भारतीय प्रौद्योगिकी संस्थान हैदराबाद

Indian Institute of Technology Hyderabad

INDIAN INSTITUTE OF TECHNOLOGY HYDERABAD

ME5700

FEM Solver for a General Composite Supersonic Wing(Static Analysis)

Submitted To:
Chandra Prakash
Asst. Professor
Mechanical and Aerospace Engineering

Submitted By :
Anirudh K
ME22MTECH14017
MTech
2nd Semester

Contents

1 Introduction

A supersonic wing is usually thin in terms of thickness which has Low Aspect ratio i.e lesser span and larger area to minimize supersonic drag i.e wave drag. Hence it can be assumed as **Plate**.
The plate has been modelled as a Mindlin Plate which can take account for thickness effects whether it is a thick or thin plate.
The FEM solver also take account of displacements of spars,ribs and skin. In terms of spars and ribs it takes care of spar web,spar cap similarly spar ribs and spar cap which also contain composites.
The paper [4] has taken account for Isotropic material using mindlin plate theory and the paper [6] takes care of the transverse shear for Mindlin Plate. The paper [5] takes care of a general airfoil which takes care of supersonic airfoils available by giving appropriate constants.
This project is small and innovative because it can take any sweep,root chord,any type of ply and thickness of each ply as well any dimensions of spar cap, rib cap,spar web, rib web and any load obtained from Computational Fluid Dynamics data.

1.1 Formulation of FEM solver

The paper [4] has assumed a Mindlin Plate theory along with a basis function as Legendre Polynomials for the displacements obtained.
The plate is assumed to be trapezoidal hence it can take care of the geometric dimensions.The Jacobian is taken care by the Lagrangian shape function [4].

Assumptions involved in making of composite plates

To account for the transverse shear stress we break assumptions of that in the lamina the "transverse shear strain is neglected" to account for the mindlin plate independent degrees of freedom.
We know that from Mindlin Plate theory the constitutive matrix is for composite plate(in plane)

$$\begin{bmatrix}\sigma_1\\ \sigma_2\\ \sigma_6\end{bmatrix} = \sum_{i=0}^{n} \begin{bmatrix}\bar{Q_{11}} & \bar{Q_{12}} & \bar{Q_{16}}\\ \bar{Q_{12}} & \bar{Q_{22}} & \bar{Q_{26}}\\ \bar{Q_{16}} & \bar{Q_{26}} & \bar{Q_{66}}\end{bmatrix} \begin{bmatrix}\epsilon_1\\ \epsilon_2\\ \epsilon_6\end{bmatrix} \tag{1}$$

For transverse we get

$$\begin{bmatrix}\sigma_4\\ \sigma_5\end{bmatrix} = \sum_{i=0}^{n} \begin{bmatrix}\bar{Q_{55}} & \bar{Q_{56}}\\ \bar{Q_{65}} & \bar{Q_{66}}\end{bmatrix} \begin{bmatrix}\epsilon_4\\ \epsilon_5\end{bmatrix} \tag{2}$$

We apply reduced order integration to transverse shear to account for shear locking effect and less computational cost.

The stress are integrated with thickness i.e

$$\int_{-h/2}^{h/2} \sigma dz = \sigma z|_{-h/2}^{h/2} \tag{3}$$

which has been implemented in the code. From this the displacements are integrated with thickness which was taught in Analysis and Design of Composites.

1.1.1 Boundary conditions

As per the paper [7] the boundary conditions are Legendre polynomial which has artificial springs and rotational springs to take care of rigid body motions along with the method of elimination of variables which is often used in FEM.
To give appropriate deformation in the tips and convergence of displacement reduced order integration was done at the tips of the wing in artificial stiffness.

2 Prediction of coefficients using Airfoil coordinates [5]

As per the paper [4] coordinates of along z in the x-z plane of airfoil was done by using inverse airfoil method of predicting Berstein Coefficients using nonlinear least square method which is available in the MATLAB inbuilt function "lsqcurvefit".
The inverse method of predicting the coefficients are followed from [5]. Prediction are given as follows in the following figures 1 and 2.
Using the coefficients we can use the values to predict the z coordinates to use it in the program of calculating the thickness variation of the skin,spar and rib as per the formulation. [4].

3 Materials used in Skin,Spar,Rib of a wing

As explained in the introduction we can use any composite material at any angle hence for the test case for convergence we will be using.
The material property has been taken from [1].
The values are $E_1 = 70GPa$, $E_2 = 70GPa$, $G_6 = 5GPa$.
Using the Material values we get that the values of transverse modulus isn't given but since we are using reduced order integration we can use $G_5 = G_6$ and $G_4 = G_6$.
The Poisson effect has been taken in in plane constitutive model.
The material is throughout carbon fibre for skin,spar,ribs but can be changed appropriately to suit the user's requirements to calculate any form of displacement for

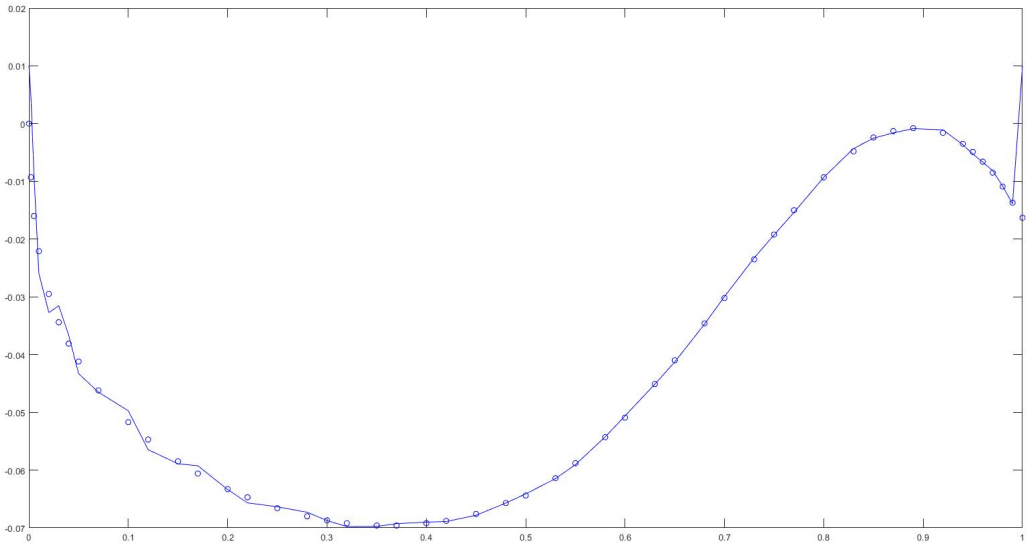


Figure 1: Predicting the lower surface of airfoil

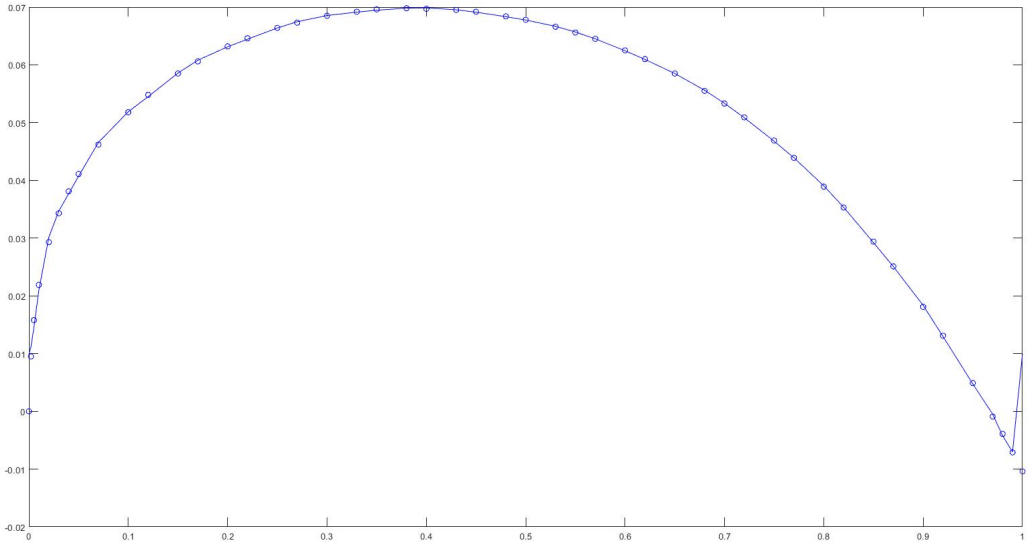


Figure 2: Predicting the upper surface of airfoil

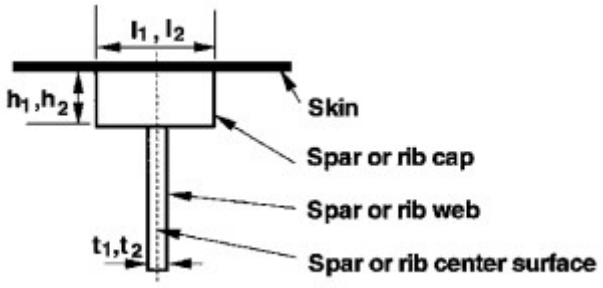


Figure 3: Dimensions of spar or rib [4]

a Load/Area force or Moment/Area.

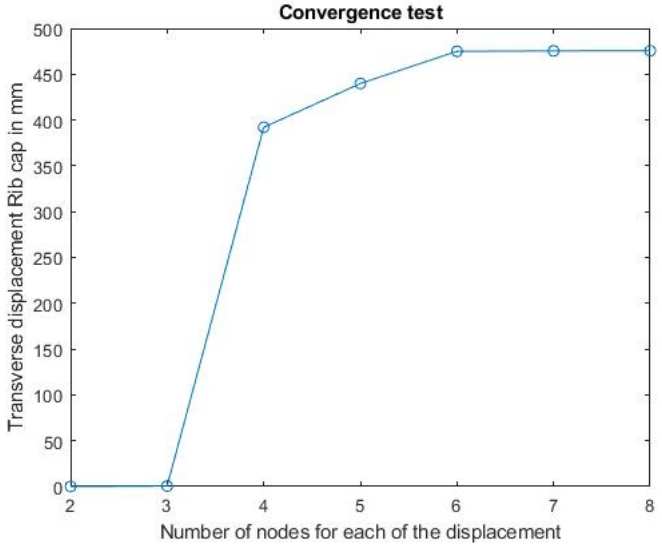
Drawback of taking account of transverse material coefficients

The lamina can't be rotated by an angle of 45 degrees in any of the quadrant since it leads to singular matrix while mapping from strains to displacements which results in high stiffness values.

4 Results

Using above material convergence test and the values were obtained for a geometric property as follows.

- Leading edge Sweep angle= 30 degrees
- Span of the entire wing =13.56 m
- Root chord = 6 m
- Aspect ratio = 3.3541 (calculated from the point 1 and 2)
- Airfoil used in the code is [3] which is supercritical airfoil which satisfies the aerodynamic requirements.
- Weight used is F-22 Raptor aircraft which is available in [2] which is a supersonic aircraft 38000 kg.
- The weight is divided by area too give uniform distributed force of 0.0277 N/mm^2.
- Number of lamina for this test is 4 which is $[0\,90]_s$ each with carbon fibre.
- Thickness is assumed to be same of 0.001m or 1mm.
- Dimensions of spar and rib are 1000 mm each as shown in figure 3.



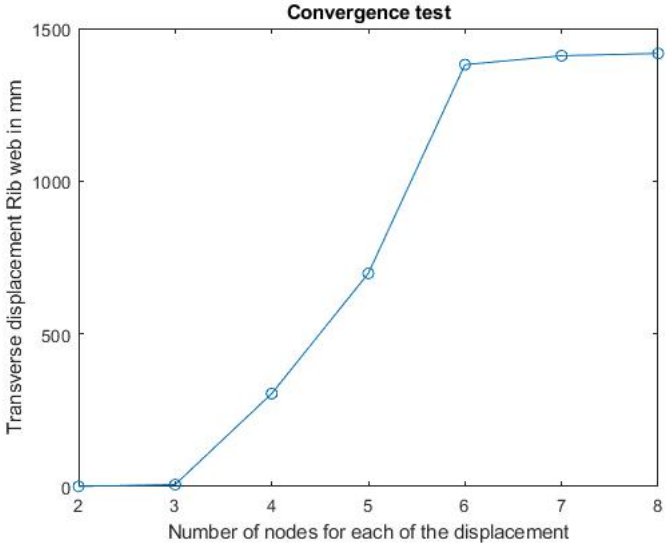

```
>> tolerance=(Final_converge(end)-Final_converge(end-1))/Final_converge(end)

tolerance =

   4.9337e-04
```

Figure 4: Convergence test of maximum transverse displacement

- As the paper [4] the number of nodes was from 2 to 8 for each ribs(web and cap),spar(web and cap),skin(web and cap).

Using the input parameters the convergence test and the value of transverse displacement at skin,spar(spar cap and web),ribs (rib cap and rib web) 4,5 ,6,7,8,9.

- The displacements are reasonable considering the effect of meters in geometric dimensions.
- Can be used for any geometric dimension, material and angle of each ply and thickness of each ply.
- Run time for each section of the code is around 2-3 minutes which makes it run faster than commercial packages which has a lot of degrees of freedom.
- Tolerance is 1e-03 is makes it convenient to use it commercially.
- Can be extrapolated to find stress vs strain graph,failure stress,strain.




```
>> tolerance=(Final_converge(end)-Final_converge(end-1))/Final_converge(end)

tolerance =

    0.0017
```

Figure 5: Convergence test of maximum transverse displacement

References

[1] Carbon fibre material property. `http://www.performance-composites.com/carbonfibre/mechanicalproperties_2.asp`.

[2] F-22 raptor weight. `https://www.af.mil/About-Us/Fact-Sheets/Display/Article/104506/f-22-raptor/`.

[3] Sc(2)-0714 supercritical airfoil nasasc2-0714-il. `http://airfoiltools.com/airfoil/details?airfoil=nasasc2-0714-il`.

[4] Rakesh K. Kapania and Youhua Liu. Static and vibration analyses of general wing structures using equivalent-plate models. *AIAA Journal*, 38(7):1269–1277, 2000.

[5] Brenda M. Kulfan. Universal parametric geometry representation method. *Journal of Aircraft*, 45(1):142–158, 2008.

[6] Eli Livne. Equivalent plate structural modeling for wing shape optimization including transverse shear. *AIAA Journal*, 32(6):1278–1288, 1994.

[7] Andrew Elwyn Lovejoy. *Natural frequencies and an atlas of mode shapes for generally-laminated, thick, skew, trapezoidal plates*. PhD thesis, Virginia Tech, 1994.

5 Appendix

5.1 Main Codes

```
clc
final_result=zeros(1,5);
gauss_points_in_x=[-1/sqrt(3) 1/sqrt(3)];
gauss_points_in_z=[-1/sqrt(3) 1/sqrt(3)];
 gauss_points_trans_x=0;
 gauss_points_in_y=[-1/sqrt(3) 1/sqrt(3)];
gauss_points_trans_y=0;
weights_in_x=[1 1];
 weights_in_y=[1 1];
weights_trans_x=2;
 weights_trans_y=2;
sweep=30;%degree
 b=13.56*10^3;%scaled to meter
 cr=6*10^3;%scaled to meter
 sweep_rad=30*(pi/180);
 b=b/2;
 tip=cr-tan(sweep_rad)*(b/2);
 Aspect_ratio=(b^2)/(0.5*b*0.5*(cr+tip));
 Area=0.5*b*0.5*(cr+tip);
 weight=38000*10;
 theta=[0 90 90 0];
 thickness_of_each_ply=0.001*ones(1,length(theta));
l_1=1000;
 c=1000;
 h=1000;
  h_1=1000;
  t_1=1000;
  h_2=1000;
  t_2=1000;
 force_local=[0;0;weight/Area;0;0];
 Final_result=0;
 Final_converge=zeros(1,5);
 iter=0;
%  force_vector_final=zeros(rep*rep+rep*rep+rep*rep+rep*rep+rep*S,1);
po=1;
tic
for boom=2:8
    for po=1:length(thickness_of_each_ply)
 I=boom;J=boom;K=boom;L=boom;
 M=boom;N=boom;P=boom;Q=boom;R=boom;S=boom;%skin
 I_spar=boom;J_spar=boom;K_spar=boom;L_spar=boom;
 M_spar=boom;N_spar=boom;P_spar=boom;Q_spar=boom;R_spar=boom;S_spar=boom;%spar
 I_rib=boom;J_rib=boom;K_rib=boom;L_rib=boom;
 M_rib=boom;N_rib=boom;P_rib=boom;Q_rib=boom;R_rib=boom;S_rib=boom;%ribs
 %Stiffness_of_skin=FEM_Main_file_skin(gauss_points_in_x,gauss_points_in_y,gauss_p
  %Stiffness_of_spars_web=FEM_Main_file_spars_web(gauss_points_in_x,gauss_points_i
%
 Stiffness_of_spars_cap=FEM_Main_file_spars_cap(gauss_points_in_x,gauss_points_in_
%
 Stiffness_of_rib_web=FEM_Main_file_rib_web(gauss_points_in_x,gauss_points_in_y,ga

 Stiffness_of_rib_cap=FEM_Main_file_ribs_cap(gauss_points_in_x,gauss_points_in_y,g
```

```
 Stiffness_of_bc=Artificial_Stiffness(I,J,K,L,M,N,P,Q,R,S,sweep,b,cr,gauss_points_
 Final_Stiffness=Stiffness_of_bc+Stiffness_of_rib_cap;%
+Stiffness_of_rib_cap+Stiffness_of_rib_web+Stiffness_of_spars_cap
+Stiffness_of_spars_web;Stiffness_of_skin
 force_local=[0;0;(5.2079);0;0];
 force_vector_final=force_vector(sweep,cr,b,I,J,K,L,M,N,P,Q,R,S,gauss_points_in_x,
 bc_u=linspace(1,I*J,I*J);%,I*J+K*L+M*N+P*Q+R*S];
 bc_v=linspace(I*J+1,I*J+K*L,I*J);
 bc_w=I*J+K*L+2:2:I*J+K*L+M*N-1;
 bc_theta_x=linspace(I*J+K*L+M*N+1,I*J+K*L+M*N+P*Q,I*J);
 bc_theta_y=linspace(I*J+K*L+M*N+P*Q+1,I*J+K*L+M*N+P*Q+R*S,I*J);
ibc=0;
for i=1:length(bc_u)

force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_u(i))*ibc);
  Final_Stiffness(:,bc_u(i))=0;
  Final_Stiffness(bc_u(i),:)=0;
  Final_Stiffness(bc_u(i),bc_u(i))=1;
  force_vector_final(bc_u(i),1)=ibc;
end
for i=1:length(bc_v)
force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_v(i))*ibc);
  Final_Stiffness(:,bc_v(i))=0;
  Final_Stiffness(bc_v(i),:)=0;
  Final_Stiffness(bc_v(i),bc_v(i))=1;
  force_vector_final(bc_v(i),1)=ibc;
end
for i=1:length(bc_w)
force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_w(i))*ibc);
  Final_Stiffness(:,bc_w(i))=0;
  Final_Stiffness(bc_w(i),:)=0;
  Final_Stiffness(bc_w(i),bc_w(i))=1;
  force_vector_final(bc_w(i),1)=ibc;
end
for i=1:length(bc_theta_x)
force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_theta_x(i))*ibc);
  Final_Stiffness(:,bc_theta_x(i))=0;
  Final_Stiffness(bc_theta_x(i),:)=0;
  Final_Stiffness(bc_theta_x(i),bc_theta_x(i))=1;
  force_vector_final(bc_theta_x(i),1)=ibc;
end
for i=1:length(bc_theta_y)
force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_theta_y(i))*ibc);
  Final_Stiffness(:,bc_theta_y(i))=0;
  Final_Stiffness(bc_theta_y(i),:)=0;
  Final_Stiffness(bc_theta_y(i),bc_theta_y(i))=1;
  force_vector_final(bc_theta_y(i),1)=ibc;
end
```

```
 goal=det(Final_Stiffness);
 if goal==0
     disp("Fail")
     break
 else
     iter=iter+1;
     disp("Success")

%%bc
 final_disp=pinv(Final_Stiffness)*force_vector_final;
 values=find(final_disp>=0.001);
 convergence=final_disp(values);
 values_transverse=find(force_vector_final>0);
 end
 final_result(po)=max(final_disp(values_transverse));
 for i=1:length(gauss_points_in_x)
     Final_result=Final_result
+final_result(po)*(sum(thickness_of_each_ply)/2)*weights_in_x(i);
 end
    end
     if goal==0
     disp("Fail")
     break
     end
Final_converge(boom)=Final_result;
end
toc
plot(2:boom,Final_converge(2:boom),"-o");
xlabel("Number of nodes for each of the displacement")
ylabel("Transverse displacement Rib cap in mm")
title("Convergence test")
```

Local minimum found.

Optimization completed because the size of the gradient is less than the value of the optimality tolerance.

Local minimum found.

Optimization completed because the size of the gradient is less than the value of the optimality tolerance.

Local minimum found.

Optimization completed because the size of the gradient is less than the value of the optimality tolerance.

Local minimum found.

Optimization completed because the size of the gradient is less than

```
clc
final_result=zeros(1,5);
gauss_points_in_x=[-1/sqrt(3) 1/sqrt(3)];
gauss_points_in_z=[-1/sqrt(3) 1/sqrt(3)];
 gauss_points_trans_x=0;
 gauss_points_in_y=[-1/sqrt(3) 1/sqrt(3)];
gauss_points_trans_y=0;
weights_in_x=[1 1];
 weights_in_y=[1 1];
weights_trans_x=2;
 weights_trans_y=2;
sweep=30;%degree
 b=13.56*10^3;%scaled to meter
 cr=6*10^3;%scaled to meter
 sweep_rad=30*(pi/180);
 b=b/2;
 tip=cr-tan(sweep_rad)*(b/2);
 Aspect_ratio=(b^2)/(0.5*b*0.5*(cr+tip));
 Area=0.5*b*0.5*(cr+tip);
 weight=38000*10;
 theta=[0 90 90 0];
 thickness_of_each_ply=0.001*ones(1,length(theta));
l_1=1000;
 c=1000;
 h=1000;
  h_1=1000;
  t_1=1000;
  h_2=1000;
  t_2=1000;
 force_local=[0;0;weight/Area;0;0];
 Final_result=0;
 Final_converge=zeros(1,5);
 iter=0;
%  force_vector_final=zeros(rep*rep+rep*rep+rep*rep+rep*rep+rep*S,1);
po=1;
tic
for boom=2:8
    for po=1:length(thickness_of_each_ply)
 I=boom;J=boom;K=boom;L=boom;
 M=boom;N=boom;P=boom;Q=boom;R=boom;S=boom;%skin
 I_spar=boom;J_spar=boom;K_spar=boom;L_spar=boom;
 M_spar=boom;N_spar=boom;P_spar=boom;Q_spar=boom;R_spar=boom;S_spar=boom;%spar
 I_rib=boom;J_rib=boom;K_rib=boom;L_rib=boom;
 M_rib=boom;N_rib=boom;P_rib=boom;Q_rib=boom;R_rib=boom;S_rib=boom;%ribs
 %Stiffness_of_skin=FEM_Main_file_skin(gauss_points_in_x,gauss_points_in_y,gauss_p
 %
 Stiffness_of_spars_web=FEM_Main_file_spars_web(gauss_points_in_x,gauss_points_in_
%
 Stiffness_of_spars_cap=FEM_Main_file_spars_cap(gauss_points_in_x,gauss_points_in_

 Stiffness_of_rib_web=FEM_Main_file_rib_web(gauss_points_in_x,gauss_points_in_y,ga
```

```
%
 Stiffness_of_rib_cap=FEM_Main_file_ribs_cap(gauss_points_in_x,gauss_points_in_y,g

 Stiffness_of_bc=Artificial_Stiffness(I,J,K,L,M,N,P,Q,R,S,sweep,b,cr,gauss_points_
 Final_Stiffness=Stiffness_of_bc+Stiffness_of_rib_web;%
+Stiffness_of_rib_cap+Stiffness_of_rib_web+Stiffness_of_spars_cap
+Stiffness_of_spars_web;Stiffness_of_skin
 force_local=[0;0;(5.2079);0;0];
 force_vector_final=force_vector(sweep,cr,b,I,J,K,L,M,N,P,Q,R,S,gauss_points_in_x,
 bc_u=linspace(1,I*J,I*J);%,I*J+K*L+M*N+P*Q+R*S];
 bc_v=linspace(I*J+1,I*J+K*L,I*J);
 bc_w=I*J+K*L+2:2:I*J+K*L+M*N-1;
 bc_theta_x=linspace(I*J+K*L+M*N+1,I*J+K*L+M*N+P*Q,I*J);
 bc_theta_y=linspace(I*J+K*L+M*N+P*Q+1,I*J+K*L+M*N+P*Q+R*S,I*J);
ibc=0;
for i=1:length(bc_u)

force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_u(i))*ibc);
  Final_Stiffness(:,bc_u(i))=0;
  Final_Stiffness(bc_u(i),:)=0;
  Final_Stiffness(bc_u(i),bc_u(i))=1;
  force_vector_final(bc_u(i),1)=ibc;
end
for i=1:length(bc_v)
force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_v(i))*ibc);
  Final_Stiffness(:,bc_v(i))=0;
  Final_Stiffness(bc_v(i),:)=0;
  Final_Stiffness(bc_v(i),bc_v(i))=1;
  force_vector_final(bc_v(i),1)=ibc;
end
for i=1:length(bc_w)
force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_w(i))*ibc);
  Final_Stiffness(:,bc_w(i))=0;
  Final_Stiffness(bc_w(i),:)=0;
  Final_Stiffness(bc_w(i),bc_w(i))=1;
  force_vector_final(bc_w(i),1)=ibc;
end
for i=1:length(bc_theta_x)
force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_theta_x(i))*ibc);
  Final_Stiffness(:,bc_theta_x(i))=0;
  Final_Stiffness(bc_theta_x(i),:)=0;
  Final_Stiffness(bc_theta_x(i),bc_theta_x(i))=1;
  force_vector_final(bc_theta_x(i),1)=ibc;
end
for i=1:length(bc_theta_y)
force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_theta_y(i))*ibc);
  Final_Stiffness(:,bc_theta_y(i))=0;
  Final_Stiffness(bc_theta_y(i),:)=0;
  Final_Stiffness(bc_theta_y(i),bc_theta_y(i))=1;
```

```
  force_vector_final(bc_theta_y(i),1)=ibc;
end
 goal=det(Final_Stiffness);
 if goal==0
     disp("Fail")
     break
 else
     iter=iter+1;
     disp("Success")

%%bc
 final_disp=pinv(Final_Stiffness)*force_vector_final;
 values=find(final_disp>=0.001);
 convergence=final_disp(values);
 values_transverse=find(force_vector_final>0);
 end
 final_result(po)=max(final_disp(values_transverse));
 for i=1:length(gauss_points_in_x)
     Final_result=Final_result
+final_result(po)*(sum(thickness_of_each_ply)/2)*weights_in_x(i);
 end
    end
     if goal==0
     disp("Fail")
     break
     end
Final_converge(boom)=Final_result;
end
toc
plot(2:boom,Final_converge(2:boom),"-o");
xlabel("Number of nodes for each of the displacement")
ylabel("Transverse displacement Rib web in mm")
title("Convergence test")
```

Local minimum found.

