD);
% Calculating divergence velocity and Mach number
Divergence_speed = sqrt(2*PdynD/rho^2);
MachNo = Divergence_speed/318; % Divergence Mach no at 6000m elevation
% Print results
fprintf('\n Divergence dynamic pressure = %0.4f Pa', PdynD);
fprintf('\n Divergence Velocity = %0.4f m/s', Divergence_speed);
fprintf('\n Divergence Mach No. = %0.4f', MachNo);
%-----------------------------------------------------------------------------%
```

REFERENCE

1. MATHWORKS MATLAB DOCUMENTATION (FOR CODE)
2. AE 678 CLASS NOTES
3. JN REDDY : ‘INTRODUCTION TO FINITE ELEMENT METHOD’
4. JD. ANDERSON : ‘FUNDAMENTALS OF AERODYNAMICS’