Optimization completed because the size of the gradient is less than the value of the optimality tolerance.

Local minimum found.

Optimization completed because the size of the gradient is less than the value of the optimality tolerance.

Local minimum found.

Optimization completed because the size of the gradient is less than the value of the optimality tolerance.

Local minimum found.

```
clc
final_result=zeros(1,5);
gauss_points_in_x=[-1/sqrt(3) 1/sqrt(3)];
gauss_points_in_z=[-1/sqrt(3) 1/sqrt(3)];
 gauss_points_trans_x=0;
 gauss_points_in_y=[-1/sqrt(3) 1/sqrt(3)];
gauss_points_trans_y=0;
weights_in_x=[1 1];
 weights_in_y=[1 1];
weights_trans_x=2;
 weights_trans_y=2;
sweep=30;%degree
 b=13.56*10^3;%scaled to millimeter
 cr=6*10^3;%scaled to millimeter
 sweep_rad=30*(pi/180);
 b=b/2;
 tip=cr-tan(sweep_rad)*(b/2);
 Aspect_ratio=2*((b/2)^2)/(0.5*b*0.5*(cr+tip));
 Area=0.5*b*0.5*(cr+tip);
 weight=38000*10;
 theta=[0 90 90 0];
 thickness_of_each_ply=0.001*ones(1,length(theta));
l_1=1000;
 c=1000;
 h=1000;
  h_1=1000;
  t_1=1000;
  h_2=1000;
  t_2=1000;
  force_local=[0;0;weight/Area;0;0];
 Final_result=0;
 Final_converge=zeros(1,5);
 iter=0;
%  force_vector_final=zeros(rep*rep+rep*rep+rep*rep+rep*rep+rep*S,1);
po=1;
tic
for boom=2:8
    for po=1:length(thickness_of_each_ply)
 I=boom;J=boom;K=boom;L=boom;
 M=boom;N=boom;P=boom;Q=boom;R=boom;S=boom;%skin
 I_spar=boom;J_spar=boom;K_spar=boom;L_spar=boom;
 M_spar=boom;N_spar=boom;P_spar=boom;Q_spar=boom;R_spar=boom;S_spar=boom;%spar
 I_rib=boom;J_rib=boom;K_rib=boom;L_rib=boom;
 M_rib=boom;N_rib=boom;P_rib=boom;Q_rib=boom;R_rib=boom;S_rib=boom;%ribs
 Stiffness_of_skin=FEM_Main_file_skin(gauss_points_in_x,gauss_points_in_y,gauss_po
%
 Stiffness_of_spars_web=FEM_Main_file_spars_web(gauss_points_in_x,gauss_points_in_
%
 Stiffness_of_spars_cap=FEM_Main_file_spars_cap(gauss_points_in_x,gauss_points_in_
%
 Stiffness_of_rib_web=FEM_Main_file_rib_web(gauss_points_in_x,gauss_points_in_y,ga
```

```
%
 Stiffness_of_rib_cap=FEM_Main_file_ribs_cap(gauss_points_in_x,gauss_points_in_y,g

 Stiffness_of_bc=Artificial_Stiffness(I,J,K,L,M,N,P,Q,R,S,sweep,b,cr,gauss_points_
 Final_Stiffness=Stiffness_of_bc+Stiffness_of_skin;%
+Stiffness_of_rib_cap+Stiffness_of_rib_web+Stiffness_of_spars_cap
+Stiffness_of_spars_web;
%  force_local=[0;0;(0.52079);0;0];
 force_vector_final=force_vector(sweep,cr,b,I,J,K,L,M,N,P,Q,R,S,gauss_points_in_x,
 bc_u=linspace(1,I*J,I*J);%,I*J+K*L+M*N+P*Q+R*S];
 bc_v=linspace(I*J+1,I*J+K*L,I*J);
 bc_w=I*J+K*L+2:2:I*J+K*L+M*N-1;
 bc_theta_x=linspace(I*J+K*L+M*N+1,I*J+K*L+M*N+P*Q,I*J);
 bc_theta_y=linspace(I*J+K*L+M*N+P*Q+1,I*J+K*L+M*N+P*Q+R*S,I*J);
ibc=0;
for i=1:length(bc_u)

force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_u(i))*ibc);
  Final_Stiffness(:,bc_u(i))=0;
  Final_Stiffness(bc_u(i),:)=0;
  Final_Stiffness(bc_u(i),bc_u(i))=1;
  force_vector_final(bc_u(i),1)=ibc;
end
for i=1:length(bc_v)
force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_v(i))*ibc);
  Final_Stiffness(:,bc_v(i))=0;
  Final_Stiffness(bc_v(i),:)=0;
  Final_Stiffness(bc_v(i),bc_v(i))=1;
  force_vector_final(bc_v(i),1)=ibc;
end
for i=1:length(bc_w)
force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_w(i))*ibc);
  Final_Stiffness(:,bc_w(i))=0;
  Final_Stiffness(bc_w(i),:)=0;
  Final_Stiffness(bc_w(i),bc_w(i))=1;
  force_vector_final(bc_w(i),1)=ibc;
end
for i=1:length(bc_theta_x)
force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_theta_x(i))*ibc);
  Final_Stiffness(:,bc_theta_x(i))=0;
  Final_Stiffness(bc_theta_x(i),:)=0;
  Final_Stiffness(bc_theta_x(i),bc_theta_x(i))=1;
  force_vector_final(bc_theta_x(i),1)=ibc;
end
for i=1:length(bc_theta_y)
force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_theta_y(i))*ibc);
  Final_Stiffness(:,bc_theta_y(i))=0;
  Final_Stiffness(bc_theta_y(i),:)=0;
  Final_Stiffness(bc_theta_y(i),bc_theta_y(i))=1;
```

```
  force_vector_final(bc_theta_y(i),1)=ibc;
end
 goal=det(Final_Stiffness);
 if goal==0
     disp("Fail")
     break
 else
     iter=iter+1;
     disp("Success")

%%bc
 final_disp=pinv(Final_Stiffness)*force_vector_final;
 values=find(final_disp>=0.001);
 convergence=final_disp(values);
 values_transverse=find(force_vector_final>0);
 end
 final_result(po)=max(final_disp(values_transverse));
 for i=1:length(gauss_points_in_x)
     Final_result=Final_result
+final_result(po)*(sum(thickness_of_each_ply)/2)*weights_in_x(i);
 end
    end
     if goal==0
     disp("Fail")
     break
     end
Final_converge(boom)=Final_result;
end
toc
plot(2:boom,Final_converge(2:boom),"-o");
xlabel("Number of nodes for each of the displacement")
ylabel("Transverse displacement skin in mm")
title("Convergence test")
```

```
Local minimum found.

Optimization completed because the size of the gradient is less than
the value of the optimality tolerance.


Local minimum found.

Optimization completed because the size of the gradient is less than
the value of the optimality tolerance.


Local minimum found.

Optimization completed because the size of the gradient is less than
the value of the optimality tolerance.


Local minimum found.
```

```
clc
final_result=zeros(1,5);
gauss_points_in_x=[-1/sqrt(3) 1/sqrt(3)];
gauss_points_in_z=[-1/sqrt(3) 1/sqrt(3)];
 gauss_points_trans_x=0;
 gauss_points_in_y=[-1/sqrt(3) 1/sqrt(3)];
gauss_points_trans_y=0;
weights_in_x=[1 1];
 weights_in_y=[1 1];
weights_trans_x=2;
 weights_trans_y=2;
sweep_angle=30;%degree
 b=13.56*10^3;%scaled to millimeter
 cr=6*10^3;%scaled to millimeter
 sweep_rad=30*(pi/180);
 b=b/2;
 tip=cr-tan(sweep_rad)*(b/2);
 Aspect_ratio=(b^2)/(0.5*b*0.5*(cr+tip));
 Area=0.5*b*0.5*(cr+tip);
 weight=38000*10;
 theta=[0 90 90 0];
 thickness_of_each_ply=0.001*ones(1,length(theta));
l_1=1000;
 c=1000;
 h=1000;
  h_1=1000;
  t_1=1000;
  h_2=1000;
  t_2=1000;
 force_local=[0;0;weight/Area;0;0];
 Final_result=0;
 Final_converge=zeros(1,5);
 iter=0;
%  force_vector_final=zeros(rep*rep+rep*rep+rep*rep+rep*rep+rep*S,1);
po=1;
tic
for boom=2:8
    for po=1:length(thickness_of_each_ply)
 I=boom;J=boom;K=boom;L=boom;
 M=boom;N=boom;P=boom;Q=boom;R=boom;S=boom;%skin
 I_spar=boom;J_spar=boom;K_spar=boom;L_spar=boom;
 M_spar=boom;N_spar=boom;P_spar=boom;Q_spar=boom;R_spar=boom;S_spar=boom;%spar
 I_rib=boom;J_rib=boom;K_rib=boom;L_rib=boom;
 M_rib=boom;N_rib=boom;P_rib=boom;Q_rib=boom;R_rib=boom;S_rib=boom;%ribs
 %Stiffness_of_skin=FEM_Main_file_skin(gauss_points_in_x,gauss_points_in_y,gauss_p
 %
 Stiffness_of_spars_web=FEM_Main_file_spars_web(gauss_points_in_x,gauss_points_in_
 Stiffness_of_spars_cap=FEM_Main_file_spars_cap(gauss_points_in_x,gauss_points_in_
%
 Stiffness_of_rib_web=FEM_Main_file_rib_web(gauss_points_in_x,gauss_points_in_y,ga
%
 Stiffness_of_rib_cap=FEM_Main_file_ribs_cap(gauss_points_in_x,gauss_points_in_y,g
```

```
 Stiffness_of_bc=Artificial_Stiffness(I,J,K,L,M,N,P,Q,R,S,sweep_angle,b,cr,gauss_p
 Final_Stiffness=Stiffness_of_bc+Stiffness_of_spars_cap;%
+Stiffness_of_rib_cap+Stiffness_of_rib_web+Stiffness_of_spars_cap
+Stiffness_of_spars_web;Stiffness_of_skin
 force_local=[0;0;(5.2079);0;0];
 force_vector_final=force_vector(sweep_angle,cr,b,I,J,K,L,M,N,P,Q,R,S,gauss_points
 bc_u=linspace(1,I*J,I*J);%,I*J+K*L+M*N+P*Q+R*S];
 bc_v=linspace(I*J+1,I*J+K*L,I*J);
 bc_w=I*J+K*L+2:2:I*J+K*L+M*N-1;
 bc_theta_x=linspace(I*J+K*L+M*N+1,I*J+K*L+M*N+P*Q,I*J);
 bc_theta_y=linspace(I*J+K*L+M*N+P*Q+1,I*J+K*L+M*N+P*Q+R*S,I*J);
ibc=0;
for i=1:length(bc_u)

force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_u(i))*ibc);
  Final_Stiffness(:,bc_u(i))=0;
  Final_Stiffness(bc_u(i),:)=0;
  Final_Stiffness(bc_u(i),bc_u(i))=1;
  force_vector_final(bc_u(i),1)=ibc;
end
for i=1:length(bc_v)
force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_v(i))*ibc);
  Final_Stiffness(:,bc_v(i))=0;
  Final_Stiffness(bc_v(i),:)=0;
  Final_Stiffness(bc_v(i),bc_v(i))=1;
  force_vector_final(bc_v(i),1)=ibc;
end
for i=1:length(bc_w)
force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_w(i))*ibc);
  Final_Stiffness(:,bc_w(i))=0;
  Final_Stiffness(bc_w(i),:)=0;
  Final_Stiffness(bc_w(i),bc_w(i))=1;
  force_vector_final(bc_w(i),1)=ibc;
end
for i=1:length(bc_theta_x)
force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_theta_x(i))*ibc);
  Final_Stiffness(:,bc_theta_x(i))=0;
  Final_Stiffness(bc_theta_x(i),:)=0;
  Final_Stiffness(bc_theta_x(i),bc_theta_x(i))=1;
  force_vector_final(bc_theta_x(i),1)=ibc;
end
for i=1:length(bc_theta_y)
force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_theta_y(i))*ibc);
  Final_Stiffness(:,bc_theta_y(i))=0;
  Final_Stiffness(bc_theta_y(i),:)=0;
  Final_Stiffness(bc_theta_y(i),bc_theta_y(i))=1;
  force_vector_final(bc_theta_y(i),1)=ibc;
end
```

```
 goal=det(Final_Stiffness);
 if goal==0
     disp("Fail")
     break
 else
     iter=iter+1;
     disp("Success")

%%bc
 final_disp=pinv(Final_Stiffness)*force_vector_final;
 values=find(final_disp>=0.001);
 convergence=final_disp(values);
 values_transverse=find(force_vector_final>0);
 end
 final_result(po)=max(final_disp(values_transverse));
 for i=1:length(gauss_points_in_x)
     Final_result=Final_result
+final_result(po)*(sum(thickness_of_each_ply)/2)*weights_in_x(i);
 end
    end
     if goal==0
     disp("Fail")
     break
     end
Final_converge(boom)=Final_result;
end
toc
plot(2:boom,Final_converge(2:boom),"-o");
xlabel("Number of nodes for each of the displacement")
ylabel("Transverse displacement spar cap in mm")
title("Convergence test")
```

Local minimum found.

Optimization completed because the size of the gradient is less than the value of the optimality tolerance.

Local minimum found.

Optimization completed because the size of the gradient is less than the value of the optimality tolerance.

Local minimum found.

Optimization completed because the size of the gradient is less than the value of the optimality tolerance.

Local minimum found.

Optimization completed because the size of the gradient is less than

```
clc
final_result=zeros(1,5);
gauss_points_in_x=[-1/sqrt(3) 1/sqrt(3)];
gauss_points_in_z=[-1/sqrt(3) 1/sqrt(3)];
 gauss_points_trans_x=0;
 gauss_points_in_y=[-1/sqrt(3) 1/sqrt(3)];
gauss_points_trans_y=0;
weights_in_x=[1 1];
 weights_in_y=[1 1];
weights_trans_x=2;
 weights_trans_y=2;
sweep=30;%degree
 b=13.56*10^3;%scaled to meter
 cr=6*10^3;%scaled to meter
 sweep_rad=30*(pi/180);
 b=b/2;
 tip=cr-tan(sweep_rad)*(b/2);
 Aspect_ratio=(b^2)/(0.5*b*0.5*(cr+tip));
 Area=0.5*b*0.5*(cr+tip);
 weight=38000*10;
 theta=[0 90 90 0];
 thickness_of_each_ply=0.001*ones(1,length(theta));
l_1=1000;
 c=1000;
 h=1000;
  h_1=1000;
  t_1=1000;
  h_2=1000;
  t_2=1000;
 force_local=[0;0;weight/Area;0;0];
 Final_result=0;
 Final_converge=zeros(1,5);
 iter=0;
%  force_vector_final=zeros(rep*rep+rep*rep+rep*rep+rep*rep+rep*S,1);
po=1;
for boom=2:8
    for po=1:length(thickness_of_each_ply)
 I=boom;J=boom;K=boom;L=boom;
 M=boom;N=boom;P=boom;Q=boom;R=boom;S=boom;%skin
 I_spar=boom;J_spar=boom;K_spar=boom;L_spar=boom;
 M_spar=boom;N_spar=boom;P_spar=boom;Q_spar=boom;R_spar=boom;S_spar=boom;%spar
 I_rib=boom;J_rib=boom;K_rib=boom;L_rib=boom;
 M_rib=boom;N_rib=boom;P_rib=boom;Q_rib=boom;R_rib=boom;S_rib=boom;%ribs
 %Stiffness_of_skin=FEM_Main_file_skin(gauss_points_in_x,gauss_points_in_y,gauss_p

 Stiffness_of_spars_web=FEM_Main_file_spars_web(gauss_points_in_x,gauss_points_in_
%
 Stiffness_of_spars_cap=FEM_Main_file_spars_cap(gauss_points_in_x,gauss_points_in_
%
 Stiffness_of_rib_web=FEM_Main_file_rib_web(gauss_points_in_x,gauss_points_in_y,ga
%
 Stiffness_of_rib_cap=FEM_Main_file_ribs_cap(gauss_points_in_x,gauss_points_in_y,g
```

```
 Stiffness_of_bc=Artificial_Stiffness(I,J,K,L,M,N,P,Q,R,S,sweep,b,cr,gauss_points_
 Final_Stiffness=Stiffness_of_bc+Stiffness_of_spars_web;%
+Stiffness_of_rib_cap+Stiffness_of_rib_web+Stiffness_of_spars_cap
+Stiffness_of_spars_web;Stiffness_of_skin
 force_local=[0;0;(5.2079);0;0];
 force_vector_final=force_vector(sweep,cr,b,I,J,K,L,M,N,P,Q,R,S,gauss_points_in_x,
 bc_u=linspace(1,I*J,I*J);%,I*J+K*L+M*N+P*Q+R*S];
 bc_v=linspace(I*J+1,I*J+K*L,I*J);
 bc_w=I*J+K*L+2:2:I*J+K*L+M*N-1;
 bc_theta_x=linspace(I*J+K*L+M*N+1,I*J+K*L+M*N+P*Q,I*J);
 bc_theta_y=linspace(I*J+K*L+M*N+P*Q+1,I*J+K*L+M*N+P*Q+R*S,I*J);
ibc=0;
for i=1:length(bc_u)

force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_u(i))*ibc);
  Final_Stiffness(:,bc_u(i))=0;
  Final_Stiffness(bc_u(i),:)=0;
  Final_Stiffness(bc_u(i),bc_u(i))=1;
  force_vector_final(bc_u(i),1)=ibc;
end
for i=1:length(bc_v)
force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_v(i))*ibc);
  Final_Stiffness(:,bc_v(i))=0;
  Final_Stiffness(bc_v(i),:)=0;
  Final_Stiffness(bc_v(i),bc_v(i))=1;
  force_vector_final(bc_v(i),1)=ibc;
end
for i=1:length(bc_w)
force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_w(i))*ibc);
  Final_Stiffness(:,bc_w(i))=0;
  Final_Stiffness(bc_w(i),:)=0;
  Final_Stiffness(bc_w(i),bc_w(i))=1;
  force_vector_final(bc_w(i),1)=ibc;
end
for i=1:length(bc_theta_x)
force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_theta_x(i))*ibc);
  Final_Stiffness(:,bc_theta_x(i))=0;
  Final_Stiffness(bc_theta_x(i),:)=0;
  Final_Stiffness(bc_theta_x(i),bc_theta_x(i))=1;
  force_vector_final(bc_theta_x(i),1)=ibc;
end
for i=1:length(bc_theta_y)
force_vector_final=force_vector_final-
(Final_Stiffness(:,bc_theta_y(i))*ibc);
  Final_Stiffness(:,bc_theta_y(i))=0;
  Final_Stiffness(bc_theta_y(i),:)=0;
  Final_Stiffness(bc_theta_y(i),bc_theta_y(i))=1;
  force_vector_final(bc_theta_y(i),1)=ibc;
end
```

```
 goal=det(Final_Stiffness);
 if goal==0
     disp("Fail")
     break
 else
     iter=iter+1;
     disp("Success")

%%bc
 final_disp=pinv(Final_Stiffness)*force_vector_final;
 values=find(final_disp>=0.001);
 convergence=final_disp(values);
 values_transverse=find(force_vector_final>0);
 end
 final_result(po)=max(final_disp(values_transverse));
 for i=1:length(gauss_points_in_x)
     Final_result=Final_result
+final_result(po)*(sum(thickness_of_each_ply)/2)*weights_in_x(i);
 end
    end
     if goal==0
     disp("Fail")
     break
     end
Final_converge(boom)=Final_result;
end
plot(2:boom,Final_converge(2:boom),"-o");
xlabel("Number of nodes for each of the displacement")
ylabel("Transverse displacement spar web in mm")
title("Convergence test")
```

```
Local minimum found.

Optimization completed because the size of the gradient is less than
the value of the optimality tolerance.


Local minimum found.

Optimization completed because the size of the gradient is less than
the value of the optimality tolerance.


Local minimum found.

Optimization completed because the size of the gradient is less than
the value of the optimality tolerance.


Local minimum found.

Optimization completed because the size of the gradient is less than
the value of the optimality tolerance.
```

5.2 Spar web

```
function stiffness=abd_matrix(theta_of_each_ply,~)
%theta_of_each_ply=[0 45 45 0];
%thickness_of_each_ply=ones(1,length(theta_of_each_ply));
% height_of_lamina=sum(thickness_of_each_ply);
% ords1=zeros(1,(length(thickness_of_each_ply)+1));
% ords1(1,1)=-height_of_lamina/2;
% for i=2:(length(thickness_of_each_ply))
%     ords1(1,i)=ords1(1,i-1)+thickness_of_each_ply(1,i);
% end
% ords1(1,end)=abs(ords1(1,1));
Q_transformed_1=zeros(3,3);
for i=1:length(theta_of_each_ply)

 Q_transformed=inv(ques_3_with_arguments_transformed_in_plane(theta_of_each_ply(i)
%     abd(1:3,1:3)=abd(1:3,1:3)+Q_transformed*((ords1(1,i+1))-
ords1(1,i));
%     abd(4:end,1:3)=abd(4:end,1:3)+0.5*Q_transformed*((ords1(1,i
+1))^2-ords1(1,i)^2);
%     abd(1:3,4:6)=abd(1:3,4:6)+0.5*Q_transformed*((ords1(1,i+1))^2-
ords1(1,i)^2);
%     abd(4:6,4:6)=abd(4:6,4:6)+(1/3)*Q_transformed*((ords1(1,i+1))^3-
ords1(1,i)^3);
Q_transformed_1=Q_transformed_1+Q_transformed;
end
% z_k=0;
% z_k_2=0;
% for i=1:length(theta_of_each_ply)
% z_k=z_k+(ords1(1,i+1)-ords1(1,i));
% z_k_2=z_k_2+0.5*((ords1(1,i+1))^2-(ords1(1,i))^2);
% end
% complaince=[1 0 0 1 0 0;0 1 0 0 1 0;0 0 1 0 0 1]*inv(abd)*[z_k
 0 0;0 z_k 0;0 0 z_k;(((z_k_2)^2)/2) 0 0;0 (((z_k_2)^2)/2) 0;0 0
 (((z_k_2)^2)/2)];
stiffness=inv(Q_transformed_1);
end
```

```
Not enough input arguments.

Error in abd_matrix (line 12)
for i=1:length(theta_of_each_ply)
```

Published with MATLAB® R2020a

```
function stiffness=abd_matrix_trans(theta_of_each_ply,~)
%theta_of_each_ply=[0 90 90 0];
%DISCLAIMNER 45 DEGREE OR -45 DEGREE LEADS TO SINGULARITY OR ITS
 FACTORS IN
%Q2 Q3 OR Q4
%thickness_of_each_ply=ones(1,length(theta_of_each_ply));
% height_of_lamina=sum(thickness_of_each_ply);
% ords1=zeros(1,(length(thickness_of_each_ply)+1));
% ords1(1,1)=-height_of_lamina/2;
% for i=2:(length(thickness_of_each_ply))
%     ords1(1,i)=ords1(1,i-1)+thickness_of_each_ply(1,i);
% end
% ords1(1,end)=abs(ords1(1,1));
Q_transformed_1=zeros(2,2);
for i=1:length(theta_of_each_ply)

 Q_transformed=inv(ques_3_with_arguments_transformed_trans_plane(theta_of_each_ply
%     abd(1:2,1:2)=abd(1:2,1:2)+Q_transformed*((ords1(1,i+1))-
ords1(1,i));
%     abd(3:end,1:2)=abd(3:end,1:2)+0.5*Q_transformed*((ords1(1,i
+1))^2-ords1(1,i)^2);
%     abd(1:2,3:4)=abd(1:2,3:4)+0.5*Q_transformed*((ords1(1,i+1))^2-
ords1(1,i)^2);
%     abd(3:4,3:4)=abd(3:4,3:4)+(1/3)*Q_transformed*((ords1(1,i+1))^3-
ords1(1,i)^3);
Q_transformed_1=Q_transformed_1+Q_transformed;
end
% z_k=0;
% z_k_2=0;
% for i=1:length(theta_of_each_ply)
% z_k=z_k+(ords1(1,i+1)-ords1(1,i));
% z_k_2=z_k_2+0.5*((ords1(1,i+1))^2-(ords1(1,i))^2);
% end
% complaince=[1 0 1 0;0 1 0 1]*inv(abd)*[z_k 0;0 z_k;(z_k_2) 0;0
 z_k_2];
stiffness=inv(Q_transformed_1);
end
```

```
Not enough input arguments.

Error in abd_matrix_trans (line 14)
for i=1:length(theta_of_each_ply)
```

Published with MATLAB® R2020a

```
function
 kg_in_plane=B_matrix_in_plane(gauss_points_x,gauss_points_y,sweep_angle,b,cr,I,J,
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
% weights_x=[1 1];
% weights_y=[1 1];
% sweep_angle=30;%degree
% b=10;
% l_1=1;
% c=1;
% h=1;
% h_1=1;
 pi=3.1416;
% t_1=1;
sweep_angle_in_radians=sweep_angle*pi/180;
% cr=5;
 tip_chord=cr-((b/2)*tan(sweep_angle_in_radians));
%Linear nodes per element
% I=3;
% J=3;
% K=3;
% L=3;
% M=3;
% N=3;
% P=3;
% Q=3;
% R=3;
% S=3;
% dof=(5);
% icon_IJ=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_KL=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_PQ=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_RS=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
coeff_of_z_lower=Lower_fitting_a_curve();
coeff_of_z_upper=Upper_fitting_a_curve();
number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
%number_of_elements=number_of_nodes-1;
P_non_derive_I_J_x=zeros(1,I*J);
P_derive_I_J_x=zeros(1,I*J);
P_derive_K_L_x=zeros(1,K*L);
P_non_derive_K_L_x=zeros(1,K*L);
P_derive_M_N_x=zeros(1,M*N);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
```

```
P_derive_I_J_y=zeros(1,I*J);
P_derive_K_L_y=zeros(1,K*L);
P_non_derive_K_L_y=zeros(1,K*L);
P_derive_M_N_y=zeros(1,M*N);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_derive_R_S_y=zeros(1,R*S);
B=zeros(12,(I*J+P*Q+M*N+K*L+R*S));
%  theta=[0 90 90 0];
%  thickness_of_each_ply=0.01*ones(1,length(theta));
hope=zeros((I*J+P*Q+M*N+K*L+R*S),(I*J+P*Q+M*N+K*L+R*S));
spar_position_I_J=linspace(-1,1,I*J);
spar_position_K_L=linspace(-1,1,K*L);
spar_position_P_Q=linspace(-1,1,P*Q);
spar_position_R_S=linspace(-1,1,R*S);
%kg=zeros(((I*J)+(K*L)+(P*Q)+(R*S))*dof,
((I*J)+(K*L)+(P*Q)+(R*S))*dof);
for
 i=1:I*J%*length(icon_KL(:,1))*length(icon_KL(1,:))*length(icon_PQ(:,1))*length(ic
%     n11_IJ=icon_IJ(i,1);
%     n22_IJ=icon_IJ(i,2);
%     n33_IJ=icon_IJ(i,3);
%     n44_IJ=icon_IJ(i,4);
%     %n55_IJ=icon_IJ(i,5);
%     n11_KL=icon_KL(i,1);
%     n22_KL=icon_KL(i,2);
%     n33_KL=icon_KL(i,3);
%     n44_KL=icon_KL(i,4);
%     %n55_KL=icon_KL(i,5);
%     n11_PQ=icon_PQ(i,1);
%     n22_PQ=icon_PQ(i,2);
%     n33_PQ=icon_PQ(i,3);
%     n44_PQ=icon_PQ(i,4);
%     %n55_PQ=icon_PQ(i,5);
%     n11_RS=icon_RS(i,1);
%     n22_RS=icon_RS(i,2);
%     n33_RS=icon_RS(i,3);
%     n44_RS=icon_RS(i,4);
    %n55_RS=icon_RS(i,5);
    for op=1:length(gauss_points_y)
        c_special=0.5*cr*(1-
gauss_points_y(op))+0.5*tip_chord*(1+gauss_points_y(op));
    for j=1:length(gauss_points_x)
P_non_derive_I_J_x(1,1)=1;
P_non_derive_I_J_x(1,2)=((t_1/
(c_special))*gauss_points_x(j)+spar_position_I_J(1));
P_derive_I_J_x(1,1)=0;
P_derive_I_J_x(1,2)=1;
P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=((t_1/
c_special)*gauss_points_x(j)+spar_position_K_L(1));
P_derive_M_N_x(1,1)=0;
```

```
P_derive_M_N_x(1,2)=1;
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_derive_M_N_x(1,1)=0;
P_derive_M_N_x(1,2)=1;
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=((t_1/
c_special)*gauss_points_x(j)+spar_position_P_Q(1));
P_derive_P_Q_x(1,1)=0;
P_derive_P_Q_x(1,2)=1;
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=((t_1/
c_special)*gauss_points_x(j)+spar_position_R_S(1));
P_derive_R_S_x(1,1)=0;
P_derive_R_S_x(1,2)=1;
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=((t_1/
c_special)*gauss_points_y(op)+spar_position_I_J(1));
P_derive_I_J_y(1,1)=0;
P_derive_I_J_y(1,2)=1;
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=((t_1/
c_special)*gauss_points_y(op)+spar_position_K_L(1));
P_derive_M_N_y(1,1)=0;
P_derive_M_N_y(1,2)=1;
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_derive_M_N_y(1,1)=0;
P_derive_M_N_y(1,2)=1;
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=((t_1/
c_special)*gauss_points_y(op)+spar_position_P_Q(1));
P_derive_P_Q_y(1,1)=0;
P_derive_P_Q_y(1,2)=1;
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=((t_1/
c_special)*gauss_points_y(op)+spar_position_R_S(1));
P_derive_R_S_y(1,1)=0;
P_derive_R_S_y(1,2)=1;
        for k=3:number_of_nodes_I_J
           P_derive_I_J_x(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_x(j)+spar_position_I_J(1,k-1)))*P_derive_I_J_x(1,k-1)-
(k/(k+1)*P_derive_I_J_x(1,k-2)));
           P_non_derive_I_J_x(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_x(j)+spar_position_I_J(1,k-1)))*P_non_derive_I_J_x(1,k-1)-
(k/(k+1)*P_non_derive_I_J_x(1,k-2)));
        end
        for k=3:number_of_nodes_K_L
           P_derive_K_L_x(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_x(j)+spar_position_K_L(1,k-1)))*P_derive_K_L_x(1,k-1)-
(k/(k+1)*P_derive_K_L_x(1,k-2)));
           P_non_derive_K_L_x(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_x(j)+spar_position_K_L(1,k-1)))*P_non_derive_K_L_x(1,k-1)-
(k/(k+1)*P_non_derive_K_L_x(1,k-2)));
```

```
        end
%         for k=3:number_of_nodes_M_N
%            P_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_M_N_x(1,k-1)-(k/(k
+1)*P_derive_M_N_x(1,k-2)));
%            P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
%         end
        for k=3:number_of_nodes_P_Q
           P_derive_P_Q_x(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_x(j)+spar_position_P_Q(1,k-1)))*P_derive_P_Q_x(1,k-1)-
(k/(k+1)*P_derive_P_Q_x(1,k-2)));
           P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_x(j)+spar_position_P_Q(1,k-1)))*P_non_derive_P_Q_x(1,k-1)-
(k/(k+1)*P_non_derive_P_Q_x(1,k-2)));
        end
        for k=3:number_of_nodes_R_S
           P_derive_R_S_x(1,k)=((((2*k+1)/(k+1))*(t_1/
c_special)*(gauss_points_x(j)+spar_position_R_S(1,k-1)))*P_derive_R_S_x(1,k-1)-
(k/(k+1)*P_derive_R_S_x(1,k-2)));
           P_non_derive_R_S_x(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_x(j)+spar_position_R_S(1,k-1)))*P_non_derive_R_S_x(1,k-1)-
(k/(k+1)*P_non_derive_R_S_x(1,k-2)));
        end
        for k=3:number_of_nodes_I_J
           P_derive_I_J_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_y(op)+spar_position_I_J(1,k-1)))*P_derive_I_J_y(1,k-1)-
(k/(k+1)*P_derive_I_J_y(1,k-2)));
           P_non_derive_I_J_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_y(op)+spar_position_I_J(1,k-1)))*P_non_derive_I_J_y(1,k-1)
(k/(k+1)*P_non_derive_I_J_y(1,k-2)));
        end
        for k=3:number_of_nodes_K_L
           P_derive_K_L_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_y(op)+spar_position_K_L(1,k-1)))*P_derive_K_L_x(1,k-1)-
(k/(k+1)*P_derive_K_L_x(1,k-2)));
           P_non_derive_K_L_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_y(op)+spar_position_K_L(1,k-1)))*P_non_derive_K_L_y(1,k-1)
(k/(k+1)*P_non_derive_K_L_y(1,k-2)));
        end
%         for k=3:number_of_nodes_M_N
%            P_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(j))*P_derive_M_N_y(1,k-1)-(k/(k
+1)*P_derive_M_N_y(1,k-2)));
%            P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
%         end
        for k=3:number_of_nodes_P_Q
           P_derive_P_Q_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_y(op)+spar_position_P_Q(1,k-1)))*P_derive_P_Q_y(1,k-1)-
(k/(k+1)*P_derive_P_Q_y(1,k-2)));
```

```
            P_non_derive_P_Q_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_y(op)+spar_position_P_Q(1,k-1)))*P_non_derive_P_Q_y(1,k-1)
(k/(k+1)*P_non_derive_P_Q_y(1,k-2)));
         end
         for k=3:number_of_nodes_R_S
            P_derive_R_S_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_y(op)+spar_position_R_S(1,k-1)))*P_derive_R_S_y(1,k-1)-
(k/(k+1)*P_derive_R_S_y(1,k-2)));
            P_non_derive_R_S_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_y(op)+spar_position_R_S(1,k-1)))*P_non_derive_R_S_y(1,k-1)
(k/(k+1)*P_non_derive_R_S_y(1,k-2)));
         end

 [detJacobian,inverse]=coordinate_transformation_Jacobian(sweep_angle,cr,b,gauss_p
         inverse_trans=inverse';
         final_IJ_x=P_derive_I_J_x.*P_non_derive_I_J_x;
         final_KL_x=P_derive_K_L_x.*P_non_derive_K_L_x;
        % final_MN_x=P_derive_M_N_x.*P_non_derive_M_N_x;
         final_PQ_x=P_derive_P_Q_x.*P_non_derive_P_Q_x;
         final_RS_x=P_derive_R_S_x.*P_non_derive_R_S_x;
         final_IJ_y=P_derive_I_J_y.*P_non_derive_I_J_y;
         final_KL_y=P_derive_K_L_y.*P_non_derive_K_L_y;
         %final_MN_y=P_derive_M_N_y.*P_non_derive_M_N_y;
         final_PQ_y=P_derive_P_Q_y.*P_non_derive_P_Q_y;
         final_RS_y=P_derive_R_S_y.*P_non_derive_R_S_y;
         boom=5;
         idk=1;
%          final_IJ=[final_IJ_x;final_IJ_y];
%          final_KL=[final_KL_x;final_KL_y];
%          final_MN=[final_MN_x;final_MN_y];
%          final_PQ=[final_PQ_x;final_PQ_y];
%          final_RS=[final_RS_x;final_RS_y];
         for pi=1:length(final_IJ_x)
             B(1:12,idk:boom)=B(1:12,idk:boom)+[final_IJ_x(1,pi)
 0 0 0 0;final_IJ_y(1,pi) 0 0 0 0;0 final_KL_x(1,pi) 0 0 0;0
 final_KL_y(1,pi) 0 0 0;0 0 0 0 0;0 0 0 0 0;0 0 0 final_PQ_x(1,pi)
 0;0 0 0 final_PQ_y(1,pi) 0;0 0 0 0 final_RS_x(1,pi);0 0 0 0
 final_RS_y(1,pi);0 0 0 0 0;0 0 0 0 0];
             idk=idk+5;
             boom=boom+5;
         end

 D=D_matrix_general_in_plane(sweep_angle,cr,b,theta,thickness_of_each_ply,gauss_po
+D_matrix_upper_airfoil_in_plane(sweep_angle,cr,b,theta,thickness_of_each_ply,gaus
         hope=hope+(t_1/
c_special)*B'*D*B*weights_x(j)*weights_y(op)*detJacobian;
    end
    end
%     iv_0=[5*n11_IJ-4 5*n11_IJ-3 5*n11_IJ-2 5*n11_IJ-1 5*n11_IJ
 5*n22_IJ-4 5*n22_IJ-3 5*n22_IJ-2 5*n22_IJ-1 5*n22_IJ 5*n33_IJ-4
 5*n33_IJ-3 5*n33_IJ-2 5*n33_IJ-1 5*n33_IJ 5*n44_IJ-4 5*n44_IJ-3
 5*n44_IJ-2 5*n44_IJ-1 5*n44_IJ];
%     iv_1=[5*n11_KL-4 5*n11_KL-3 5*n11_KL-2 5*n11_KL-1 5*n11_KL
 5*n22_KL-4 5*n22_KL-3 5*n22_KL-2 5*n22_KL-1 5*n22_KL 5*n33_KL-4
```

```
 5*n33_KL-3 5*n33_KL-2 5*n33_KL-1 5*n33_KL 5*n44_KL-4 5*n44_KL-3
 5*n44_KL-2 5*n44_KL-1 5*n44_KL];
%     iv_2=[5*n11_PQ-4 5*n11_PQ-3 5*n11_PQ-2 5*n11_PQ-1 5*n11_PQ
 5*n22_PQ-4 5*n22_PQ-3 5*n22_PQ-2 5*n22_PQ-1 5*n22_PQ 5*n33_PQ-4
 5*n33_PQ-3 5*n33_PQ-2 5*n33_PQ-1 5*n33_PQ 5*n44_PQ-4 5*n44_PQ-3
 5*n44_PQ-2 5*n44_PQ-1 5*n44_PQ];
%     iv_3=[5*n11_RS-4 5*n11_RS-3 5*n11_RS-2 5*n11_RS-1 5*n11_RS
 5*n22_RS-4 5*n22_RS-3 5*n22_RS-2 5*n22_RS-1 5*n22_RS 5*n33_RS-4
 5*n33_RS-3 5*n33_RS-2 5*n33_RS-1 5*n33_RS 5*n44_RS-4 5*n44_RS-3
 5*n44_RS-2 5*n44_RS-1 5*n44_RS];
%     iv=[iv_0 iv_1 iv_2 iv_3];
    %kg(iv,iv)=kg(iv,iv)+hope;
end
kg_in_plane=hope;
end
```

```
Not enough input arguments.

Error in B_matrix_in_plane_spar_web (line 14)
sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 kg_in_plane=B_matrix_in_plane(gauss_points_x,gauss_points_y,sweep_angle,b,cr,I,J,
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
% weights_x=[1 1];
% weights_y=[1 1];
% sweep_angle=30;%degree
% b=10;
% l_1=1;
% c=1;
% h=1;
% h_1=1;
 pi=3.1416;
% t_1=1;
sweep_angle_in_radians=sweep_angle*pi/180;
% cr=5;
 tip_chord=cr-((b/2)*tan(sweep_angle_in_radians));
%Linear nodes per element
% I=3;
% J=3;
% K=3;
% L=3;
% M=3;
% N=3;
% P=3;
% Q=3;
% R=3;
% S=3;
% dof=(5);
% icon_IJ=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_KL=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_PQ=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_RS=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
coeff_of_z_lower=Lower_fitting_a_curve();
coeff_of_z_upper=Upper_fitting_a_curve();
number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
%number_of_elements=number_of_nodes-1;
P_non_derive_I_J_x=zeros(1,I*J);
P_derive_I_J_x=zeros(1,I*J);
P_derive_K_L_x=zeros(1,K*L);
P_non_derive_K_L_x=zeros(1,K*L);
P_derive_M_N_x=zeros(1,M*N);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
```

```
P_derive_I_J_y=zeros(1,I*J);
P_derive_K_L_y=zeros(1,K*L);
P_non_derive_K_L_y=zeros(1,K*L);
P_derive_M_N_y=zeros(1,M*N);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_derive_R_S_y=zeros(1,R*S);
B=zeros(12,(I*J+P*Q+M*N+K*L+R*S));
%  theta=[0 90 90 0];
%  thickness_of_each_ply=0.01*ones(1,length(theta));
hope=zeros((I*J+P*Q+M*N+K*L+R*S),(I*J+P*Q+M*N+K*L+R*S));
spar_position_I_J=linspace(-1,1,I*J);
spar_position_K_L=linspace(-1,1,K*L);
spar_position_P_Q=linspace(-1,1,P*Q);
spar_position_R_S=linspace(-1,1,R*S);
%kg=zeros(((I*J)+(K*L)+(P*Q)+(R*S))*dof,
((I*J)+(K*L)+(P*Q)+(R*S))*dof);
for
 i=1:I*J%*length(icon_KL(:,1))*length(icon_KL(1,:))*length(icon_PQ(:,1))*length(ic
%     n11_IJ=icon_IJ(i,1);
%     n22_IJ=icon_IJ(i,2);
%     n33_IJ=icon_IJ(i,3);
%     n44_IJ=icon_IJ(i,4);
%     %n55_IJ=icon_IJ(i,5);
%     n11_KL=icon_KL(i,1);
%     n22_KL=icon_KL(i,2);
%     n33_KL=icon_KL(i,3);
%     n44_KL=icon_KL(i,4);
%     %n55_KL=icon_KL(i,5);
%     n11_PQ=icon_PQ(i,1);
%     n22_PQ=icon_PQ(i,2);
%     n33_PQ=icon_PQ(i,3);
%     n44_PQ=icon_PQ(i,4);
%     %n55_PQ=icon_PQ(i,5);
%     n11_RS=icon_RS(i,1);
%     n22_RS=icon_RS(i,2);
%     n33_RS=icon_RS(i,3);
%     n44_RS=icon_RS(i,4);
    %n55_RS=icon_RS(i,5);
    for op=1:length(gauss_points_y)
        c_special=0.5*cr*(1-
gauss_points_y(op))+0.5*tip_chord*(1+gauss_points_y(op));
    for j=1:length(gauss_points_x)
P_non_derive_I_J_x(1,1)=1;
P_non_derive_I_J_x(1,2)=((t_1/
(c_special))*gauss_points_x(j)+spar_position_I_J(1));
P_derive_I_J_x(1,1)=0;
P_derive_I_J_x(1,2)=1;
P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=((t_1/
c_special)*gauss_points_x(j)+spar_position_K_L(1));
P_derive_M_N_x(1,1)=0;
```

```
P_derive_M_N_x(1,2)=1;
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_derive_M_N_x(1,1)=0;
P_derive_M_N_x(1,2)=1;
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=((t_1/
c_special)*gauss_points_x(j)+spar_position_P_Q(1));
P_derive_P_Q_x(1,1)=0;
P_derive_P_Q_x(1,2)=1;
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=((t_1/
c_special)*gauss_points_x(j)+spar_position_R_S(1));
P_derive_R_S_x(1,1)=0;
P_derive_R_S_x(1,2)=1;
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=((t_1/
c_special)*gauss_points_y(op)+spar_position_I_J(1));
P_derive_I_J_y(1,1)=0;
P_derive_I_J_y(1,2)=1;
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=((t_1/
c_special)*gauss_points_y(op)+spar_position_K_L(1));
P_derive_M_N_y(1,1)=0;
P_derive_M_N_y(1,2)=1;
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_derive_M_N_y(1,1)=0;
P_derive_M_N_y(1,2)=1;
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=((t_1/
c_special)*gauss_points_y(op)+spar_position_P_Q(1));
P_derive_P_Q_y(1,1)=0;
P_derive_P_Q_y(1,2)=1;
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=((t_1/
c_special)*gauss_points_y(op)+spar_position_R_S(1));
P_derive_R_S_y(1,1)=0;
P_derive_R_S_y(1,2)=1;
        for k=3:number_of_nodes_I_J
           P_derive_I_J_x(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_x(j)+spar_position_I_J(1,k-1)))*P_derive_I_J_x(1,k-1)-
(k/(k+1)*P_derive_I_J_x(1,k-2)));
           P_non_derive_I_J_x(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_x(j)+spar_position_I_J(1,k-1)))*P_non_derive_I_J_x(1,k-1)-
(k/(k+1)*P_non_derive_I_J_x(1,k-2)));
        end
        for k=3:number_of_nodes_K_L
           P_derive_K_L_x(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_x(j)+spar_position_K_L(1,k-1)))*P_derive_K_L_x(1,k-1)-
(k/(k+1)*P_derive_K_L_x(1,k-2)));
           P_non_derive_K_L_x(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_x(j)+spar_position_K_L(1,k-1)))*P_non_derive_K_L_x(1,k-1)-
(k/(k+1)*P_non_derive_K_L_x(1,k-2)));
```

```
        end
%         for k=3:number_of_nodes_M_N
%            P_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_M_N_x(1,k-1)-(k/(k
+1)*P_derive_M_N_x(1,k-2)));
%            P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
%         end
        for k=3:number_of_nodes_P_Q
           P_derive_P_Q_x(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_x(j)+spar_position_P_Q(1,k-1)))*P_derive_P_Q_x(1,k-1)-
(k/(k+1)*P_derive_P_Q_x(1,k-2)));
           P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_x(j)+spar_position_P_Q(1,k-1)))*P_non_derive_P_Q_x(1,k-1)-
(k/(k+1)*P_non_derive_P_Q_x(1,k-2)));
        end
        for k=3:number_of_nodes_R_S
           P_derive_R_S_x(1,k)=((((2*k+1)/(k+1))*(t_1/
c_special)*(gauss_points_x(j)+spar_position_R_S(1,k-1)))*P_derive_R_S_x(1,k-1)-
(k/(k+1)*P_derive_R_S_x(1,k-2)));
           P_non_derive_R_S_x(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_x(j)+spar_position_R_S(1,k-1)))*P_non_derive_R_S_x(1,k-1)-
(k/(k+1)*P_non_derive_R_S_x(1,k-2)));
        end
        for k=3:number_of_nodes_I_J
           P_derive_I_J_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_y(op)+spar_position_I_J(1,k-1)))*P_derive_I_J_y(1,k-1)-
(k/(k+1)*P_derive_I_J_y(1,k-2)));
           P_non_derive_I_J_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_y(op)+spar_position_I_J(1,k-1)))*P_non_derive_I_J_y(1,k-1)
(k/(k+1)*P_non_derive_I_J_y(1,k-2)));
        end
        for k=3:number_of_nodes_K_L
           P_derive_K_L_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_y(op)+spar_position_K_L(1,k-1)))*P_derive_K_L_x(1,k-1)-
(k/(k+1)*P_derive_K_L_x(1,k-2)));
           P_non_derive_K_L_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_y(op)+spar_position_K_L(1,k-1)))*P_non_derive_K_L_y(1,k-1)
(k/(k+1)*P_non_derive_K_L_y(1,k-2)));
        end
%         for k=3:number_of_nodes_M_N
%            P_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(j))*P_derive_M_N_y(1,k-1)-(k/(k
+1)*P_derive_M_N_y(1,k-2)));
%            P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
%         end
        for k=3:number_of_nodes_P_Q
           P_derive_P_Q_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_y(op)+spar_position_P_Q(1,k-1)))*P_derive_P_Q_y(1,k-1)-
(k/(k+1)*P_derive_P_Q_y(1,k-2)));
```

```
            P_non_derive_P_Q_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_y(op)+spar_position_P_Q(1,k-1)))*P_non_derive_P_Q_y(1,k-1)
(k/(k+1)*P_non_derive_P_Q_y(1,k-2)));
         end
         for k=3:number_of_nodes_R_S
            P_derive_R_S_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_y(op)+spar_position_R_S(1,k-1)))*P_derive_R_S_y(1,k-1)-
(k/(k+1)*P_derive_R_S_y(1,k-2)));
            P_non_derive_R_S_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c_special)*gauss_points_y(op)+spar_position_R_S(1,k-1)))*P_non_derive_R_S_y(1,k-1)
(k/(k+1)*P_non_derive_R_S_y(1,k-2)));
         end

 [detJacobian,inverse]=coordinate_transformation_Jacobian(sweep_angle,cr,b,gauss_p
         inverse_trans=inverse';
         final_IJ_x=P_derive_I_J_x.*P_non_derive_I_J_x;
         final_KL_x=P_derive_K_L_x.*P_non_derive_K_L_x;
        % final_MN_x=P_derive_M_N_x.*P_non_derive_M_N_x;
         final_PQ_x=P_derive_P_Q_x.*P_non_derive_P_Q_x;
         final_RS_x=P_derive_R_S_x.*P_non_derive_R_S_x;
         final_IJ_y=P_derive_I_J_y.*P_non_derive_I_J_y;
         final_KL_y=P_derive_K_L_y.*P_non_derive_K_L_y;
         %final_MN_y=P_derive_M_N_y.*P_non_derive_M_N_y;
         final_PQ_y=P_derive_P_Q_y.*P_non_derive_P_Q_y;
         final_RS_y=P_derive_R_S_y.*P_non_derive_R_S_y;
         boom=5;
         idk=1;
%          final_IJ=[final_IJ_x;final_IJ_y];
%          final_KL=[final_KL_x;final_KL_y];
%          final_MN=[final_MN_x;final_MN_y];
%          final_PQ=[final_PQ_x;final_PQ_y];
%          final_RS=[final_RS_x;final_RS_y];
         for pi=1:length(final_IJ_x)
             B(1:12,idk:boom)=B(1:12,idk:boom)+[final_IJ_x(1,pi)
 0 0 0 0;final_IJ_y(1,pi) 0 0 0 0;0 final_KL_x(1,pi) 0 0 0;0
 final_KL_y(1,pi) 0 0 0;0 0 0 0 0;0 0 0 0 0;0 0 0 final_PQ_x(1,pi)
 0;0 0 0 final_PQ_y(1,pi) 0;0 0 0 0 final_RS_x(1,pi);0 0 0 0
 final_RS_y(1,pi);0 0 0 0 0;0 0 0 0 0];
             idk=idk+5;
             boom=boom+5;
         end

 D=D_matrix_general_in_plane(sweep_angle,cr,b,theta,thickness_of_each_ply,gauss_po
+D_matrix_upper_airfoil_in_plane(sweep_angle,cr,b,theta,thickness_of_each_ply,gaus
         hope=hope+(t_1/
c_special)*B'*D*B*weights_x(j)*weights_y(op)*detJacobian;
    end
    end
%     iv_0=[5*n11_IJ-4 5*n11_IJ-3 5*n11_IJ-2 5*n11_IJ-1 5*n11_IJ
 5*n22_IJ-4 5*n22_IJ-3 5*n22_IJ-2 5*n22_IJ-1 5*n22_IJ 5*n33_IJ-4
 5*n33_IJ-3 5*n33_IJ-2 5*n33_IJ-1 5*n33_IJ 5*n44_IJ-4 5*n44_IJ-3
 5*n44_IJ-2 5*n44_IJ-1 5*n44_IJ];
%     iv_1=[5*n11_KL-4 5*n11_KL-3 5*n11_KL-2 5*n11_KL-1 5*n11_KL
 5*n22_KL-4 5*n22_KL-3 5*n22_KL-2 5*n22_KL-1 5*n22_KL 5*n33_KL-4
```

```
 5*n33_KL-3 5*n33_KL-2 5*n33_KL-1 5*n33_KL 5*n44_KL-4 5*n44_KL-3
 5*n44_KL-2 5*n44_KL-1 5*n44_KL];
%     iv_2=[5*n11_PQ-4 5*n11_PQ-3 5*n11_PQ-2 5*n11_PQ-1 5*n11_PQ
 5*n22_PQ-4 5*n22_PQ-3 5*n22_PQ-2 5*n22_PQ-1 5*n22_PQ 5*n33_PQ-4
 5*n33_PQ-3 5*n33_PQ-2 5*n33_PQ-1 5*n33_PQ 5*n44_PQ-4 5*n44_PQ-3
 5*n44_PQ-2 5*n44_PQ-1 5*n44_PQ];
%     iv_3=[5*n11_RS-4 5*n11_RS-3 5*n11_RS-2 5*n11_RS-1 5*n11_RS
 5*n22_RS-4 5*n22_RS-3 5*n22_RS-2 5*n22_RS-1 5*n22_RS 5*n33_RS-4
 5*n33_RS-3 5*n33_RS-2 5*n33_RS-1 5*n33_RS 5*n44_RS-4 5*n44_RS-3
 5*n44_RS-2 5*n44_RS-1 5*n44_RS];
%     iv=[iv_0 iv_1 iv_2 iv_3];
    %kg(iv,iv)=kg(iv,iv)+hope;
end
kg_in_plane=hope;
end
```

```
Not enough input arguments.

Error in B_matrix_in_plane_spar_web (line 14)
sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 kg_trans_plane=B_matrix_trans_plane(gauss_points_x,gauss_points_y,sweep_angle,b,c
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
% weights_x=[1 1];
% weights_y=[1 1];
% sweep_angle=30;%degree
% b=10;
pi=3.1416;
sweep_angle_in_radians=sweep_angle*pi/180;
% cr=5;
% l_1=1;
% c=1;
% h=1;
% h_1=1;
tip_chord=cr-((b/2)*tan(sweep_angle_in_radians));
%Linear nodes per element
% I=2;
% J=2;
% K=2;
% L=2;
% M=2;
% N=2;
% P=2;
% Q=2;
% R=2;
% S=2;
% dof=(5);
% icon_IJ=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_KL=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_PQ=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_RS=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
coeff_of_z_lower=Lower_fitting_a_curve();
coeff_of_z_upper=Upper_fitting_a_curve();
number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
%number_of_elements=number_of_nodes-1;
P_non_derive_I_J_x=zeros(1,I*J);
P_derive_I_J_x=zeros(1,I*J);
P_derive_K_L_x=zeros(1,K*L);
P_non_derive_K_L_x=zeros(1,K*L);
P_derive_M_N_x=zeros(1,M*N);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_derive_I_J_y=zeros(1,I*J);
```

```
P_derive_K_L_y=zeros(1,K*L);
P_non_derive_K_L_y=zeros(1,K*L);
P_derive_M_N_y=zeros(1,M*N);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_derive_R_S_y=zeros(1,R*S);
B=zeros(12,(I*J+P*Q+M*N+K*L+R*S));
% theta=[0 90 90 0];
% thickness_of_each_ply=0.01*ones(1,length(theta));
hope=zeros((I*J+P*Q+M*N+K*L+R*S),(I*J+P*Q+M*N+K*L+R*S));
spar_position_I_J=linspace(-1,1,I*J);
spar_position_K_L=linspace(-1,1,K*L);
spar_position_M_N=linspace(-1,1,M*N);
spar_position_P_Q=linspace(-1,1,P*Q);
spar_position_R_S=linspace(-1,1,R*S);
%kg=zeros(((I*J)+(K*L)+(P*Q)+(R*S))*dof,
((I*J)+(K*L)+(P*Q)+(R*S))*dof);
for
 i=1:I*J%*length(icon_KL(:,1))*length(icon_KL(1,:))*length(icon_PQ(:,1))*length(ic
%     n11_IJ=icon_IJ(i,1);
%     n22_IJ=icon_IJ(i,2);
%     n33_IJ=icon_IJ(i,3);
%     n44_IJ=icon_IJ(i,4);
%     %n55_IJ=icon_IJ(i,5);
%     n11_KL=icon_KL(i,1);
%     n22_KL=icon_KL(i,2);
%     n33_KL=icon_KL(i,3);
%     n44_KL=icon_KL(i,4);
%     %n55_KL=icon_KL(i,5);
%     n11_PQ=icon_PQ(i,1);
%     n22_PQ=icon_PQ(i,2);
%     n33_PQ=icon_PQ(i,3);
%     n44_PQ=icon_PQ(i,4);
%     %n55_PQ=icon_PQ(i,5);
%     n11_RS=icon_RS(i,1);
%     n22_RS=icon_RS(i,2);
%     n33_RS=icon_RS(i,3);
%     n44_RS=icon_RS(i,4);
    %n55_RS=icon_RS(i,5);
    for op=1:length(gauss_points_y)
        c=0.5*cr*(1-
gauss_points_y(op))+0.5*tip_chord*(1+gauss_points_y(op));
    for j=1:length(gauss_points_x)
P_non_derive_I_J_x(1,1)=1;
P_non_derive_I_J_x(1,2)=gauss_points_x(j);
P_derive_I_J_x(1,1)=0;
P_derive_I_J_x(1,2)=1;
P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=gauss_points_x(j);
P_derive_M_N_x(1,1)=0;
P_derive_M_N_x(1,2)=1;
P_non_derive_M_N_x(1,1)=1;
```

```
P_non_derive_M_N_x(1,2)=(t_1/
c)*gauss_points_x(j)+spar_position_M_N(1);
P_derive_M_N_x(1,1)=0;
P_derive_M_N_x(1,2)=(t_1/c);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=gauss_points_x(j);
P_derive_P_Q_x(1,1)=0;
P_derive_P_Q_x(1,2)=1;
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=gauss_points_x(j);
P_derive_R_S_x(1,1)=0;
P_derive_R_S_x(1,2)=1;
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=gauss_points_y(op);
P_derive_I_J_y(1,1)=0;
P_derive_I_J_y(1,2)=1;
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=gauss_points_y(op);
P_derive_M_N_y(1,1)=0;
P_derive_M_N_y(1,2)=1;
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=(t_1/
c)*gauss_points_y(op)+spar_position_M_N(1);
P_derive_M_N_y(1,1)=0;
P_derive_M_N_y(1,2)=(t_1/c);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_derive_P_Q_y(1,1)=0;
P_derive_P_Q_y(1,2)=1;
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=gauss_points_y(op);
P_derive_R_S_y(1,1)=0;
P_derive_R_S_y(1,2)=1;
        for k=3:number_of_nodes_I_J
           P_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_I_J_x(1,k-1)-(k/(k
+1)*P_derive_I_J_x(1,k-2)));
           P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2)));
        end
        for k=3:number_of_nodes_K_L
           P_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_K_L_x(1,k-1)-(k/(k
+1)*P_derive_K_L_x(1,k-2)));
           P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
        end
        for k=3:number_of_nodes_M_N
           P_derive_M_N_x(1,k)=((((2*k+1)/(k+1))*(t_1/
c)*gauss_points_x(j)+spar_position_M_N(1,k-1))*P_derive_M_N_x(1,k-1)-
(k/(k+1)*P_derive_M_N_x(1,k-2)));
```

```
            P_non_derive_M_N_x(1,k)=((((2*k+1)/(k+1))*(t_1/
c)*gauss_points_x(j)+spar_position_M_N(1,k-1))*P_non_derive_M_N_x(1,k-1)-
(k/(k+1)*P_non_derive_M_N_x(1,k-2)));
        end
        for k=3:number_of_nodes_P_Q
            P_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_derive_P_Q_x(1,k-2)));
            P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
        end
        for k=3:number_of_nodes_R_S
            P_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_R_S_x(1,k-1)-(k/(k
+1)*P_derive_R_S_x(1,k-2)));
            P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
        end
        for k=3:number_of_nodes_I_J
            P_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(j))*P_derive_I_J_y(1,k-1)-(k/(k
+1)*P_derive_I_J_y(1,k-2)));
            P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
        end
        for k=3:number_of_nodes_K_L
            P_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(j))*P_derive_K_L_x(1,k-1)-(k/(k
+1)*P_derive_K_L_x(1,k-2)));
            P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
        end
        for k=3:number_of_nodes_M_N
            P_derive_M_N_y(1,k)=((((2*k+1)/(k+1))*(t_1/
c)*gauss_points_y(op)+spar_position_M_N(1,k-1))*P_derive_M_N_y(1,k-1)-
(k/(k+1)*P_derive_M_N_y(1,k-2)));
            P_non_derive_M_N_y(1,k)=((((2*k+1)/(k+1))*(t_1/
c)*gauss_points_y(op)+spar_position_M_N(1,k-1))*P_non_derive_M_N_y(1,k-1)-
(k/(k+1)*P_non_derive_M_N_y(1,k-2)));
        end
        for k=3:number_of_nodes_P_Q
            P_derive_P_Q_y(1,k)=((((2*k+1)/(k+1))*(t_1/
c)*gauss_points_y(op)+spar_position_P_Q(1,k-1))*P_derive_P_Q_y(1,k-1)-
(k/(k+1)*P_derive_P_Q_y(1,k-2)));
            P_non_derive_P_Q_y(1,k)=((((2*k+1)/(k+1))*(t_1/
c)*gauss_points_y(op)+spar_position_P_Q(1,k-1))*P_non_derive_P_Q_y(1,k-1)-
(k/(k+1)*P_non_derive_P_Q_y(1,k-2)));
        end
        for k=3:number_of_nodes_R_S
```

```
            P_derive_R_S_y(1,k)=((((2*k+1)/(k+1))*(t_1/
c)*gauss_points_y(op)+spar_position_R_S(1,k-1))*P_derive_R_S_y(1,k-1)-
(k/(k+1)*P_derive_R_S_y(1,k-2)));
            P_non_derive_R_S_y(1,k)=((((2*k+1)/(k+1))*(t_1/
c)*gauss_points_y(op)+spar_position_R_S(1,k-1))*P_non_derive_R_S_y(1,k-1)-
(k/(k+1)*P_non_derive_R_S_y(1,k-2)));
        end

 [detJacobian,inverse]=coordinate_transformation_Jacobian(sweep_angle,cr,b,gauss_p
        inverse_trans=inverse';
%         final_IJ_x=P_derive_I_J_x.*P_non_derive_I_J_x;
%         final_KL_x=P_derive_K_L_x.*P_non_derive_K_L_x;
        final_MN_x=P_derive_M_N_x.*P_non_derive_M_N_x;
%         final_PQ_x=P_derive_P_Q_x.*P_non_derive_P_Q_x;
%         final_RS_x=P_derive_R_S_x.*P_non_derive_R_S_x;
%         final_IJ_y=P_derive_I_J_y.*P_non_derive_I_J_y;
%         final_KL_y=P_derive_K_L_y.*P_non_derive_K_L_y;
        final_MN_y=P_derive_M_N_y.*P_non_derive_M_N_y;
%         final_PQ_y=P_derive_P_Q_y.*P_non_derive_P_Q_y;
%         final_RS_y=P_derive_R_S_y.*P_non_derive_R_S_y;
final_PQ=P_non_derive_P_Q_x.*P_non_derive_P_Q_y;
final_RS=P_non_derive_R_S_x.*P_non_derive_R_S_y;
        boom=5;
        idk=1;
%         final_IJ=[final_IJ_x;final_IJ_y];
%         final_KL=[final_KL_x;final_KL_y];
%         final_MN=[final_MN_x;final_MN_y];
%         final_PQ=[final_PQ_x;final_PQ_y];
%         final_RS=[final_RS_x;final_RS_y];
        for pi=1:length(final_MN_x)
            B(1:12,idk:boom)=B(1:12,idk:boom)+[0 0 0 0 0;0 0 0 0 0;0 0
 0 0 0;0 0 0 0 0;0 0 final_MN_x(1,pi) 0 0;0 0 final_MN_y(1,pi) 0 0;0
 0 0 0 0;0 0 0 0 0;0 0 0 0 0;0 0 0 0 0;0 0 0 final_PQ(1,pi) 0;0 0 0 0
 final_RS(1,pi)];
            idk=idk+5;
            boom=boom+5;
        end

 D=D_matrix_general_trans_plane(sweep_angle,cr,b,theta,thickness_of_each_ply,gauss
+D_matrix_upper_airfoil_trans_plane(sweep_angle,cr,b,theta,thickness_of_each_ply,g
        hope=hope+(t_1/
c)*B'*D*B*weights_x(j)*weights_y(op)*detJacobian;
    end
    end
%     iv_0=[5*n11_IJ-4 5*n11_IJ-3 5*n11_IJ-2 5*n11_IJ-1 5*n11_IJ
 5*n22_IJ-4 5*n22_IJ-3 5*n22_IJ-2 5*n22_IJ-1 5*n22_IJ 5*n33_IJ-4
 5*n33_IJ-3 5*n33_IJ-2 5*n33_IJ-1 5*n33_IJ 5*n44_IJ-4 5*n44_IJ-3
 5*n44_IJ-2 5*n44_IJ-1 5*n44_IJ];
%     iv_1=[5*n11_KL-4 5*n11_KL-3 5*n11_KL-2 5*n11_KL-1 5*n11_KL
 5*n22_KL-4 5*n22_KL-3 5*n22_KL-2 5*n22_KL-1 5*n22_KL 5*n33_KL-4
 5*n33_KL-3 5*n33_KL-2 5*n33_KL-1 5*n33_KL 5*n44_KL-4 5*n44_KL-3
 5*n44_KL-2 5*n44_KL-1 5*n44_KL];
%     iv_2=[5*n11_PQ-4 5*n11_PQ-3 5*n11_PQ-2 5*n11_PQ-1 5*n11_PQ
 5*n22_PQ-4 5*n22_PQ-3 5*n22_PQ-2 5*n22_PQ-1 5*n22_PQ 5*n33_PQ-4
```

```
 5*n33_PQ-3 5*n33_PQ-2 5*n33_PQ-1 5*n33_PQ 5*n44_PQ-4 5*n44_PQ-3
 5*n44_PQ-2 5*n44_PQ-1 5*n44_PQ];
%     iv_3=[5*n11_RS-4 5*n11_RS-3 5*n11_RS-2 5*n11_RS-1 5*n11_RS
 5*n22_RS-4 5*n22_RS-3 5*n22_RS-2 5*n22_RS-1 5*n22_RS 5*n33_RS-4
 5*n33_RS-3 5*n33_RS-2 5*n33_RS-1 5*n33_RS 5*n44_RS-4 5*n44_RS-3
 5*n44_RS-2 5*n44_RS-1 5*n44_RS];
%     iv=[iv_0 iv_1 iv_2 iv_3];
    %kg(iv,iv)=kg(iv,iv)+hope;
end
kg_trans_plane=hope;
end
```

```
Not enough input arguments.

Error in B_matrix_trans_plane_spar_web (line 9)
sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 D_matrix_1=D_matrix_general_in_plane(sweep,cr,b,theta,thickness_of_each_ply,gauss
% coeff_of_z_lower=Lower_fitting_a_curve();
% coeff_of_z_upper=Upper_fitting_a_curve();
% sweep=30;
% cr=10*10^-3;
% b=20*10^-3;
thickness_lower=0.01;
Z=zeros(5,12);
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
z_2=0;
z_derive_1_lower=0;
z_derive_2_lower=0;
z_derive_3_lower=0;
z_derive_1_upper=0;
z_derive_2_upper=0;
z_derive_3_upper=0;
n=10;
z_lower=[0.000000  0.000000;
  0.002000 -0.009300;
  0.005000 -0.016000;
  0.010000 -0.022100;
  0.020000 -0.029500;
  0.030000 -0.034400;
  0.040000 -0.038100;
  0.050000 -0.041200;
  0.070000 -0.046200;
  0.100000 -0.051700;
  0.120000 -0.054700;
  0.150000 -0.058500;
  0.170000 -0.060600;
  0.200000 -0.063300;
  0.220000 -0.064700;
  0.250000 -0.066600;
  0.280000 -0.068000;
  0.300000 -0.068700;
  0.320000 -0.069200;
  0.350000 -0.069600;
  0.370000 -0.069600;
  0.400000 -0.069200;
  0.420000 -0.068800;
  0.450000 -0.067600;
  0.480000 -0.065700;
  0.500000 -0.064400;
  0.530000 -0.061400;
  0.550000 -0.058800;
  0.580000 -0.054300;
  0.600000 -0.050900;
  0.630000 -0.045100;
  0.650000 -0.041000;
```

```
  0.680000 -0.034600;
  0.700000 -0.030200;
  0.730000 -0.023500;
  0.750000 -0.019200;
  0.770000 -0.015000;
  0.800000 -0.009300;
  0.830000 -0.004800;
  0.850000 -0.002400;
  0.870000 -0.001300;
  0.890000 -0.000800;
  0.920000 -0.001600;
  0.940000 -0.003500;
  0.950000 -0.004900;
  0.960000 -0.006600;
  0.970000 -0.008500;
  0.980000 -0.010900;
  0.990000 -0.013700;
  1.000000 -0.016300];
z_upper_curve=[0.000000  0.000000;
  0.002000  0.009500;
  0.005000  0.015800;
  0.010000  0.021900;
  0.020000  0.029300;
  0.030000  0.034300;
  0.040000  0.038100;
  0.050000  0.041100;
  0.070000  0.046200;
  0.100000  0.051800;
  0.120000  0.054800;
  0.150000  0.058500;
  0.170000  0.060600;
  0.200000  0.063200;
  0.220000  0.064600;
  0.250000  0.066400;
  0.270000  0.067300;
  0.300000  0.068500;
  0.330000  0.069200;
  0.350000  0.069600;
  0.380000  0.069800;
  0.400000  0.069700;
  0.430000  0.069500;
  0.450000  0.069200;
  0.480000  0.068400;
  0.500000  0.067800;
  0.530000  0.066600;
  0.550000  0.065600;
  0.570000  0.064500;
  0.600000  0.062500;
  0.620000  0.061000;
  0.650000  0.058500;
  0.680000  0.055500;
  0.700000  0.053300;
  0.720000  0.050900;
  0.750000  0.046900;
```

```
  0.770000  0.043900;
  0.800000  0.038900;
  0.820000  0.035300;
  0.850000  0.029400;
  0.870000  0.025100;
  0.900000  0.018100;
  0.920000  0.013100;
  0.950000  0.004900;
  0.970000 -0.000900;
  0.980000 -0.003900;
  0.990000 -0.007100;
  1.000000 -0.010400;
  ];
x=z_lower(1:end,1);
y=z_upper_curve(1:end,1);
y=y';
z_lower_1=z_lower(1:end,1);
for po=1:length(coeff_of_z_lower)
    z_derive_1_lower=z_derive_1_lower
+(coeff_of_z_lower(po)*(((factorial(length(coeff_of_z_lower))/
(factorial(length(coeff_of_z_lower)-
po)*factorial(po))))*po*(length(coeff_of_z_lower)-
po)*((z_lower_1(po))^(po-1)))*((z_lower_1(po))^0.75)*((1-
z_lower(po))^0.75));
    z_derive_2_lower=z_derive_2_lower
+(coeff_of_z_lower(po))*(factorial(length(coeff_of_z_lower))/
((factorial(length(coeff_of_z_lower)-
po)*factorial(po))))*(0.75)*(z_lower_1(po))^(0.75-1)*(1-
z_lower_1(po))^0.75;
     z_derive_3_lower=z_derive_3_lower
+(coeff_of_z_lower(po)*(((factorial(length(coeff_of_z_lower))/
(factorial(length(coeff_of_z_lower))*(0.75*(1-
z_lower_1(po))^(0.75-1)*(z_lower_1(po))^(0.75)))))));
end
for po=1:length(coeff_of_z_upper)
    z_derive_1_upper=z_derive_1_upper
+(coeff_of_z_upper(po)*(((factorial(length(coeff_of_z_upper))/
(factorial(length(coeff_of_z_upper)-
po)*factorial(po))))*po*(length(coeff_of_z_upper)-
po)*((z_upper_curve(po))^(po-1)))*((z_upper_curve(po))^0.75)*((1-
z_upper_curve(po))^0.75));
    z_derive_2_upper=z_derive_2_upper
+(coeff_of_z_upper(po))*(factorial(length(coeff_of_z_upper))/
((factorial(length(coeff_of_z_upper)-
po)*factorial(po))))*(0.75)*(z_upper_curve(po))^(0.75-1)*(1-
z_upper_curve(po))^0.75;
     z_derive_3_upper=z_derive_3_upper
+(coeff_of_z_upper(po)*(((factorial(length(coeff_of_z_upper))/
(factorial(length(coeff_of_z_upper))*(0.75*(1-
z_upper_curve(po))^(0.75-1)*(z_upper_curve(po))^(0.75)))))));
end
z_derive_lower=z_derive_1_lower+z_derive_2_lower+z_derive_3_lower;
y_lower=atan(z_derive_lower);
y_lower=(y_lower*180/pi);
```

```
thickness_lower_1=thickness_lower*sqrt(1+(tan(y_lower))^2);
z_derive_upper=z_derive_1_upper+z_derive_2_upper+z_derive_3_upper;
y_upper=atan(z_derive_upper);
y_upper=(y_upper*180/pi);
thickness_upper_1=thickness_lower*sqrt(1+(tan(y_upper))^2);
%step_size=thickness_lower_1/n;
%for k=1:length(coeff_of_z_lower)
%theta=[0 45 45 0];
% thickness_of_each_ply=0.1*(ones(1,4));
z_1=(((x).^0.75).*(1-x).^(0.75));
z_2=(((y).^0.75).*(1-y).^(0.75));
z_2_lower=0;
z_2_upper=0;
x=x';
for k=1:length(x)
z_2_lower=z_2_lower+(factorial(length(coeff_of_z_lower))/
(factorial(length(coeff_of_z_lower)-
k)*factorial(k)))*coeff_of_z_lower*(((x).^k).*(ones(1,length(x))-
x).^(length(coeff_of_z_lower)-k))';
end
for k=1:length(coeff_of_z_upper)
z_2_upper=z_2_upper+(factorial(length(coeff_of_z_upper))/
(factorial(length(coeff_of_z_upper)-
k)*factorial(k)))*coeff_of_z_upper*(((y).^k).*(ones(1,length(coeff_of_z_upper))-
y).^(length(coeff_of_z_upper)-k))';
end
z_1=z_1';
z_lower_hope=z_1+z_2_lower;
z_upper_hope=z_2+z_2_upper;
z_upper=z_upper_hope-ones(1,length(y))*(thickness_upper_1/2)-
ones(1,length(y))*h_1;
z_l_lower=z_lower_hope
+(ones(1,length(x))*(thickness_lower_1/2))+ones(1,length(x))*h;
z_upper=[z_upper 0 0];
h=(z_upper-z_l_lower)./n;
integral_final=letsintegrate(z_upper,z_l_lower,sweep,cr,b,gauss_points_x,gauss_poi
for i=1:length(n)
    k=z_l_lower+i*h;
    for lo=1:length(z_l_lower)
    if rem(k(lo),3)==0
        integral_final=integral_final
+2*(function_1(k(lo),sweep,cr,b,gauss_points_x,gauss_points_y));
    else
        integral_final=integral_final
+3*(function_1(k(lo),sweep,cr,b,gauss_points_x,gauss_points_y));
    end
    end
    integral_final=integral_final*3*h(1)/8;
end
[stiffness]=abd_matrix(theta,thickness_of_each_ply);
D_matrix_1=integral_final'*stiffness*integral_final;
function func=function_1(k,sweep,cr,b,gauss_points_x,gauss_points_y)
% sweep=30;
% cr=10;
```

```
% b=20;
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
inverse_trans=inverse';
func=[(inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0
 0 0 k*(inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0
 0 0;0 0 0 (inverse_trans(1,2)+inverse_trans(2,1)) 0
 0 0 0 0 (k)*(inverse_trans(1,2)+inverse_trans(2,2))
 0 0;0 (inverse_trans(1,2)+inverse_trans(2,1))
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0
 0 k*(inverse_trans(2,2)+inverse_trans(2,1))
 k*(inverse_trans(1,1)+inverse_trans(1,2)) 0 0 0];
end
function
 integral=letsintegrate(z_pp,z_l_lower,sweep,cr,b,gauss_points_x,gauss_points_y)
% sweep=30;
% cr=10*10^-3;
% b=20*10^-3;
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
inverse_trans=inverse';
integration_1=zeros(3,12);
integration_2=zeros(3,12);
for i=1:length(z_pp)
integration_1=integration_1+[(inverse_trans(1,1)+inverse_trans(2,1))
 0 0 0 0 0 z_pp(i)*(inverse_trans(1,1)+inverse_trans(2,1))
 0 0 0 0 0;0 0 0 (inverse_trans(1,2)+inverse_trans(2,1)) 0
 0 0 0 0 (z_pp(i))*(inverse_trans(1,2)+inverse_trans(2,2))
 0 0;0 (inverse_trans(1,2)+inverse_trans(2,1))
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0 0
 z_pp(i)*(inverse_trans(2,2)+inverse_trans(2,1))
 z_pp(i)*(inverse_trans(1,1)+inverse_trans(1,2)) 0 0 0];
integration_2=integration_2+[(inverse_trans(1,1)+inverse_trans(2,1))
 0 0 0 0 0 z_l_lower(i)*(inverse_trans(1,1)+inverse_trans(2,1))
 0 0 0 0 0;0 0 0 (inverse_trans(1,2)+inverse_trans(2,1)) 0 0
 0 0 0 (z_l_lower(i))*(inverse_trans(1,2)+inverse_trans(2,2))
 0 0;0 (inverse_trans(1,2)+inverse_trans(2,1))
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0 0
 z_l_lower(i)*(inverse_trans(2,2)+inverse_trans(2,1))
 z_l_lower(i)*(inverse_trans(1,1)+inverse_trans(1,2)) 0 0 0];
end
integral=integration_1+integration_2;
end
end
```

```
Not enough input arguments.

Error in D_matrix_general_in_plane (line 11)
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
```

Published with MATLAB® R2020a

```
function
 D_matrix_trans_1=D_matrix_general_trans_plane(sweep,cr,b,theta,thickness_of_each_
% coeff_of_z_lower=Lower_fitting_a_curve();
% coeff_of_z_upper=Upper_fitting_a_curve();
% sweep=30;
% cr=10*10^-3;
% b=20*10^-3;
% h_1=1;
% h=1;
thickness_lower=0.01;
Z=zeros(5,12);
% gauss_points_x=0;
% gauss_points_y=0;
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
z_2=0;
z_derive_l_lower=0;
z_derive_2_lower=0;
z_derive_3_lower=0;
z_derive_l_upper=0;
z_derive_2_upper=0;
z_derive_3_upper=0;
n=50;
z_lower_l=[0.000000  0.000000;
  0.002000 -0.009300;
  0.005000 -0.016000;
  0.010000 -0.022100;
  0.020000 -0.029500;
  0.030000 -0.034400;
  0.040000 -0.038100;
  0.050000 -0.041200;
  0.070000 -0.046200;
  0.100000 -0.051700;
  0.120000 -0.054700;
  0.150000 -0.058500;
  0.170000 -0.060600;
  0.200000 -0.063300;
  0.220000 -0.064700;
  0.250000 -0.066600;
  0.280000 -0.068000;
  0.300000 -0.068700;
  0.320000 -0.069200;
  0.350000 -0.069600;
  0.370000 -0.069600;
  0.400000 -0.069200;
  0.420000 -0.068800;
  0.450000 -0.067600;
  0.480000 -0.065700;
  0.500000 -0.064400;
  0.530000 -0.061400;
  0.550000 -0.058800;
  0.580000 -0.054300;
  0.600000 -0.050900;
```

```
  0.630000 -0.045100;
  0.650000 -0.041000;
  0.680000 -0.034600;
  0.700000 -0.030200;
  0.730000 -0.023500;
  0.750000 -0.019200;
  0.770000 -0.015000;
  0.800000 -0.009300;
  0.830000 -0.004800;
  0.850000 -0.002400;
  0.870000 -0.001300;
  0.890000 -0.000800;
  0.920000 -0.001600;
  0.940000 -0.003500;
  0.950000 -0.004900;
  0.960000 -0.006600;
  0.970000 -0.008500;
  0.980000 -0.010900;
  0.990000 -0.013700;
  1.000000 -0.016300];
z_upper_curve=[0.000000  0.000000;
  0.002000  0.009500;
  0.005000  0.015800;
  0.010000  0.021900;
  0.020000  0.029300;
  0.030000  0.034300;
  0.040000  0.038100;
  0.050000  0.041100;
  0.070000  0.046200;
  0.100000  0.051800;
  0.120000  0.054800;
  0.150000  0.058500;
  0.170000  0.060600;
  0.200000  0.063200;
  0.220000  0.064600;
  0.250000  0.066400;
  0.270000  0.067300;
  0.300000  0.068500;
  0.330000  0.069200;
  0.350000  0.069600;
  0.380000  0.069800;
  0.400000  0.069700;
  0.430000  0.069500;
  0.450000  0.069200;
  0.480000  0.068400;
  0.500000  0.067800;
  0.530000  0.066600;
  0.550000  0.065600;
  0.570000  0.064500;
  0.600000  0.062500;
  0.620000  0.061000;
  0.650000  0.058500;
  0.680000  0.055500;
  0.700000  0.053300;
```

```
  0.720000  0.050900;
  0.750000  0.046900;
  0.770000  0.043900;
  0.800000  0.038900;
  0.820000  0.035300;
  0.850000  0.029400;
  0.870000  0.025100;
  0.900000  0.018100;
  0.920000  0.013100;
  0.950000  0.004900;
  0.970000 -0.000900;
  0.980000 -0.003900;
  0.990000 -0.007100;
  1.000000 -0.010400;
  ];
x=z_lower_l(1:end,1);
y_1=z_upper_curve(1:end,1);
%z_lower_1=z_lower(1:end,1);
for po=1:length(coeff_of_z_lower)
    z_derive_l_lower=z_derive_l_lower
+(coeff_of_z_lower(po)*(((factorial(length(coeff_of_z_lower))/
(factorial(length(coeff_of_z_lower)-
po)*factorial(po))))*po*(length(coeff_of_z_lower)-
po)*((z_lower_l(po))^(po-1)))*((z_lower_l(po))^0.75)*((1-
z_lower_l(po))^0.75));
    z_derive_2_lower=z_derive_2_lower
+(coeff_of_z_lower(po))*(factorial(length(coeff_of_z_lower))/
((factorial(length(coeff_of_z_lower)-
po)*factorial(po))))*(0.75)*(z_lower_l(po))^(0.75-1)*(1-
z_lower_l(po))^0.75;
     z_derive_3_lower=z_derive_3_lower
+(coeff_of_z_lower(po)*(((factorial(length(coeff_of_z_lower))/
(factorial(length(coeff_of_z_lower))*(0.75*(1-
z_lower_l(po))^(0.75-1)*(z_lower_l(po))^(0.75)))))));
end
for po=1:length(coeff_of_z_upper)
    z_derive_l_upper=z_derive_l_upper
+(coeff_of_z_upper(po)*(((factorial(length(coeff_of_z_upper))/
(factorial(length(coeff_of_z_upper)-
po)*factorial(po))))*po*(length(coeff_of_z_upper)-
po)*((z_upper_curve(po))^(po-1)))*((z_upper_curve(po))^0.75)*((1-
z_upper_curve(po))^0.75));
    z_derive_2_upper=z_derive_2_upper
+(coeff_of_z_upper(po))*(factorial(length(coeff_of_z_upper))/
((factorial(length(coeff_of_z_upper)-
po)*factorial(po))))*(0.75)*(z_upper_curve(po))^(0.75-1)*(1-
z_upper_curve(po))^0.75;
     z_derive_3_upper=z_derive_3_upper
+(coeff_of_z_upper(po)*(((factorial(length(coeff_of_z_upper))/
(factorial(length(coeff_of_z_upper))*(0.75*(1-
z_upper_curve(po))^(0.75-1)*(z_upper_curve(po))^(0.75)))))));
end
z_derive_lower=z_derive_l_lower+z_derive_2_lower+z_derive_3_lower;
y=atan(z_derive_lower);
```

```
y=(y*180/pi);
thickness_lower_1=thickness_lower*sqrt(1+((tan(y)))^2);
z_derive_upper=z_derive_l_upper+z_derive_2_upper+z_derive_3_upper;
y_upper=atan(z_derive_upper);
y_upper=(y_upper*180/pi);
thickness_upper_1=thickness_lower*sqrt(1+(tan(y_upper))^2);
%step_size=thickness_lower_1/n;
%for k=1:length(coeff_of_z_lower)
% theta=[0 90 90 0];
% thickness_of_each_ply=0.1*(ones(1,4));
z_1=(((x).^0.75).*(1-x).^(0.75));
z_2=(((y_1).^0.75).*(1-y_1).^(0.75));
z_2_lower=0;
z_2_upper=0;
x=x';
y_1=y_1';
for k=1:length(x)
z_2_lower=z_2_lower+(factorial(length(coeff_of_z_lower))/
(factorial(length(coeff_of_z_lower)-
k)*factorial(k)))*coeff_of_z_lower*(((x).^k).*(ones(1,length(x))-
x).^(length(coeff_of_z_lower)-k))';
end
for k=1:length(y_1)
z_2_upper=z_2_upper+(factorial(length(coeff_of_z_upper))/
(factorial(length(coeff_of_z_upper)-
k)*factorial(k)))*coeff_of_z_upper*(((y_1).^k).*(ones(1,length(coeff_of_z_upper))-
y_1).^(length(coeff_of_z_upper)-k))';
end
z_1=z_1';
z_lower_hope=z_1+z_2_lower;
z_upper_hope=z_2+z_2_upper;
z_upper=z_upper_hope-(ones(1,length(y))*(thickness_upper_1/2))-
(ones(1,length(y))*h_1);
z_l_lower=z_lower_hope
+((ones(1,length(x))*(thickness_lower_1/2)))+(ones(1,length(x))*h);
z_upper=z_upper';
z_upper=[z_upper 0 0];
h_1=abs((z_upper-z_l_lower))./n;
integral_final=letsintegrate(z_upper,z_l_lower,sweep,cr,b,gauss_points_x,gauss_poi
for i=1:length(n)
    k=z_l_lower+i*h_1;
    for lo=1:length(z_l_lower)
    if rem(k(lo),3)==0
        integral_final=integral_final
+2*(function_1(k(lo),sweep,cr,b,gauss_points_x,gauss_points_y));
    else
        integral_final=integral_final
+3*(function_1(k(lo),sweep,cr,b,gauss_points_x,gauss_points_y));
    end
    end
    integral_final=integral_final*3*h_1(1)/8;
end
[stiffness]=abd_matrix_trans(theta,thickness_of_each_ply);
D_matrix_trans_1=integral_final'*stiffness*integral_final;
```

```
function func=function_1(k,sweep,cr,b,gauss_points_x,gauss_points_y)
% sweep=30;
% cr=10;
% b=20;
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
inverse_trans=inverse';
func=[0 0 0 0 0 (inverse_trans(1,2)+inverse_trans(2,2)) 0 0 0 0 0 1;0
 0 0 0 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0 0 0 1 0];
end
function
 integral=letsintegrate(z_pp,z_l_lower,sweep,cr,b,gauss_points_x,gauss_points_y)
% sweep=30;
% cr=10*10^-3;
% b=20*10^-3;
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
inverse_trans=inverse';
integration_1=zeros(2,12);
integration_2=zeros(2,12);
for i=1:length(z_pp)
integration_1=integration_1+[0 0 0 0 0
 (inverse_trans(1,2)+inverse_trans(2,2)) 0 0 0 0 0 1;0 0 0 0
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0 0 0 1 0];
integration_2=integration_2+[0 0 0 0 0
 (inverse_trans(1,2)+inverse_trans(2,2)) 0 0 0 0 0 1;0 0 0 0
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0 0 0 1 0];
end
integral=integration_1+integration_2;
end
end
```

```
Not enough input arguments.

Error in D_matrix_general_trans_plane (line 13)
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
```

Published with MATLAB® R2020a

```
function
 spar_web_Stiffness=FEM_Main_file_spars_web(gauss_points_in_x,gauss_points_in_y,ga
% gauss_points_in_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_trans_x=0;
% gauss_points_in_y=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_trans_y=0;
% weights_in_x=[1 1];
% weights_in_y=[1 1];
% weights_trans_x=2;
% weights_trans_y=2;
% sweep=30;%degree
% b=10;
% cr=5;
% I=3;
% J=3;
% K=3;
% L=3;
% M=3;
% N=3;
% P=3;
% Q=3;
% R=3;
% S=3;
% l_1=1;
% %c=1;
% h=1;
% h_1=1;
% t_1=1;
% theta=[0 90 90 0];
% thickness_of_each_ply=1*ones(1,length(theta));
B_in_plane=B_matrix_in_plane_spar_web(gauss_points_in_x,gauss_points_in_y,sweep,b,
B_trans_plane=B_matrix_trans_plane_spar_web(gauss_points_trans_x,gauss_points_tran
spar_web_Stiffness=B_in_plane+B_trans_plane;
end
```

```
Not enough input arguments.

Error in FEM_Main_file_spars_web (line 30)
B_in_plane=B_matrix_in_plane_spar_web(gauss_points_in_x,gauss_points_in_y,sweep,b,
```

Published with MATLAB® R2020a

```
function curve_fit_Lower=Lower_fitting_a_curve()
z_lower=[0.000000  0.000000;
  0.002000 -0.009300;
  0.005000 -0.016000;
  0.010000 -0.022100;
  0.020000 -0.029500;
  0.030000 -0.034400;
  0.040000 -0.038100;
  0.050000 -0.041200;
  0.070000 -0.046200;
  0.100000 -0.051700;
  0.120000 -0.054700;
  0.150000 -0.058500;
  0.170000 -0.060600;
  0.200000 -0.063300;
  0.220000 -0.064700;
  0.250000 -0.066600;
  0.280000 -0.068000;
  0.300000 -0.068700;
  0.320000 -0.069200;
  0.350000 -0.069600;
  0.370000 -0.069600;
  0.400000 -0.069200;
  0.420000 -0.068800;
  0.450000 -0.067600;
  0.480000 -0.065700;
  0.500000 -0.064400;
  0.530000 -0.061400;
  0.550000 -0.058800;
  0.580000 -0.054300;
  0.600000 -0.050900;
  0.630000 -0.045100;
  0.650000 -0.041000;
  0.680000 -0.034600;
  0.700000 -0.030200;
  0.730000 -0.023500;
  0.750000 -0.019200;
  0.770000 -0.015000;
  0.800000 -0.009300;
  0.830000 -0.004800;
  0.850000 -0.002400;
  0.870000 -0.001300;
  0.890000 -0.000800;
  0.920000 -0.001600;
  0.940000 -0.003500;
  0.950000 -0.004900;
  0.960000 -0.006600;
  0.970000 -0.008500;
  0.980000 -0.010900;
  0.990000 -0.013700;
  1.000000 -0.016300];
x_guess=ones(1,length(z_lower));
```

```
airfoil_equation_lower_curve_2=@airfoil_equation_lower_curve;
curve_fit_Lower=lsqcurvefit(airfoil_equation_lower_curve_2,x_guess,z_lower(1:end,1
z_predict=z_lower(1:end,1)';
thickness_ratio_lower_surface=0.01;
S_predict=zeros(length(z_lower),length(z_lower));
for i=1:length(z_predict)
   S_predict(i,1:length(z_predict))=(factorial(length(z_lower))/
(factorial(length(z_lower)-
i)*factorial(i)))*(((z_predict).^i).*((ones(1,length(z_predict))-
z_predict).^(length(z_predict)-i)));
end
% predicited_graph=((z_predict).^0.75).*((ones(1,length(z_predict))-
z_predict).^0.75).*(curve_fit_Lower*S_predict)+ones(1,length(z_predict))*thickness
% plot(z_lower(1:end,1)',predicited_graph,"b")
% hold on
% plot(z_lower(1:end,1),z_lower(1:end,2),"bo")
% hold off
function z=airfoil_equation_lower_curve(x,x_data)
x_data=x_data';
thickness_ratio_lower_surface=0.01;
S=zeros(length(x_data),length(x_data));
for i=1:length(x_data)
   S(i,1:length(x_data))=(factorial(length(x_data))/
(factorial(length(x_data)-
i)*factorial(i)))*(((x_data).^i).*((ones(1,length(x_data))-
x_data).^(length(x_data)-i)));
end
matrix=zeros(1,length(x_data));
for i=1:length(x_data)
    matrix(1,i)=x(i);
end
z=((x_data).^0.75).*((ones(1,length(x_data))-
x_data).^0.75).*(matrix*S)+ones(1,length(x_data))*thickness_ratio_lower_surface;
end
end
```

```
Local minimum found.

Optimization completed because the size of the gradient is less than
the value of the optimality tolerance.


ans =

  Columns 1 through 7

   -6.8476   16.4317  -26.3682   19.0586    1.1947  -12.2233   -0.0915

  Columns 8 through 14

    7.5818    1.1945   -5.4205   -3.4790    2.0104    3.4490    0.2375

  Columns 15 through 21
```

```
  -2.8556   -2.4605    0.2447    1.9473    1.0842   -0.9910   -1.9690

Columns 22 through 28

  -1.0438    0.5741    1.1811    0.3165   -0.9685   -1.3421   -0.5335

Columns 29 through 35

   0.5210    0.7394   -0.0024   -0.8314   -0.8664   -0.1462    0.4839

Columns 36 through 42

   0.3620   -0.2620   -0.5966   -0.3027    0.1513    0.2203    0.0051

Columns 43 through 49

  -0.1396   -0.2629   -0.2347    0.4094    0.2315   -1.3475    1.2399

Column 50

  -1.7220
```

Published with MATLAB® R2020a

```
function C=ques_1_compliance()
% prompt="Enter the value of E";
% prompt_1="Enter the value of poisson's ratio";
% prompt_2="Enter the value of G";
E1=70;
E2=70;
%E3=input(prompt);
poisson_ratio_12=0.1;
%poisson_ratio_13=input(prompt_1);
%poisson_ratio_23=input(prompt_1);
G12=5;
%G13=input(prompt_2);
%G23=input(prompt_2);
S=zeros(3,3);
S(1,1)=1/E1;
S(1,2)=-poisson_ratio_12/E2;
%S(1,3)=poisson_ratio_13/E3;
S(2,1)=-poisson_ratio_12/E1;
S(2,2)=1/E2;
%S(2,3)=poisson_ratio_23/E3;
%S(3,1)=poisson_ratio_13/E1;
%S(3,2)=poisson_ratio_23/E2;
%S(3,3)=1/E3;
%S(4,4)=1/G23;
%S(5,5)=1/G13;
S(3,3)=1/G12;
C=inv(S);
end
```

```
ans =

   70.7071    7.0707         0
    7.0707   70.7071         0
         0         0    5.0000
```

Published with MATLAB® R2020a

```
function C=ques_1_compliance_trans_plane()
% prompt="Enter the value of E";
% prompt_1="Enter the value of poisson's ratio";
% prompt_2="Enter the value of G";
%E1=15;
%E2=20;
%E3=input(prompt);
%poisson_ratio_12=0.3;
%poisson_ratio_13=input(prompt_1);
%poisson_ratio_23=input(prompt_1);
%G12=5;
G13=5;
G23=5;
S=zeros(2,2);
S(1,1)=1/G13;
S(1,2)=0;
%S(1,3)=poisson_ratio_13/E3;
S(2,1)=0;
S(2,2)=1/G23;
%S(2,3)=poisson_ratio_23/E3;
%S(3,1)=poisson_ratio_13/E1;
%S(3,2)=poisson_ratio_23/E2;
%S(3,3)=1/E3;
%S(4,4)=1/G23;
%S(5,5)=1/G13;
%S(3,3)=1/G12;
C=inv(S);
end
```

```
ans =

     5     0
     0     5
```

Published with MATLAB® R2020a

```
function Q=ques_2_Q_matrix_in_plane()
Stiffness=ques_1_compliance_in_plane();
Q=zeros(3,3);
Q(1,1)=Stiffness(1,1);%-(Stiffness(1,3)^2/Stiffness(3,3));
Q(1,2)=Stiffness(1,2);%-((Stiffness(1,3)*Stiffness(2,3))/
Stiffness(3,3));
Q(2,2)=Stiffness(2,2);%-(Stiffness(2,3)^2/Stiffness(3,3));
Q(3,3)=Stiffness(3,3);
Q(2,1)=Q(1,2);
end
```

```
ans =

   70.7071    7.0707         0
    7.0707   70.7071         0
         0         0    5.0000
```

Published with MATLAB® R2020a

```
function Q=ques_2_Q_matrix_trans_plane()
Stiffness=ques_1_compliance_trans_plane();
Q=Stiffness;
% Q(1,1)=Stiffness(1,1);%-(Stiffness(1,3)^2/Stiffness(3,3));
% Q(1,2)=Stiffness(1,2);%-((Stiffness(1,3)*Stiffness(2,3))/
Stiffness(3,3));
% Q(2,2)=Stiffness(2,2);%-(Stiffness(2,3)^2/Stiffness(3,3));
% Q(3,3)=Stiffness(3,3);
% Q(2,1)=Q(1,2);
end
```

```
ans =

     5     0
     0     5
```

Published with MATLAB® R2020a

```
function C=ques_3_with_arguments_transformed(theta)
%prompt="Enter the value of theta in degree";
%theta=input(prompt);
theta=(pi*theta)/180;
C=zeros(3,3);
C(1,1)=cos(theta)^2;
C(1,2)=sin(theta)^2;
C(1,3)=2*cos(theta)*sin(theta);
C(2,1)=sin(theta)^2;
C(2,2)=cos(theta)^2;
C(2,3)=-2*cos(theta)*sin(theta);
C(3,1)=-cos(theta)*sin(theta);
C(3,2)=cos(theta)*sin(theta);
C(3,3)=cos(theta)^2-sin(theta)^2;
end
```

```
Not enough input arguments.

Error in ques_3_with_arguments_transformed_in_plane (line 4)
theta=(pi*theta)/180;
```

Published with MATLAB® R2020a

```
function C=ques_3_with_arguments_transformed_trans_plane(theta)
%prompt="Enter the value of theta in degree";
%theta=input(prompt);
%theta=30;
theta=(pi*theta)/180;
C=zeros(2,2);
C(1,1)=cos(theta);
C(1,2)=-sin(theta);
%C(1,3)=2*cos(theta)*sin(theta);
C(2,1)=-sin(theta);
C(2,2)=cos(theta);
% C(2,3)=-2*cos(theta)*sin(theta);
% C(3,1)=-cos(theta)*sin(theta);
% C(3,2)=cos(theta)*sin(theta);
% C(3,3)=cos(theta)^2-sin(theta)^2;
end
```

```
Not enough input arguments.

Error in ques_3_with_arguments_transformed_trans_plane (line 5)
theta=(pi*theta)/180;
```

Published with MATLAB® R2020a

```
function curve_fit_upper=Upper_fitting_a_curve()
z_upper=[0.000000  0.000000;
  0.002000  0.009500;
  0.005000  0.015800;
  0.010000  0.021900;
  0.020000  0.029300;
  0.030000  0.034300;
  0.040000  0.038100;
  0.050000  0.041100;
  0.070000  0.046200;
  0.100000  0.051800;
  0.120000  0.054800;
  0.150000  0.058500;
  0.170000  0.060600;
  0.200000  0.063200;
  0.220000  0.064600;
  0.250000  0.066400;
  0.270000  0.067300;
  0.300000  0.068500;
  0.330000  0.069200;
  0.350000  0.069600;
  0.380000  0.069800;
  0.400000  0.069700;
  0.430000  0.069500;
  0.450000  0.069200;
  0.480000  0.068400;
  0.500000  0.067800;
  0.530000  0.066600;
  0.550000  0.065600;
  0.570000  0.064500;
  0.600000  0.062500;
  0.620000  0.061000;
  0.650000  0.058500;
  0.680000  0.055500;
  0.700000  0.053300;
  0.720000  0.050900;
  0.750000  0.046900;
  0.770000  0.043900;
  0.800000  0.038900;
  0.820000  0.035300;
  0.850000  0.029400;
  0.870000  0.025100;
  0.900000  0.018100;
  0.920000  0.013100;
  0.950000  0.004900;
  0.970000 -0.000900;
  0.980000 -0.003900;
  0.990000 -0.007100;
  1.000000 -0.010400;
  ];
x_guess=ones(1,length(z_upper));
airfoil_equation_lower_curve_2=@airfoil_equation_lower_curve;
```

```
curve_fit_upper=lsqcurvefit(airfoil_equation_lower_curve_2,x_guess,z_upper(1:end,1
z_predict=z_upper(1:end,1)';
thickness_ratio_lower_surface=0.01;
S_predict=zeros(length(z_upper),length(z_upper));
for i=1:length(z_predict)
   S_predict(i,1:length(z_predict))=(factorial(length(z_upper))/
(factorial(length(z_upper)-
i)*factorial(i)))*(((z_predict).^i).*((ones(1,length(z_predict))-
z_predict).^(length(z_predict)-i)));
end
% predicited_graph=((z_predict).^0.75).*((ones(1,length(z_predict))-
z_predict).^0.75).*(curve_fit_upper*S_predict)+ones(1,length(z_predict))*thickness
% plot(z_upper(1:end,1)',predicited_graph,"b")
% hold on
% plot(z_upper(1:end,1),z_upper(1:end,2),"bo")
% hold off
function z=airfoil_equation_lower_curve(x,x_data)
x_data=x_data';
thickness_ratio_lower_surface=0.01;
S=zeros(length(x_data),length(x_data));
for i=1:length(x_data)
   S(i,1:length(x_data))=(factorial(length(x_data))/
(factorial(length(x_data)-
i)*factorial(i)))*(((x_data).^i).*((ones(1,length(x_data))-
x_data).^(length(x_data)-i)));
end
matrix=zeros(1,length(x_data));
for i=1:length(x_data)
    matrix(1,i)=x(i);
end
z=((x_data).^0.75).*((ones(1,length(x_data))-
x_data).^0.75).*(matrix*S)+ones(1,length(x_data))*thickness_ratio_lower_surface;
end
end
```

Local minimum found.

Optimization completed because the size of the gradient is less than the value of the optimality tolerance.

ans =

Columns 1 through 7

1.4012 -1.1876 1.2572 0.4493 -0.2935 0.0879 0.5228

Columns 8 through 14

0.4273 0.1085 -0.0065 0.1345 0.3086 0.3318 0.2166

Columns 15 through 21

```
    0.0980    0.0795    0.1532    0.2356    0.2548    0.2051    0.1366

  Columns 22 through 28

    0.1040    0.1243    0.1720    0.2059    0.2026    0.1689    0.1313

  Columns 29 through 35

    0.1141    0.1246    0.1522    0.1777    0.1829    0.1583    0.1120

  Columns 36 through 42

    0.0756    0.0893    0.1579    0.2111    0.1484   -0.0132   -0.0575

  Columns 43 through 48

    0.1703    0.3062   -0.1698   -0.3775    0.6231   -1.1358
```

Published with MATLAB® R2020a

5.3 Skin code

```
function
 kg_in_plane=B_matrix_in_plane(gauss_points_x,gauss_points_y,sweep_angle,b,cr,I,J,
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
% weights_x=[1 1];
% weights_y=[1 1];
% sweep_angle=30;%degree
% b=10;
% %sweep_angle_in_radians=sweep_angle*pi/180;
% cr=5;
%tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%Linear nodes per element
% I=2;
% J=2;
% K=2;
% L=2;
% M=2;
% N=2;
% P=2;
% Q=2;
% R=2;
% S=2;
% dof=(5);
% icon_IJ=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_KL=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_PQ=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_RS=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
coeff_of_z_lower=Lower_fitting_a_curve();
coeff_of_z_upper=Upper_fitting_a_curve();
number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
%number_of_elements=number_of_nodes-1;
P_non_derive_I_J_x=zeros(1,I*J);
P_derive_I_J_x=zeros(1,I*J);
P_derive_K_L_x=zeros(1,K*L);
P_non_derive_K_L_x=zeros(1,K*L);
P_derive_M_N_x=zeros(1,M*N);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_derive_I_J_y=zeros(1,I*J);
P_derive_K_L_y=zeros(1,K*L);
P_non_derive_K_L_y=zeros(1,K*L);
P_derive_M_N_y=zeros(1,M*N);
P_non_derive_M_N_y=zeros(1,M*N);
```

```
P_non_derive_P_Q_y=zeros(1,P*Q);
P_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_derive_R_S_y=zeros(1,R*S);
B=zeros(12,(I*J+P*Q+M*N+K*L+R*S));
% theta=[0 90 90 0];
% thickness_of_each_ply=0.01*ones(1,length(theta));
hope=zeros((I*J+P*Q+M*N+K*L+R*S),(I*J+P*Q+M*N+K*L+R*S));
%kg=zeros(((I*J)+(K*L)+(P*Q)+(R*S))*dof,
((I*J)+(K*L)+(P*Q)+(R*S))*dof);
for
 i=1:I*J%*length(icon_KL(:,1))*length(icon_KL(1,:))*length(icon_PQ(:,1))*length(ic
%     n11_IJ=icon_IJ(i,1);
%     n22_IJ=icon_IJ(i,2);
%     n33_IJ=icon_IJ(i,3);
%     n44_IJ=icon_IJ(i,4);
%     %n55_IJ=icon_IJ(i,5);
%     n11_KL=icon_KL(i,1);
%     n22_KL=icon_KL(i,2);
%     n33_KL=icon_KL(i,3);
%     n44_KL=icon_KL(i,4);
%     %n55_KL=icon_KL(i,5);
%     n11_PQ=icon_PQ(i,1);
%     n22_PQ=icon_PQ(i,2);
%     n33_PQ=icon_PQ(i,3);
%     n44_PQ=icon_PQ(i,4);
%     %n55_PQ=icon_PQ(i,5);
%     n11_RS=icon_RS(i,1);
%     n22_RS=icon_RS(i,2);
%     n33_RS=icon_RS(i,3);
%     n44_RS=icon_RS(i,4);
    %n55_RS=icon_RS(i,5);
    for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
P_non_derive_I_J_x(1,1)=1;
P_non_derive_I_J_x(1,2)=gauss_points_x(j);
P_derive_I_J_x(1,1)=0;
P_derive_I_J_x(1,2)=1;
P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=gauss_points_x(j);
P_derive_M_N_x(1,1)=0;
P_derive_M_N_x(1,2)=1;
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_derive_M_N_x(1,1)=0;
P_derive_M_N_x(1,2)=1;
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=gauss_points_x(j);
P_derive_P_Q_x(1,1)=0;
P_derive_P_Q_x(1,2)=1;
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=gauss_points_x(j);
P_derive_R_S_x(1,1)=0;
P_derive_R_S_x(1,2)=1;
```

```
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=gauss_points_y(op);
P_derive_I_J_y(1,1)=0;
P_derive_I_J_y(1,2)=1;
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=gauss_points_y(op);
P_derive_M_N_y(1,1)=0;
P_derive_M_N_y(1,2)=1;
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_derive_M_N_y(1,1)=0;
P_derive_M_N_y(1,2)=1;
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_derive_P_Q_y(1,1)=0;
P_derive_P_Q_y(1,2)=1;
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=gauss_points_y(op);
P_derive_R_S_y(1,1)=0;
P_derive_R_S_y(1,2)=1;
        for k=3:number_of_nodes_I_J
           P_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_I_J_x(1,k-1)-(k/(k
+1)*P_derive_I_J_x(1,k-2)));
           P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2)));
        end
        for k=3:number_of_nodes_K_L
           P_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_K_L_x(1,k-1)-(k/(k
+1)*P_derive_K_L_x(1,k-2)));
           P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
        end
%         for k=3:number_of_nodes_M_N
%            P_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_M_N_x(1,k-1)-(k/(k
+1)*P_derive_M_N_x(1,k-2)));
%            P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
%         end
        for k=3:number_of_nodes_P_Q
           P_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_derive_P_Q_x(1,k-2)));
           P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
        end
        for k=3:number_of_nodes_R_S
```

```
            P_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_R_S_x(1,k-1)-(k/(k
+1)*P_derive_R_S_x(1,k-2)));
            P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
        end
        for k=3:number_of_nodes_I_J
            P_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(j))*P_derive_I_J_y(1,k-1)-(k/(k
+1)*P_derive_I_J_y(1,k-2)));
            P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
        end
        for k=3:number_of_nodes_K_L
            P_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(j))*P_derive_K_L_x(1,k-1)-(k/(k
+1)*P_derive_K_L_x(1,k-2)));
            P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
        end
%         for k=3:number_of_nodes_M_N
%             P_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(j))*P_derive_M_N_y(1,k-1)-(k/(k
+1)*P_derive_M_N_y(1,k-2)));
%             P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
%         end
        for k=3:number_of_nodes_P_Q
            P_derive_P_Q_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(j))*P_derive_P_Q_y(1,k-1)-(k/(k
+1)*P_derive_P_Q_y(1,k-2)));
            P_non_derive_P_Q_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_P_Q_y(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y(1,k-2)));
        end
        for k=3:number_of_nodes_R_S
            P_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(j))*P_derive_R_S_y(1,k-1)-(k/(k
+1)*P_derive_R_S_y(1,k-2)));
            P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
        end

 [detJacobian,inverse]=coordinate_transformation_Jacobian(sweep_angle,cr,b,gauss_p
        inverse_trans=inverse';
        final_IJ_x=P_derive_I_J_x.*P_non_derive_I_J_x;
        final_KL_x=P_derive_K_L_x.*P_non_derive_K_L_x;
       % final_MN_x=P_derive_M_N_x.*P_non_derive_M_N_x;
        final_PQ_x=P_derive_P_Q_x.*P_non_derive_P_Q_x;
```

```
        final_RS_x=P_derive_R_S_x.*P_non_derive_R_S_x;
        final_IJ_y=P_derive_I_J_y.*P_non_derive_I_J_y;
        final_KL_y=P_derive_K_L_y.*P_non_derive_K_L_y;
        %final_MN_y=P_derive_M_N_y.*P_non_derive_M_N_y;
        final_PQ_y=P_derive_P_Q_y.*P_non_derive_P_Q_y;
        final_RS_y=P_derive_R_S_y.*P_non_derive_R_S_y;
        boom=5;
        idk=1;
%         final_IJ=[final_IJ_x;final_IJ_y];
%         final_KL=[final_KL_x;final_KL_y];
%         final_MN=[final_MN_x;final_MN_y];
%         final_PQ=[final_PQ_x;final_PQ_y];
%         final_RS=[final_RS_x;final_RS_y];
        for pi=1:length(final_IJ_x)
            B(1:12,idk:boom)=B(1:12,idk:boom)+[final_IJ_x(1,pi)
 0 0 0 0;final_IJ_y(1,pi) 0 0 0 0;0 final_KL_x(1,pi) 0 0 0;0
 final_KL_y(1,pi) 0 0 0;0 0 0 0 0;0 0 0 0 0;0 0 0 final_PQ_x(1,pi)
 0;0 0 0 final_PQ_y(1,pi) 0;0 0 0 0 final_RS_x(1,pi);0 0 0 0
 final_RS_y(1,pi);0 0 0 0 0;0 0 0 0 0];
            idk=idk+5;
            boom=boom+5;
        end

 D=D_matrix_lower_airfoil_in_plane(sweep_angle,cr,b,theta,thickness_of_each_ply,ga
        hope=hope+B'*D*B*weights_x(j)*weights_y(op)*detJacobian;
    end
    end
%     iv_0=[5*n11_IJ-4 5*n11_IJ-3 5*n11_IJ-2 5*n11_IJ-1 5*n11_IJ
 5*n22_IJ-4 5*n22_IJ-3 5*n22_IJ-2 5*n22_IJ-1 5*n22_IJ 5*n33_IJ-4
 5*n33_IJ-3 5*n33_IJ-2 5*n33_IJ-1 5*n33_IJ 5*n44_IJ-4 5*n44_IJ-3
 5*n44_IJ-2 5*n44_IJ-1 5*n44_IJ];
%     iv_1=[5*n11_KL-4 5*n11_KL-3 5*n11_KL-2 5*n11_KL-1 5*n11_KL
 5*n22_KL-4 5*n22_KL-3 5*n22_KL-2 5*n22_KL-1 5*n22_KL 5*n33_KL-4
 5*n33_KL-3 5*n33_KL-2 5*n33_KL-1 5*n33_KL 5*n44_KL-4 5*n44_KL-3
 5*n44_KL-2 5*n44_KL-1 5*n44_KL];
%     iv_2=[5*n11_PQ-4 5*n11_PQ-3 5*n11_PQ-2 5*n11_PQ-1 5*n11_PQ
 5*n22_PQ-4 5*n22_PQ-3 5*n22_PQ-2 5*n22_PQ-1 5*n22_PQ 5*n33_PQ-4
 5*n33_PQ-3 5*n33_PQ-2 5*n33_PQ-1 5*n33_PQ 5*n44_PQ-4 5*n44_PQ-3
 5*n44_PQ-2 5*n44_PQ-1 5*n44_PQ];
%     iv_3=[5*n11_RS-4 5*n11_RS-3 5*n11_RS-2 5*n11_RS-1 5*n11_RS
 5*n22_RS-4 5*n22_RS-3 5*n22_RS-2 5*n22_RS-1 5*n22_RS 5*n33_RS-4
 5*n33_RS-3 5*n33_RS-2 5*n33_RS-1 5*n33_RS 5*n44_RS-4 5*n44_RS-3
 5*n44_RS-2 5*n44_RS-1 5*n44_RS];
%     iv=[iv_0 iv_1 iv_2 iv_3];
    %kg(iv,iv)=kg(iv,iv)+hope;
end
kg_in_plane=hope;
end
```

Local minimum found.

Optimization completed because the size of the gradient is less than the value of the optimality tolerance.

```
Local minimum found.

Optimization completed because the size of the gradient is less than
the value of the optimality tolerance.


Not enough input arguments.

Error in B_matrix_in_plane_skin (line 29)
number_of_nodes_I_J=((I*J));
```

Published with MATLAB® R2020a

```
function
 kg_trans_plane=B_matrix_trans_plane(gauss_points_x,gauss_points_y,sweep_angle,b,c
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
% weights_x=[1 1];
% weights_y=[1 1];
% sweep_angle=30;%degree
% b=10;
% %sweep_angle_in_radians=sweep_angle*pi/180;
% cr=5;
%tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%Linear nodes per element
% I=2;
% J=2;
% K=2;
% L=2;
% M=2;
% N=2;
% P=2;
% Q=2;
% R=2;
% S=2;
% dof=(5);
% icon_IJ=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_KL=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_PQ=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_RS=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
coeff_of_z_lower=Lower_fitting_a_curve();
coeff_of_z_upper=Upper_fitting_a_curve();
number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
%number_of_elements=number_of_nodes-1;
P_non_derive_I_J_x=zeros(1,I*J);
P_derive_I_J_x=zeros(1,I*J);
P_derive_K_L_x=zeros(1,K*L);
P_non_derive_K_L_x=zeros(1,K*L);
P_derive_M_N_x=zeros(1,M*N);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_derive_I_J_y=zeros(1,I*J);
P_derive_K_L_y=zeros(1,K*L);
P_non_derive_K_L_y=zeros(1,K*L);
P_derive_M_N_y=zeros(1,M*N);
P_non_derive_M_N_y=zeros(1,M*N);
```

```
P_non_derive_P_Q_y=zeros(1,P*Q);
P_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_derive_R_S_y=zeros(1,R*S);
B=zeros(12,(I*J+P*Q+M*N+K*L+R*S));
% theta=[0 90 90 0];
% thickness_of_each_ply=0.01*ones(1,length(theta));
hope=zeros((I*J+P*Q+M*N+K*L+R*S),(I*J+P*Q+M*N+K*L+R*S));
%kg=zeros(((I*J)+(K*L)+(P*Q)+(R*S))*dof,
((I*J)+(K*L)+(P*Q)+(R*S))*dof);
for
 i=1:I*J%*length(icon_KL(:,1))*length(icon_KL(1,:))*length(icon_PQ(:,1))*length(ic
%     n11_IJ=icon_IJ(i,1);
%     n22_IJ=icon_IJ(i,2);
%     n33_IJ=icon_IJ(i,3);
%     n44_IJ=icon_IJ(i,4);
%     %n55_IJ=icon_IJ(i,5);
%     n11_KL=icon_KL(i,1);
%     n22_KL=icon_KL(i,2);
%     n33_KL=icon_KL(i,3);
%     n44_KL=icon_KL(i,4);
%     %n55_KL=icon_KL(i,5);
%     n11_PQ=icon_PQ(i,1);
%     n22_PQ=icon_PQ(i,2);
%     n33_PQ=icon_PQ(i,3);
%     n44_PQ=icon_PQ(i,4);
%     %n55_PQ=icon_PQ(i,5);
%     n11_RS=icon_RS(i,1);
%     n22_RS=icon_RS(i,2);
%     n33_RS=icon_RS(i,3);
%     n44_RS=icon_RS(i,4);
    %n55_RS=icon_RS(i,5);
    for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
P_non_derive_I_J_x(1,1)=1;
P_non_derive_I_J_x(1,2)=gauss_points_x(j);
P_derive_I_J_x(1,1)=0;
P_derive_I_J_x(1,2)=1;
P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=gauss_points_x(j);
P_derive_M_N_x(1,1)=0;
P_derive_M_N_x(1,2)=1;
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_derive_M_N_x(1,1)=0;
P_derive_M_N_x(1,2)=1;
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=gauss_points_x(j);
P_derive_P_Q_x(1,1)=0;
P_derive_P_Q_x(1,2)=1;
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=gauss_points_x(j);
P_derive_R_S_x(1,1)=0;
P_derive_R_S_x(1,2)=1;
```

```
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=gauss_points_y(op);
P_derive_I_J_y(1,1)=0;
P_derive_I_J_y(1,2)=1;
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=gauss_points_y(op);
P_derive_M_N_y(1,1)=0;
P_derive_M_N_y(1,2)=1;
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_derive_M_N_y(1,1)=0;
P_derive_M_N_y(1,2)=1;
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_derive_P_Q_y(1,1)=0;
P_derive_P_Q_y(1,2)=1;
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=gauss_points_y(op);
P_derive_R_S_y(1,1)=0;
P_derive_R_S_y(1,2)=1;
        for k=3:number_of_nodes_I_J
           P_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_I_J_x(1,k-1)-(k/(k
+1)*P_derive_I_J_x(1,k-2)));
           P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2)));
        end
        for k=3:number_of_nodes_K_L
           P_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_K_L_x(1,k-1)-(k/(k
+1)*P_derive_K_L_x(1,k-2)));
           P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
        end
        for k=3:number_of_nodes_M_N
           P_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_M_N_x(1,k-1)-(k/(k
+1)*P_derive_M_N_x(1,k-2)));
           P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
        end
        for k=3:number_of_nodes_P_Q
           P_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_derive_P_Q_x(1,k-2)));
           P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
        end
        for k=3:number_of_nodes_R_S
```

```
            P_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_R_S_x(1,k-1)-(k/(k
+1)*P_derive_R_S_x(1,k-2)));
            P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
        end
        for k=3:number_of_nodes_I_J
            P_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(j))*P_derive_I_J_y(1,k-1)-(k/(k
+1)*P_derive_I_J_y(1,k-2)));
            P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
        end
        for k=3:number_of_nodes_K_L
            P_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(j))*P_derive_K_L_x(1,k-1)-(k/(k
+1)*P_derive_K_L_x(1,k-2)));
            P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
        end
        for k=3:number_of_nodes_M_N
            P_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(j))*P_derive_M_N_y(1,k-1)-(k/(k
+1)*P_derive_M_N_y(1,k-2)));
            P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
        end
        for k=3:number_of_nodes_P_Q
            P_derive_P_Q_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(j))*P_derive_P_Q_y(1,k-1)-(k/(k
+1)*P_derive_P_Q_y(1,k-2)));
            P_non_derive_P_Q_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_P_Q_y(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y(1,k-2)));
        end
        for k=3:number_of_nodes_R_S
            P_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(j))*P_derive_R_S_y(1,k-1)-(k/(k
+1)*P_derive_R_S_y(1,k-2)));
            P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
        end

 [detJacobian,inverse]=coordinate_transformation_Jacobian(sweep_angle,cr,b,gauss_p
        inverse_trans=inverse';
%         final_IJ_x=P_derive_I_J_x.*P_non_derive_I_J_x;
%         final_KL_x=P_derive_K_L_x.*P_non_derive_K_L_x;
        final_MN_x=P_derive_M_N_x.*P_non_derive_M_N_x;
%         final_PQ_x=P_derive_P_Q_x.*P_non_derive_P_Q_x;
```

```
%         final_RS_x=P_derive_R_S_x.*P_non_derive_R_S_x;
%         final_IJ_y=P_derive_I_J_y.*P_non_derive_I_J_y;
%         final_KL_y=P_derive_K_L_y.*P_non_derive_K_L_y;
        final_MN_y=P_derive_M_N_y.*P_non_derive_M_N_y;
%         final_PQ_y=P_derive_P_Q_y.*P_non_derive_P_Q_y;
%         final_RS_y=P_derive_R_S_y.*P_non_derive_R_S_y;
final_PQ=P_non_derive_P_Q_x.*P_non_derive_P_Q_y;
final_RS=P_non_derive_R_S_x.*P_non_derive_R_S_y;
        boom=5;
        idk=1;
%         final_IJ=[final_IJ_x;final_IJ_y];
%         final_KL=[final_KL_x;final_KL_y];
%         final_MN=[final_MN_x;final_MN_y];
%         final_PQ=[final_PQ_x;final_PQ_y];
%         final_RS=[final_RS_x;final_RS_y];
        for pi=1:length(final_MN_x)
            B(1:12,idk:boom)=B(1:12,idk:boom)+[0 0 0 0 0;0 0 0 0 0;0 0
 0 0 0;0 0 0 0 0;0 0 final_MN_x(1,pi) 0 0;0 0 final_MN_y(1,pi) 0 0;0
 0 0 0 0;0 0 0 0 0;0 0 0 0 0;0 0 0 0 0;0 0 0 final_PQ(1,pi) 0;0 0 0 0
 final_RS(1,pi)];
            idk=idk+5;
            boom=boom+5;
        end

 D=D_matrix_lower_airfoil_trans_plane(sweep_angle,cr,b,theta,thickness_of_each_ply
        hope=hope+B'*D*B*weights_x(j)*weights_y(op)*detJacobian;
    end
    end
%     iv_0=[5*n11_IJ-4 5*n11_IJ-3 5*n11_IJ-2 5*n11_IJ-1 5*n11_IJ
 5*n22_IJ-4 5*n22_IJ-3 5*n22_IJ-2 5*n22_IJ-1 5*n22_IJ 5*n33_IJ-4
 5*n33_IJ-3 5*n33_IJ-2 5*n33_IJ-1 5*n33_IJ 5*n44_IJ-4 5*n44_IJ-3
 5*n44_IJ-2 5*n44_IJ-1 5*n44_IJ];
%     iv_1=[5*n11_KL-4 5*n11_KL-3 5*n11_KL-2 5*n11_KL-1 5*n11_KL
 5*n22_KL-4 5*n22_KL-3 5*n22_KL-2 5*n22_KL-1 5*n22_KL 5*n33_KL-4
 5*n33_KL-3 5*n33_KL-2 5*n33_KL-1 5*n33_KL 5*n44_KL-4 5*n44_KL-3
 5*n44_KL-2 5*n44_KL-1 5*n44_KL];
%     iv_2=[5*n11_PQ-4 5*n11_PQ-3 5*n11_PQ-2 5*n11_PQ-1 5*n11_PQ
 5*n22_PQ-4 5*n22_PQ-3 5*n22_PQ-2 5*n22_PQ-1 5*n22_PQ 5*n33_PQ-4
 5*n33_PQ-3 5*n33_PQ-2 5*n33_PQ-1 5*n33_PQ 5*n44_PQ-4 5*n44_PQ-3
 5*n44_PQ-2 5*n44_PQ-1 5*n44_PQ];
%     iv_3=[5*n11_RS-4 5*n11_RS-3 5*n11_RS-2 5*n11_RS-1 5*n11_RS
 5*n22_RS-4 5*n22_RS-3 5*n22_RS-2 5*n22_RS-1 5*n22_RS 5*n33_RS-4
 5*n33_RS-3 5*n33_RS-2 5*n33_RS-1 5*n33_RS 5*n44_RS-4 5*n44_RS-3
 5*n44_RS-2 5*n44_RS-1 5*n44_RS];
%     iv=[iv_0 iv_1 iv_2 iv_3];
    %kg(iv,iv)=kg(iv,iv)+hope;
end
kg_trans_plane=hope;
end


Local minimum found.

Optimization completed because the size of the gradient is less than
```

```
the value of the optimality tolerance.


Local minimum found.

Optimization completed because the size of the gradient is less than
the value of the optimality tolerance.


Not enough input arguments.

Error in B_matrix_trans_plane_skin (line 29)
number_of_nodes_I_J=((I*J));
```

Published with MATLAB® R2020a

```
function
 D_matrix_1=D_matrix_lower_airfoil_in_plane(sweep,cr,b,theta,thickness_of_each_ply
%coeff_of_z_lower=Lower_fitting_a_curve();
%coeff_of_z_upper=Upper_fitting_a_curve();
% sweep=30;
%cr=10*10^-3;
% b=20*10^-3;
thickness_lower=0.01;
Z=zeros(5,12);
%gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
%gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
z_2=0;
z_derive_1=0;
z_derive_2=0;
z_derive_3=0;
n=10;
z_lower=[0.000000  0.000000;
  0.002000 -0.009300;
  0.005000 -0.016000;
  0.010000 -0.022100;
  0.020000 -0.029500;
  0.030000 -0.034400;
  0.040000 -0.038100;
  0.050000 -0.041200;
  0.070000 -0.046200;
  0.100000 -0.051700;
  0.120000 -0.054700;
  0.150000 -0.058500;
  0.170000 -0.060600;
  0.200000 -0.063300;
  0.220000 -0.064700;
  0.250000 -0.066600;
  0.280000 -0.068000;
  0.300000 -0.068700;
  0.320000 -0.069200;
  0.350000 -0.069600;
  0.370000 -0.069600;
  0.400000 -0.069200;
  0.420000 -0.068800;
  0.450000 -0.067600;
  0.480000 -0.065700;
  0.500000 -0.064400;
  0.530000 -0.061400;
  0.550000 -0.058800;
  0.580000 -0.054300;
  0.600000 -0.050900;
  0.630000 -0.045100;
  0.650000 -0.041000;
  0.680000 -0.034600;
  0.700000 -0.030200;
  0.730000 -0.023500;
```

```
  0.750000 -0.019200;
  0.770000 -0.015000;
  0.800000 -0.009300;
  0.830000 -0.004800;
  0.850000 -0.002400;
  0.870000 -0.001300;
  0.890000 -0.000800;
  0.920000 -0.001600;
  0.940000 -0.003500;
  0.950000 -0.004900;
  0.960000 -0.006600;
  0.970000 -0.008500;
  0.980000 -0.010900;
  0.990000 -0.013700;
  1.000000 -0.016300];
x=z_lower(1:end,1);
z_lower_1=z_lower(1:end,1);
for po=1:length(coeff_of_z_lower)

 z_derive_1=z_derive_1+(coeff_of_z_lower(po)*(((factorial(length(coeff_of_z_lower)
(factorial(length(coeff_of_z_lower)-
po)*factorial(po))))*po*(length(coeff_of_z_lower)-
po)*((z_lower_1(po))^(po-1)))*((z_lower_1(po))^0.75)*((1-
z_lower(po))^0.75));

 z_derive_2=z_derive_2+(coeff_of_z_lower(po))*(factorial(length(coeff_of_z_lower))
((factorial(length(coeff_of_z_lower)-
po)*factorial(po))))*(0.75)*(z_lower_1(po))^(0.75-1)*(1-
z_lower_1(po))^0.75;

 z_derive_3=z_derive_3+(coeff_of_z_lower(po)*(((factorial(length(coeff_of_z_lower)
(factorial(length(coeff_of_z_lower))*(0.75*(1-
z_lower_1(po))^(0.75-1)*(z_lower_1(po))^(0.75)))))));
end
z_derive=z_derive_1+z_derive_2+z_derive_3;
y=atan(z_derive);
y=(y*180/pi);
thickness_lower_1=thickness_lower*sqrt(1+(tan(y))^2);
%step_size=thickness_lower_1/n;
%for k=1:length(coeff_of_z_lower)
%theta=[0 45 45 0];
% thickness_of_each_ply=0.1*(ones(1,4));
z_1=(((x).^0.75).*(1-x).^(0.75));
for k=1:length(x)
z_2=z_2+(factorial(length(coeff_of_z_lower))/
(factorial(length(coeff_of_z_lower)-
k)*factorial(k)))*coeff_of_z_lower*(((x).^k).*(ones(1,length(x))-
x).^(length(coeff_of_z_lower)-k))';
end
z_1=z_1';
z=z_1+z_2;
z_upper=z+ones(1,length(x))*(thickness_lower_1/2);
z_l_lower=z-(ones(1,length(x))*(thickness_lower_1/2));
h=(z_upper-z_l_lower)./n;
```

```
integral_final=letsintegrate(z_upper,z_l_lower,sweep,cr,b,gauss_points_x,gauss_poi
for i=1:length(n)
    k=z_l_lower+i*h;
    for lo=1:length(z_l_lower)
    if rem(k(lo),3)==0
        integral_final=integral_final
+2*(function_1(k(lo),sweep,cr,b,gauss_points_x,gauss_points_y));
    else
        integral_final=integral_final
+3*(function_1(k(lo),sweep,cr,b,gauss_points_x,gauss_points_y));
    end
    end
    integral_final=integral_final*3*h(1)/8;
end
[stiffness]=abd_matrix(theta,thickness_of_each_ply);
D_matrix_1=integral_final'*stiffness*integral_final;
function func=function_1(k,sweep,cr,b,gauss_points_x,gauss_points_y)
% sweep=30;
% cr=10;
% b=20;
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
inverse_trans=inverse';
func=[(inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0
 0 0 k*(inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0
 0 0;0 0 0 (inverse_trans(1,2)+inverse_trans(2,1)) 0
 0 0 0 0 (k)*(inverse_trans(1,2)+inverse_trans(2,2))
 0 0;0 (inverse_trans(1,2)+inverse_trans(2,1))
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0
 0 k*(inverse_trans(2,2)+inverse_trans(2,1))
 k*(inverse_trans(1,1)+inverse_trans(1,2)) 0 0 0];
end
function
 integral=letsintegrate(z_pp,z_l_lower,sweep,cr,b,gauss_points_x,gauss_points_y)
% sweep=30;
% cr=10*10^-3;
% b=20*10^-3;
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
inverse_trans=inverse';
integration_1=zeros(3,12);
integration_2=zeros(3,12);
for i=1:length(z_pp)
integration_1=integration_1+[(inverse_trans(1,1)+inverse_trans(2,1))
 0 0 0 0 0 z_pp(i)*(inverse_trans(1,1)+inverse_trans(2,1))
 0 0 0 0 0;0 0 0 (inverse_trans(1,2)+inverse_trans(2,1)) 0
 0 0 0 0 (z_pp(i))*(inverse_trans(1,2)+inverse_trans(2,2))
 0 0;0 (inverse_trans(1,2)+inverse_trans(2,1))
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0 0
 z_pp(i)*(inverse_trans(2,2)+inverse_trans(2,1))
 z_pp(i)*(inverse_trans(1,1)+inverse_trans(1,2)) 0 0 0];
```

```
integration_2=integration_2+[(inverse_trans(1,1)+inverse_trans(2,1))
 0 0 0 0 0 z_l_lower(i)*(inverse_trans(1,1)+inverse_trans(2,1))
 0 0 0 0 0;0 0 0 (inverse_trans(1,2)+inverse_trans(2,1)) 0 0
 0 0 0 (z_l_lower(i))*(inverse_trans(1,2)+inverse_trans(2,2))
 0 0;0 (inverse_trans(1,2)+inverse_trans(2,1))
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0 0
 z_l_lower(i)*(inverse_trans(2,2)+inverse_trans(2,1))
 z_l_lower(i)*(inverse_trans(1,1)+inverse_trans(1,2)) 0 0 0];
end
integral=integration_1+integration_2;
end
end
```

```
Not enough input arguments.

Error in D_matrix_lower_airfoil_in_plane (line 11)
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
```

Published with MATLAB® R2020a

```
function
 D_matrix_trans_1=D_matrix_lower_airfoil_trans_plane(sweep,cr,b,theta,thickness_of
% coeff_of_z_lower=Lower_fitting_a_curve();
% coeff_of_z_upper=Upper_fitting_a_curve();
% sweep=30;
% cr=10*10^-3;
% b=20*10^-3;
thickness_lower=0.01;
Z=zeros(5,12);
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
z_2=0;
z_derive_1=0;
z_derive_2=0;
z_derive_3=0;
n=10;
z_lower=[0.000000  0.000000;
  0.002000 -0.009300;
  0.005000 -0.016000;
  0.010000 -0.022100;
  0.020000 -0.029500;
  0.030000 -0.034400;
  0.040000 -0.038100;
  0.050000 -0.041200;
  0.070000 -0.046200;
  0.100000 -0.051700;
  0.120000 -0.054700;
  0.150000 -0.058500;
  0.170000 -0.060600;
  0.200000 -0.063300;
  0.220000 -0.064700;
  0.250000 -0.066600;
  0.280000 -0.068000;
  0.300000 -0.068700;
  0.320000 -0.069200;
  0.350000 -0.069600;
  0.370000 -0.069600;
  0.400000 -0.069200;
  0.420000 -0.068800;
  0.450000 -0.067600;
  0.480000 -0.065700;
  0.500000 -0.064400;
  0.530000 -0.061400;
  0.550000 -0.058800;
  0.580000 -0.054300;
  0.600000 -0.050900;
  0.630000 -0.045100;
  0.650000 -0.041000;
  0.680000 -0.034600;
  0.700000 -0.030200;
  0.730000 -0.023500;
```

```
  0.750000 -0.019200;
  0.770000 -0.015000;
  0.800000 -0.009300;
  0.830000 -0.004800;
  0.850000 -0.002400;
  0.870000 -0.001300;
  0.890000 -0.000800;
  0.920000 -0.001600;
  0.940000 -0.003500;
  0.950000 -0.004900;
  0.960000 -0.006600;
  0.970000 -0.008500;
  0.980000 -0.010900;
  0.990000 -0.013700;
  1.000000 -0.016300];
x=z_lower(1:end,1);
z_lower_1=z_lower(1:end,1);
for po=1:length(coeff_of_z_lower)

 z_derive_1=z_derive_1+(coeff_of_z_lower(po)*(((factorial(length(coeff_of_z_lower)
(factorial(length(coeff_of_z_lower)-
po)*factorial(po))))*po*(length(coeff_of_z_lower)-
po)*((z_lower_1(po))^(po-1)))*((z_lower_1(po))^0.75)*((1-
z_lower(po))^0.75));

 z_derive_2=z_derive_2+(coeff_of_z_lower(po))*(factorial(length(coeff_of_z_lower))
((factorial(length(coeff_of_z_lower)-
po)*factorial(po))))*(0.75)*(z_lower_1(po))^(0.75-1)*(1-
z_lower_1(po))^0.75;

 z_derive_3=z_derive_3+(coeff_of_z_lower(po)*(((factorial(length(coeff_of_z_lower)
(factorial(length(coeff_of_z_lower))*(0.75*(1-
z_lower_1(po))^(0.75-1)*(z_lower_1(po))^(0.75)))))));
end
z_derive=z_derive_1+z_derive_2+z_derive_3;
y=atan(z_derive);
y=(y*180/pi);
thickness_lower_1=thickness_lower*sqrt(1+(tan(y))^2);
%step_size=thickness_lower_1/n;
%for k=1:length(coeff_of_z_lower)
% theta=[0 90 90 0];
% thickness_of_each_ply=0.1*(ones(1,4));
z_1=(((x).^0.75).*(1-x).^(0.75));
for k=1:length(x)
z_2=z_2+(factorial(length(coeff_of_z_lower))/
(factorial(length(coeff_of_z_lower)-
k)*factorial(k)))*coeff_of_z_lower*(((x).^k).*(ones(1,length(x))-
x).^(length(coeff_of_z_lower)-k))';
end
z_1=z_1';
z=z_1+z_2;
z_upper=z+ones(1,length(x))*(thickness_lower_1/2);
z_l_lower=z-(ones(1,length(x))*(thickness_lower_1/2));
h=(z_upper-z_l_lower)./n;
```

```
integral_final=letsintegrate(z_upper,z_l_lower,sweep,cr,b,gauss_points_x,gauss_poi
for i=1:length(n)
    k=z_l_lower+i*h;
    for lo=1:length(z_l_lower)
    if rem(k(lo),3)==0
        integral_final=integral_final
+2*(function_1(k(lo),sweep,cr,b,gauss_points_x,gauss_points_y));
    else
        integral_final=integral_final
+3*(function_1(k(lo),sweep,cr,b,gauss_points_x,gauss_points_y));
    end
    end
    integral_final=integral_final*3*h(1)/8;
end
[stiffness]=abd_matrix_trans(theta,thickness_of_each_ply);
D_matrix_trans_1=integral_final'*stiffness*integral_final;
function func=function_1(k,sweep,cr,b,gauss_points_x,gauss_points_y)
% sweep=30;
% cr=10;
% b=20;
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
inverse_trans=inverse';
func=[0 0 0 0 0 (inverse_trans(1,2)+inverse_trans(2,2)) 0 0 0 0 0 1;0
 0 0 0 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0 0 0 1 0];
end
function
 integral=letsintegrate(z_pp,z_l_lower,sweep,cr,b,gauss_points_x,gauss_points_y)
% sweep=30;
% cr=10*10^-3;
% b=20*10^-3;
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
inverse_trans=inverse';
integration_1=zeros(2,12);
integration_2=zeros(2,12);
for i=1:length(z_pp)
integration_1=integration_1+[0 0 0 0 0
 (inverse_trans(1,2)+inverse_trans(2,2)) 0 0 0 0 0 1;0 0 0 0
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0 0 0 1 0];
integration_2=integration_2+[0 0 0 0 0
 (inverse_trans(1,2)+inverse_trans(2,2)) 0 0 0 0 0 1;0 0 0 0
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0 0 0 1 0];
end
integral=integration_1+integration_2;
end
end
```

```
Not enough input arguments.

Error in D_matrix_lower_airfoil_trans_plane (line 11)
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
```

```
function
 D_matrix_2=D_matrix_upper_airfoil_trans_plane(sweep,cr,b,theta,thickness_of_each_
%coeff_of_z_lower=Lower_fitting_a_curve();
% coeff_of_z_upper=Upper_fitting_a_curve();
% sweep=30;
% cr=10*10^-3;
% b=20*10^-3;
thickness_lower=0.01;
Z=zeros(5,12);
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
z_2=0;
z_derive_1=0;
z_derive_2=0;
z_derive_3=0;
n=10;
z_uppe_1=[0.000000  0.000000;
  0.002000  0.009500;
  0.005000  0.015800;
  0.010000  0.021900;
  0.020000  0.029300;
  0.030000  0.034300;
  0.040000  0.038100;
  0.050000  0.041100;
  0.070000  0.046200;
  0.100000  0.051800;
  0.120000  0.054800;
  0.150000  0.058500;
  0.170000  0.060600;
  0.200000  0.063200;
  0.220000  0.064600;
  0.250000  0.066400;
  0.270000  0.067300;
  0.300000  0.068500;
  0.330000  0.069200;
  0.350000  0.069600;
  0.380000  0.069800;
  0.400000  0.069700;
  0.430000  0.069500;
  0.450000  0.069200;
  0.480000  0.068400;
  0.500000  0.067800;
  0.530000  0.066600;
  0.550000  0.065600;
  0.570000  0.064500;
  0.600000  0.062500;
  0.620000  0.061000;
  0.650000  0.058500;
  0.680000  0.055500;
  0.700000  0.053300;
  0.720000  0.050900;
```

```
  0.750000  0.046900;
  0.770000  0.043900;
  0.800000  0.038900;
  0.820000  0.035300;
  0.850000  0.029400;
  0.870000  0.025100;
  0.900000  0.018100;
  0.920000  0.013100;
  0.950000  0.004900;
  0.970000 -0.000900;
  0.980000 -0.003900;
  0.990000 -0.007100;
  1.000000 -0.010400;
  ];
x=z_uppe_1(1:end,1);
z_lower_actual=z_uppe_1(1:end,1);
for po=1:length(coeff_of_z_upper)

 z_derive_1=z_derive_1+(coeff_of_z_upper(po)*(((factorial(length(coeff_of_z_upper)
(factorial(length(coeff_of_z_upper)-
po)*factorial(po))))*po*(length(coeff_of_z_upper)-
po)*((z_lower_actual(po))^(po-1)))*((z_lower_actual(po))^0.75)*((1-
z_lower_actual(po))^0.75));

 z_derive_2=z_derive_2+(coeff_of_z_upper(po))*(factorial(length(coeff_of_z_upper))
((factorial(length(coeff_of_z_upper)-
po)*factorial(po))))*(0.75)*(z_lower_actual(po))^(0.75-1)*(1-
z_lower_actual(po))^0.75;

 z_derive_3=z_derive_3+(coeff_of_z_upper(po)*(((factorial(length(coeff_of_z_upper)
(factorial(length(coeff_of_z_upper))*(0.75*(1-
z_lower_actual(po))^(0.75-1)*(z_lower_actual(po))^(0.75)))))));
end
z_derive=z_derive_1+z_derive_2+z_derive_3;
y=atan(z_derive);
y=(y*180/pi);
thickness_lower_1=thickness_lower*sqrt(1+(tan(y))^2);
%step_size=thickness_lower_1/n;
%for k=1:length(coeff_of_z_lower)

% theta=[0 90 90 0];
% thickness_of_each_ply=0.1*(ones(1,4));
z_1=(((x).^0.75).*(1-x).^(0.75));
for k=1:length(x)
z_2=z_2+(factorial(length(coeff_of_z_upper))/
(factorial(length(coeff_of_z_upper)-
k)*factorial(k)))*coeff_of_z_upper*(((x).^k).*(ones(1,length(x))-
x).^(length(coeff_of_z_upper)-k))';
end
z_1=z_1';
z=z_1+z_2;
z_upper=z+ones(1,length(x))*(thickness_lower_1/2);
z_l_lower=z-(ones(1,length(x))*(thickness_lower_1/2));
h=(z_upper-z_l_lower)./n;
```

```
integral_final=letsintegrate(z_upper,z_l_lower,sweep,cr,b,gauss_points_x,gauss_poi
for i=1:length(n)
    k=z_l_lower+i*h;
    for lo=1:length(z_l_lower)
    if rem(k(lo),3)==0
        integral_final=integral_final
+2*(function_1(k(lo),sweep,cr,b,gauss_points_x,gauss_points_y));
    else
        integral_final=integral_final
+3*(function_1(k(lo),sweep,cr,b,gauss_points_x,gauss_points_y));
    end
    end
    integral_final=integral_final*3*h(1)/8;
end
[stiffness]=abd_matrix_trans(theta,thickness_of_each_ply);
D_matrix_2=integral_final'*stiffness*integral_final;
function func=function_1(k,sweep,cr,b,gauss_points_x,gauss_points_y)
% sweep=30;
% cr=10;
% b=20;
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
inverse_trans=inverse';
func=[0 0 0 0 0 (inverse_trans(1,2)+inverse_trans(2,2)) 0 0 0 0 0 1;0
 0 0 0 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0 0 0 1 0];
end
function
 integral=letsintegrate(z_pp,z_l_lower,sweep,cr,b,gauss_points_x,gauss_points_y)
% sweep=30;
% cr=10*10^-3;
% b=20*10^-3;
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
inverse_trans=inverse';
integration_1=zeros(2,12);
integration_2=zeros(2,12);
for i=1:length(z_pp)
integration_1=integration_1+[0 0 0 0 0
 (inverse_trans(1,2)+inverse_trans(2,2)) 0 0 0 0 0 1;0 0 0 0
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0 0 0 1 0];
integration_2=integration_2+[0 0 0 0 0
 (inverse_trans(1,2)+inverse_trans(2,2)) 0 0 0 0 0 1;0 0 0 0
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0 0 0 1 0];
end
integral=integration_1+integration_2;
end
end
```

```
Not enough input arguments.

Error in D_matrix_upper_airfoil_trans_plane (line 11)
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
```

```
function
 skin_Stiffness=FEM_Main_file_skin(gauss_points_in_x,gauss_points_in_y,gauss_point
% gauss_points_in_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_trans_x=0;
% gauss_points_in_y=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_trans_y=0;
% weights_in_x=[1 1];
% weights_in_y=[1 1];
% weights_trans_x=2;
% weights_trans_y=2;
% sweep=30;%degree
% b=10;
% cr=5;
% I=3;
% J=3;
% K=3;
% L=3;
% M=3;
% N=3;
% P=3;
% Q=3;
% R=3;
% S=3;
B_in_plane=B_matrix_in_plane_skin(gauss_points_in_x,gauss_points_in_y,sweep,b,cr,I
B_trans_plane=B_matrix_trans_plane_skin(gauss_points_trans_x,gauss_points_trans_y,
skin_Stiffness=B_in_plane+B_trans_plane;
end
```

```
Not enough input arguments.

Error in FEM_Main_file_skin (line 23)
B_in_plane=B_matrix_in_plane_skin(gauss_points_in_x,gauss_points_in_y,sweep,b,cr,I
```

Published with MATLAB® R2020a

```
function curve_fit_Lower=Lower_fitting_a_curve()
z_lower=[0.000000  0.000000;
  0.002000 -0.009300;
  0.005000 -0.016000;
  0.010000 -0.022100;
  0.020000 -0.029500;
  0.030000 -0.034400;
  0.040000 -0.038100;
  0.050000 -0.041200;
  0.070000 -0.046200;
  0.100000 -0.051700;
  0.120000 -0.054700;
  0.150000 -0.058500;
  0.170000 -0.060600;
  0.200000 -0.063300;
  0.220000 -0.064700;
  0.250000 -0.066600;
  0.280000 -0.068000;
  0.300000 -0.068700;
  0.320000 -0.069200;
  0.350000 -0.069600;
  0.370000 -0.069600;
  0.400000 -0.069200;
  0.420000 -0.068800;
  0.450000 -0.067600;
  0.480000 -0.065700;
  0.500000 -0.064400;
  0.530000 -0.061400;
  0.550000 -0.058800;
  0.580000 -0.054300;
  0.600000 -0.050900;
  0.630000 -0.045100;
  0.650000 -0.041000;
  0.680000 -0.034600;
  0.700000 -0.030200;
  0.730000 -0.023500;
  0.750000 -0.019200;
  0.770000 -0.015000;
  0.800000 -0.009300;
  0.830000 -0.004800;
  0.850000 -0.002400;
  0.870000 -0.001300;
  0.890000 -0.000800;
  0.920000 -0.001600;
  0.940000 -0.003500;
  0.950000 -0.004900;
  0.960000 -0.006600;
  0.970000 -0.008500;
  0.980000 -0.010900;
  0.990000 -0.013700;
  1.000000 -0.016300];
x_guess=ones(1,length(z_lower));
```

```
airfoil_equation_lower_curve_2=@airfoil_equation_lower_curve;
curve_fit_Lower=lsqcurvefit(airfoil_equation_lower_curve_2,x_guess,z_lower(1:end,1
z_predict=z_lower(1:end,1)';
thickness_ratio_lower_surface=0.01;
S_predict=zeros(length(z_lower),length(z_lower));
for i=1:length(z_predict)
   S_predict(i,1:length(z_predict))=(factorial(length(z_lower))/
(factorial(length(z_lower)-
i)*factorial(i)))*(((z_predict).^i).*((ones(1,length(z_predict))-
z_predict).^(length(z_predict)-i)));
end
% predicited_graph=((z_predict).^0.75).*((ones(1,length(z_predict))-
z_predict).^0.75).*(curve_fit_Lower*S_predict)+ones(1,length(z_predict))*thickness
% plot(z_lower(1:end,1)',predicited_graph,"b")
% hold on
% plot(z_lower(1:end,1),z_lower(1:end,2),"bo")
% hold off
function z=airfoil_equation_lower_curve(x,x_data)
x_data=x_data';
thickness_ratio_lower_surface=0.01;
S=zeros(length(x_data),length(x_data));
for i=1:length(x_data)
   S(i,1:length(x_data))=(factorial(length(x_data))/
(factorial(length(x_data)-
i)*factorial(i)))*(((x_data).^i).*((ones(1,length(x_data))-
x_data).^(length(x_data)-i)));
end
matrix=zeros(1,length(x_data));
for i=1:length(x_data)
    matrix(1,i)=x(i);
end
z=((x_data).^0.75).*((ones(1,length(x_data))-
x_data).^0.75).*(matrix*S)+ones(1,length(x_data))*thickness_ratio_lower_surface;
end
end
```

```
Local minimum found.

Optimization completed because the size of the gradient is less than
the value of the optimality tolerance.


ans =

  Columns 1 through 7

   -6.8476   16.4317  -26.3682   19.0586    1.1947  -12.2233   -0.0915

  Columns 8 through 14

    7.5818    1.1945   -5.4205   -3.4790    2.0104    3.4490    0.2375

  Columns 15 through 21
```

```
  -2.8556   -2.4605    0.2447    1.9473    1.0842   -0.9910   -1.9690

Columns 22 through 28

  -1.0438    0.5741    1.1811    0.3165   -0.9685   -1.3421   -0.5335

Columns 29 through 35

   0.5210    0.7394   -0.0024   -0.8314   -0.8664   -0.1462    0.4839

Columns 36 through 42

   0.3620   -0.2620   -0.5966   -0.3027    0.1513    0.2203    0.0051

Columns 43 through 49

  -0.1396   -0.2629   -0.2347    0.4094    0.2315   -1.3475    1.2399

Column 50

  -1.7220
```

Published with MATLAB® R2020a

```
function C=ques_1_compliance()
% prompt="Enter the value of E";
% prompt_1="Enter the value of poisson's ratio";
% prompt_2="Enter the value of G";
E1=70.7;
E2=70.7;
%E3=input(prompt);
poisson_ratio_12=0.3;
%poisson_ratio_13=input(prompt_1);
%poisson_ratio_23=input(prompt_1);
G12=27.19;
%G13=input(prompt_2);
%G23=input(prompt_2);
S=zeros(3,3);
S(1,1)=1/E1;
S(1,2)=-poisson_ratio_12/E2;
%S(1,3)=poisson_ratio_13/E3;
S(2,1)=-poisson_ratio_12/E1;
S(2,2)=1/E2;
%S(2,3)=poisson_ratio_23/E3;
%S(3,1)=poisson_ratio_13/E1;
%S(3,2)=poisson_ratio_23/E2;
%S(3,3)=1/E3;
%S(4,4)=1/G23;
%S(5,5)=1/G13;
S(3,3)=1/G12;
C=inv(S);
end
```

```
ans =

   77.6923   23.3077         0
   23.3077   77.6923         0
         0         0   27.1900
```

Published with MATLAB® R2020a

```
function C=ques_1_compliance_trans_plane()
% prompt="Enter the value of E";
% prompt_1="Enter the value of poisson's ratio";
% prompt_2="Enter the value of G";
%E1=15;
%E2=20;
%E3=input(prompt);
%poisson_ratio_12=0.3;
%poisson_ratio_13=input(prompt_1);
%poisson_ratio_23=input(prompt_1);
%G12=5;
G13=27.19;
G23=27.19;
S=zeros(2,2);
S(1,1)=1/G13;
S(1,2)=0;
%S(1,3)=poisson_ratio_13/E3;
S(2,1)=0;
S(2,2)=1/G23;
%S(2,3)=poisson_ratio_23/E3;
%S(3,1)=poisson_ratio_13/E1;
%S(3,2)=poisson_ratio_23/E2;
%S(3,3)=1/E3;
%S(4,4)=1/G23;
%S(5,5)=1/G13;
%S(3,3)=1/G12;
C=inv(S);
end
```

```
ans =

   27.1900         0
         0   27.1900
```

Published with MATLAB® R2020a

```
function Q=ques_2_Q_matrix_in_plane()
Stiffness=ques_1_compliance_in_plane();
Q=zeros(3,3);
Q(1,1)=Stiffness(1,1);%-(Stiffness(1,3)^2/Stiffness(3,3));
Q(1,2)=Stiffness(1,2);%-((Stiffness(1,3)*Stiffness(2,3))/
Stiffness(3,3));
Q(2,2)=Stiffness(2,2);%-(Stiffness(2,3)^2/Stiffness(3,3));
Q(3,3)=Stiffness(3,3);
Q(2,1)=Q(1,2);
end
```

```
ans =

   77.6923   23.3077         0
   23.3077   77.6923         0
         0         0   27.1900
```

Published with MATLAB® R2020a

```
function C=ques_3_with_arguments_transformed(theta)
%prompt="Enter the value of theta in degree";
%theta=input(prompt);
theta=(pi*theta)/180;
C=zeros(3,3);
C(1,1)=cos(theta)^2;
C(1,2)=sin(theta)^2;
C(1,3)=2*cos(theta)*sin(theta);
C(2,1)=sin(theta)^2;
C(2,2)=cos(theta)^2;
C(2,3)=-2*cos(theta)*sin(theta);
C(3,1)=-cos(theta)*sin(theta);
C(3,2)=cos(theta)*sin(theta);
C(3,3)=cos(theta)^2-sin(theta)^2;
end
```

```
Not enough input arguments.

Error in ques_3_with_arguments_transformed_in_plane (line 4)
theta=(pi*theta)/180;
```

Published with MATLAB® R2020a

```
function C=ques_3_with_arguments_transformed_trans_plane(theta)
%prompt="Enter the value of theta in degree";
%theta=input(prompt);
%theta=30;
theta=(pi*theta)/180;
C=zeros(2,2);
C(1,1)=cos(theta);
C(1,2)=-sin(theta);
%C(1,3)=2*cos(theta)*sin(theta);
C(2,1)=-sin(theta);
C(2,2)=cos(theta);
% C(2,3)=-2*cos(theta)*sin(theta);
% C(3,1)=-cos(theta)*sin(theta);
% C(3,2)=cos(theta)*sin(theta);
% C(3,3)=cos(theta)^2-sin(theta)^2;
end
```

```
Not enough input arguments.

Error in ques_3_with_arguments_transformed_trans_plane (line 5)
theta=(pi*theta)/180;
```

Published with MATLAB® R2020a

```
function curve_fit_upper=Upper_fitting_a_curve()
z_upper=[0.000000  0.000000;
  0.002000  0.009500;
  0.005000  0.015800;
  0.010000  0.021900;
  0.020000  0.029300;
  0.030000  0.034300;
  0.040000  0.038100;
  0.050000  0.041100;
  0.070000  0.046200;
  0.100000  0.051800;
  0.120000  0.054800;
  0.150000  0.058500;
  0.170000  0.060600;
  0.200000  0.063200;
  0.220000  0.064600;
  0.250000  0.066400;
  0.270000  0.067300;
  0.300000  0.068500;
  0.330000  0.069200;
  0.350000  0.069600;
  0.380000  0.069800;
  0.400000  0.069700;
  0.430000  0.069500;
  0.450000  0.069200;
  0.480000  0.068400;
  0.500000  0.067800;
  0.530000  0.066600;
  0.550000  0.065600;
  0.570000  0.064500;
  0.600000  0.062500;
  0.620000  0.061000;
  0.650000  0.058500;
  0.680000  0.055500;
  0.700000  0.053300;
  0.720000  0.050900;
  0.750000  0.046900;
  0.770000  0.043900;
  0.800000  0.038900;
  0.820000  0.035300;
  0.850000  0.029400;
  0.870000  0.025100;
  0.900000  0.018100;
  0.920000  0.013100;
  0.950000  0.004900;
  0.970000 -0.000900;
  0.980000 -0.003900;
  0.990000 -0.007100;
  1.000000 -0.010400;
  ];
x_guess=ones(1,length(z_upper));
airfoil_equation_lower_curve_2=@airfoil_equation_lower_curve;
```

```
curve_fit_upper=lsqcurvefit(airfoil_equation_lower_curve_2,x_guess,z_upper(1:end,1
z_predict=z_upper(1:end,1)';
thickness_ratio_lower_surface=0.01;
S_predict=zeros(length(z_upper),length(z_upper));
for i=1:length(z_predict)
   S_predict(i,1:length(z_predict))=(factorial(length(z_upper))/
(factorial(length(z_upper)-
i)*factorial(i)))*(((z_predict).^i).*((ones(1,length(z_predict))-
z_predict).^(length(z_predict)-i)));
end
% predicited_graph=((z_predict).^0.75).*((ones(1,length(z_predict))-
z_predict).^0.75).*(curve_fit_upper*S_predict)+ones(1,length(z_predict))*thickness
% plot(z_upper(1:end,1)',predicited_graph,"b")
% hold on
% plot(z_upper(1:end,1),z_upper(1:end,2),"bo")
% hold off
function z=airfoil_equation_lower_curve(x,x_data)
x_data=x_data';
thickness_ratio_lower_surface=0.01;
S=zeros(length(x_data),length(x_data));
for i=1:length(x_data)
   S(i,1:length(x_data))=(factorial(length(x_data))/
(factorial(length(x_data)-
i)*factorial(i)))*(((x_data).^i).*((ones(1,length(x_data))-
x_data).^(length(x_data)-i)));
end
matrix=zeros(1,length(x_data));
for i=1:length(x_data)
    matrix(1,i)=x(i);
end
z=((x_data).^0.75).*((ones(1,length(x_data))-
x_data).^0.75).*(matrix*S)+ones(1,length(x_data))*thickness_ratio_lower_surface;
end
end
```

Local minimum found.

Optimization completed because the size of the gradient is less than the value of the optimality tolerance.

ans =

Columns 1 through 7

1.4012 -1.1876 1.2572 0.4493 -0.2935 0.0879 0.5228

Columns 8 through 14

0.4273 0.1085 -0.0065 0.1345 0.3086 0.3318 0.2166

Columns 15 through 21

```
    0.0980    0.0795    0.1532    0.2356    0.2548    0.2051    0.1366

  Columns 22 through 28

    0.1040    0.1243    0.1720    0.2059    0.2026    0.1689    0.1313

  Columns 29 through 35

    0.1141    0.1246    0.1522    0.1777    0.1829    0.1583    0.1120

  Columns 36 through 42

    0.0756    0.0893    0.1579    0.2111    0.1484   -0.0132   -0.0575

  Columns 43 through 48

    0.1703    0.3062   -0.1698   -0.3775    0.6231   -1.1358
```

Published with MATLAB® R2020a

5.4 Rib web

```
function stiffness=abd_matrix(theta_of_each_ply,~)
%theta_of_each_ply=[0 45 45 0];
%thickness_of_each_ply=ones(1,length(theta_of_each_ply));
% height_of_lamina=sum(thickness_of_each_ply);
% ords1=zeros(1,(length(thickness_of_each_ply)+1));
% ords1(1,1)=-height_of_lamina/2;
% for i=2:(length(thickness_of_each_ply))
%     ords1(1,i)=ords1(1,i-1)+thickness_of_each_ply(1,i);
% end
% ords1(1,end)=abs(ords1(1,1));
Q_transformed_1=zeros(3,3);
for i=1:length(theta_of_each_ply)

 Q_transformed=inv(ques_3_with_arguments_transformed_in_plane(theta_of_each_ply(i)
%     abd(1:3,1:3)=abd(1:3,1:3)+Q_transformed*((ords1(1,i+1))-
ords1(1,i));
%     abd(4:end,1:3)=abd(4:end,1:3)+0.5*Q_transformed*((ords1(1,i
+1))^2-ords1(1,i)^2);
%     abd(1:3,4:6)=abd(1:3,4:6)+0.5*Q_transformed*((ords1(1,i+1))^2-
ords1(1,i)^2);
%     abd(4:6,4:6)=abd(4:6,4:6)+(1/3)*Q_transformed*((ords1(1,i+1))^3-
ords1(1,i)^3);
Q_transformed_1=Q_transformed_1+Q_transformed;
end
% z_k=0;
% z_k_2=0;
% for i=1:length(theta_of_each_ply)
% z_k=z_k+(ords1(1,i+1)-ords1(1,i));
% z_k_2=z_k_2+0.5*((ords1(1,i+1))^2-(ords1(1,i))^2);
% end
% complaince=[1 0 0 1 0 0;0 1 0 0 1 0;0 0 1 0 0 1]*inv(abd)*[z_k
 0 0;0 z_k 0;0 0 z_k;(((z_k_2)^2)/2) 0 0;0 (((z_k_2)^2)/2) 0;0 0
 (((z_k_2)^2)/2)];
stiffness=inv(Q_transformed_1);
end
```

```
Not enough input arguments.

Error in abd_matrix (line 12)
for i=1:length(theta_of_each_ply)
```

Published with MATLAB® R2020a

```
function
 kg_in_plane=B_matrix_in_plane(gauss_points_x,gauss_points_y,sweep_angle,b,cr,I,J,
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
% weights_x=[1 1];
% weights_y=[1 1];
% sweep_angle=30;%degree
% b=10;
% l_1=1;
% c=1;
% h=1;
% h_1=1;
pi=3.1416;
sweep_angle_in_radians=sweep_angle*pi/180;
%cr=5;
tip_chord=cr-((b/2)*tan(sweep_angle_in_radians));
%Linear nodes per element
% I=2;
% J=2;
% K=2;
% L=2;
% M=2;
% N=2;
% P=2;
% Q=2;
% R=2;
% S=2;
% dof=(5);
% icon_IJ=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_KL=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_PQ=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_RS=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
coeff_of_z_lower=Lower_fitting_a_curve();
coeff_of_z_upper=Upper_fitting_a_curve();
number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
%number_of_elements=number_of_nodes-1;
P_non_derive_I_J_x=zeros(1,I*J);
P_derive_I_J_x=zeros(1,I*J);
P_derive_K_L_x=zeros(1,K*L);
P_non_derive_K_L_x=zeros(1,K*L);
P_derive_M_N_x=zeros(1,M*N);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_derive_I_J_y=zeros(1,I*J);
```

```
P_derive_K_L_y=zeros(1,K*L);
P_non_derive_K_L_y=zeros(1,K*L);
P_derive_M_N_y=zeros(1,M*N);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_derive_R_S_y=zeros(1,R*S);
B=zeros(12,(I*J+P*Q+M*N+K*L+R*S));
% theta=[0 90 90 0];
% thickness_of_each_ply=0.01*ones(1,length(theta));
hope=zeros((I*J+P*Q+M*N+K*L+R*S),(I*J+P*Q+M*N+K*L+R*S));
spar_position_I_J=linspace(-1,1,I*J);
spar_position_K_L=linspace(-1,1,P*Q);
spar_position_P_Q=linspace(-1,1,M*N);
spar_position_R_S=linspace(-1,1,R*S);
%kg=zeros(((I*J)+(K*L)+(P*Q)+(R*S))*dof,
((I*J)+(K*L)+(P*Q)+(R*S))*dof);
for
 i=1:I*J%*length(icon_KL(:,1))*length(icon_KL(1,:))*length(icon_PQ(:,1))*length(ic
%     n11_IJ=icon_IJ(i,1);
%     n22_IJ=icon_IJ(i,2);
%     n33_IJ=icon_IJ(i,3);
%     n44_IJ=icon_IJ(i,4);
%     %n55_IJ=icon_IJ(i,5);
%     n11_KL=icon_KL(i,1);
%     n22_KL=icon_KL(i,2);
%     n33_KL=icon_KL(i,3);
%     n44_KL=icon_KL(i,4);
%     %n55_KL=icon_KL(i,5);
%     n11_PQ=icon_PQ(i,1);
%     n22_PQ=icon_PQ(i,2);
%     n33_PQ=icon_PQ(i,3);
%     n44_PQ=icon_PQ(i,4);
%     %n55_PQ=icon_PQ(i,5);
%     n11_RS=icon_RS(i,1);
%     n22_RS=icon_RS(i,2);
%     n33_RS=icon_RS(i,3);
%     n44_RS=icon_RS(i,4);
    %n55_RS=icon_RS(i,5);
    for op=1:length(gauss_points_y)
        c=0.5*cr*(1-
gauss_points_y(op))+0.5*tip_chord*(1+gauss_points_y(op));
    for j=1:length(gauss_points_x)
P_non_derive_I_J_x(1,1)=1;
P_non_derive_I_J_x(1,2)=((t_1/
c)*gauss_points_x(j)+spar_position_I_J(1,1));
P_derive_I_J_x(1,1)=0;
P_derive_I_J_x(1,2)=(t_1/c);
P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=((t_1/
c)*gauss_points_x(j)+spar_position_K_L(1,1));
P_derive_M_N_x(1,1)=0;
P_derive_M_N_x(1,2)=(t_1/c);
```

```
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_derive_M_N_x(1,1)=0;
P_derive_M_N_x(1,2)=(t_1/c);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=((t_1/
c)*gauss_points_x(j)+spar_position_P_Q(1,1));
P_derive_P_Q_x(1,1)=0;
P_derive_P_Q_x(1,2)=(t_1/c);
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=((t_1/
c)*gauss_points_x(j)+spar_position_R_S(1,1));
P_derive_R_S_x(1,1)=0;
P_derive_R_S_x(1,2)=(t_1/c);
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=((t_1/
c)*gauss_points_y(op)+spar_position_I_J(1,1));
P_derive_I_J_y(1,1)=0;
P_derive_I_J_y(1,2)=(t_1/c);
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=((t_1/
c)*gauss_points_y(op)+spar_position_K_L(1,1));
P_derive_M_N_y(1,1)=0;
P_derive_M_N_y(1,2)=(t_1/c);
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=((t_1/
c)*gauss_points_y(op)+spar_position_K_L(1,1));
P_derive_M_N_y(1,1)=0;
P_derive_M_N_y(1,2)=(t_1/c);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=((t_1/
c)*gauss_points_y(op)+spar_position_P_Q(1,1));
P_derive_P_Q_y(1,1)=0;
P_derive_P_Q_y(1,2)=(t_1/c);
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=((t_1/
c)*gauss_points_y(op)+spar_position_R_S(1,1));
P_derive_R_S_y(1,1)=0;
P_derive_R_S_y(1,2)=(t_1/c);
        for k=3:number_of_nodes_I_J
           P_derive_I_J_x(1,k)=((((2*k+1)/(k+1))*((t_1/
c)*gauss_points_x(j)+spar_position_I_J(1,k-1)))*P_derive_I_J_x(1,k-1)-
(k/(k+1)*P_derive_I_J_x(1,k-2)));
           P_non_derive_I_J_x(1,k)=((((2*k+1)/(k+1))*((t_1/
c)*gauss_points_x(j)+spar_position_I_J(1,k-1)))*P_non_derive_I_J_x(1,k-1)-
(k/(k+1)*P_non_derive_I_J_x(1,k-2)));
        end
        for k=3:number_of_nodes_K_L
           P_derive_K_L_x(1,k)=((((2*k+1)/(k+1))*((t_1/
c)*gauss_points_x(j)+spar_position_K_L(1,k-1)))*P_derive_K_L_x(1,k-1)-
(k/(k+1)*P_derive_K_L_x(1,k-2)));
           P_non_derive_K_L_x(1,k)=((((2*k+1)/(k+1))*((t_1/
c)*gauss_points_x(j)+spar_position_K_L(1,k-1)))*P_non_derive_K_L_x(1,k-1)-
(k/(k+1)*P_non_derive_K_L_x(1,k-2)));
```

```
        end
%         for k=3:number_of_nodes_M_N
%            P_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_M_N_x(1,k-1)-(k/(k
+1)*P_derive_M_N_x(1,k-2)));
%            P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
%         end
        for k=3:number_of_nodes_P_Q
           P_derive_P_Q_x(1,k)=((((2*k+1)/(k+1))*((t_1/
c)*gauss_points_x(j)+spar_position_P_Q(1,k-1)))*P_derive_P_Q_x(1,k-1)-
(k/(k+1)*P_derive_P_Q_x(1,k-2)));
           P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k+1))*((t_1/
c)*gauss_points_x(j)+spar_position_P_Q(1,k-1)))*P_non_derive_P_Q_x(1,k-1)-
(k/(k+1)*P_non_derive_P_Q_x(1,k-2)));
        end
        for k=3:number_of_nodes_R_S
           P_derive_R_S_x(1,k)=((((2*k+1)/(k+1))*(t_1/
c)*(gauss_points_x(j)+spar_position_R_S(1,k-1)))*P_derive_R_S_x(1,k-1)-
(k/(k+1)*P_derive_R_S_x(1,k-2)));
           P_non_derive_R_S_x(1,k)=((((2*k+1)/(k+1))*((t_1/
c)*gauss_points_x(j)+spar_position_R_S(1,k-1)))*P_non_derive_R_S_x(1,k-1)-
(k/(k+1)*P_non_derive_R_S_x(1,k-2)));
        end
        for k=3:number_of_nodes_I_J
           P_derive_I_J_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c)*gauss_points_y(op)+spar_position_I_J(1,k-1)))*P_derive_I_J_y(1,k-1)-
(k/(k+1)*P_derive_I_J_y(1,k-2)));
           P_non_derive_I_J_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c)*gauss_points_y(op)+spar_position_I_J(1,k-1)))*P_non_derive_I_J_y(1,k-1)-
(k/(k+1)*P_non_derive_I_J_y(1,k-2)));
        end
        for k=3:number_of_nodes_K_L
           P_derive_K_L_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c)*gauss_points_y(op)+spar_position_K_L(1,k-1)))*P_derive_K_L_x(1,k-1)-
(k/(k+1)*P_derive_K_L_x(1,k-2)));
           P_non_derive_K_L_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c)*gauss_points_y(op)+spar_position_K_L(1,k-1)))*P_non_derive_K_L_y(1,k-1)-
(k/(k+1)*P_non_derive_K_L_y(1,k-2)));
        end
%         for k=3:number_of_nodes_M_N
%            P_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(j))*P_derive_M_N_y(1,k-1)-(k/(k
+1)*P_derive_M_N_y(1,k-2)));
%            P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
%         end
        for k=3:number_of_nodes_P_Q
           P_derive_P_Q_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c)*gauss_points_y(op)+spar_position_P_Q(1,k-1)))*P_derive_P_Q_y(1,k-1)-
(k/(k+1)*P_derive_P_Q_y(1,k-2)));
```

```
            P_non_derive_P_Q_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c)*gauss_points_y(op)+spar_position_P_Q(1,k-1)))*P_non_derive_P_Q_y(1,k-1)-
(k/(k+1)*P_non_derive_P_Q_y(1,k-2)));
          end
          for k=3:number_of_nodes_R_S
             P_derive_R_S_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c)*gauss_points_y(op)+spar_position_R_S(1,k-1)))*P_derive_R_S_y(1,k-1)-
(k/(k+1)*P_derive_R_S_y(1,k-2)));
             P_non_derive_R_S_y(1,k)=((((2*k+1)/(k+1))*((t_1/
c)*gauss_points_y(op)+spar_position_R_S(1,k-1)))*P_non_derive_R_S_y(1,k-1)-
(k/(k+1)*P_non_derive_R_S_y(1,k-2)));
          end

 [detJacobian,inverse]=coordinate_transformation_Jacobian(sweep_angle,cr,b,gauss_p
          inverse_trans=inverse';
          final_IJ_x=P_derive_I_J_x.*P_non_derive_I_J_x;
          final_KL_x=P_derive_K_L_x.*P_non_derive_K_L_x;
         % final_MN_x=P_derive_M_N_x.*P_non_derive_M_N_x;
          final_PQ_x=P_derive_P_Q_x.*P_non_derive_P_Q_x;
          final_RS_x=P_derive_R_S_x.*P_non_derive_R_S_x;
          final_IJ_y=P_derive_I_J_y.*P_non_derive_I_J_y;
          final_KL_y=P_derive_K_L_y.*P_non_derive_K_L_y;
          %final_MN_y=P_derive_M_N_y.*P_non_derive_M_N_y;
          final_PQ_y=P_derive_P_Q_y.*P_non_derive_P_Q_y;
          final_RS_y=P_derive_R_S_y.*P_non_derive_R_S_y;
          boom=5;
          idk=1;
%           final_IJ=[final_IJ_x;final_IJ_y];
%           final_KL=[final_KL_x;final_KL_y];
%           final_MN=[final_MN_x;final_MN_y];
%           final_PQ=[final_PQ_x;final_PQ_y];
%           final_RS=[final_RS_x;final_RS_y];
          for pi=1:length(final_IJ_x)
              B(1:12,idk:boom)=B(1:12,idk:boom)+[final_IJ_x(1,pi)
 0 0 0 0;final_IJ_y(1,pi) 0 0 0 0;0 final_KL_x(1,pi) 0 0 0;0
 final_KL_y(1,pi) 0 0 0;0 0 0 0 0;0 0 0 0 0;0 0 0 final_PQ_x(1,pi)
 0;0 0 0 final_PQ_y(1,pi) 0;0 0 0 0 final_RS_x(1,pi);0 0 0 0
 final_RS_y(1,pi);0 0 0 0 0;0 0 0 0 0];
              idk=idk+5;
              boom=boom+5;
          end

 D=D_matrix_general_in_plane(sweep_angle,cr,b,theta,thickness_of_each_ply,gauss_po
+D_matrix_upper_airfoil_in_plane(sweep_angle,cr,b,theta,thickness_of_each_ply,gaus
          hope=hope+(t_1/
c)*B'*D*B*weights_x(j)*weights_y(op)*detJacobian;
      end
      end
%     iv_0=[5*n11_IJ-4 5*n11_IJ-3 5*n11_IJ-2 5*n11_IJ-1 5*n11_IJ
 5*n22_IJ-4 5*n22_IJ-3 5*n22_IJ-2 5*n22_IJ-1 5*n22_IJ 5*n33_IJ-4
 5*n33_IJ-3 5*n33_IJ-2 5*n33_IJ-1 5*n33_IJ 5*n44_IJ-4 5*n44_IJ-3
 5*n44_IJ-2 5*n44_IJ-1 5*n44_IJ];
%     iv_1=[5*n11_KL-4 5*n11_KL-3 5*n11_KL-2 5*n11_KL-1 5*n11_KL
 5*n22_KL-4 5*n22_KL-3 5*n22_KL-2 5*n22_KL-1 5*n22_KL 5*n33_KL-4
```

```
 5*n33_KL-3 5*n33_KL-2 5*n33_KL-1 5*n33_KL 5*n44_KL-4 5*n44_KL-3
 5*n44_KL-2 5*n44_KL-1 5*n44_KL];
%     iv_2=[5*n11_PQ-4 5*n11_PQ-3 5*n11_PQ-2 5*n11_PQ-1 5*n11_PQ
 5*n22_PQ-4 5*n22_PQ-3 5*n22_PQ-2 5*n22_PQ-1 5*n22_PQ 5*n33_PQ-4
 5*n33_PQ-3 5*n33_PQ-2 5*n33_PQ-1 5*n33_PQ 5*n44_PQ-4 5*n44_PQ-3
 5*n44_PQ-2 5*n44_PQ-1 5*n44_PQ];
%     iv_3=[5*n11_RS-4 5*n11_RS-3 5*n11_RS-2 5*n11_RS-1 5*n11_RS
 5*n22_RS-4 5*n22_RS-3 5*n22_RS-2 5*n22_RS-1 5*n22_RS 5*n33_RS-4
 5*n33_RS-3 5*n33_RS-2 5*n33_RS-1 5*n33_RS 5*n44_RS-4 5*n44_RS-3
 5*n44_RS-2 5*n44_RS-1 5*n44_RS];
%     iv=[iv_0 iv_1 iv_2 iv_3];
    %kg(iv,iv)=kg(iv,iv)+hope;
end
kg_in_plane=hope;
end
```

```
Not enough input arguments.

Error in B_matrix_in_plane_rib_web (line 13)
sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 kg_trans_plane=B_matrix_trans_plane(gauss_points_x,gauss_points_y,sweep_angle,b,c
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
% weights_x=[1 1];
% weights_y=[1 1];
% sweep_angle=30;%degree
% b=10;
pi=3.1416;
sweep_angle_in_radians=sweep_angle*pi/180;
% cr=5;
% l_1=1;
% c=1;
% h=1;
% h_1=1;
% t_1=1;
% t=1;
tip_chord=cr-((b/2)*tan(sweep_angle_in_radians));
%Linear nodes per element
% I=2;
% J=2;
% K=2;
% L=2;
% M=2;
% N=2;
% P=2;
% Q=2;
% R=2;
% S=2;
% dof=(5);
% icon_IJ=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_KL=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_PQ=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_RS=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
coeff_of_z_lower=Lower_fitting_a_curve();
coeff_of_z_upper=Upper_fitting_a_curve();
number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
%number_of_elements=number_of_nodes-1;
P_non_derive_I_J_x=zeros(1,I*J);
P_derive_I_J_x=zeros(1,I*J);
P_derive_K_L_x=zeros(1,K*L);
P_non_derive_K_L_x=zeros(1,K*L);
P_derive_M_N_x=zeros(1,M*N);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_derive_R_S_x=zeros(1,R*S);
```

```
P_non_derive_I_J_y=zeros(1,I*J);
P_derive_I_J_y=zeros(1,I*J);
P_derive_K_L_y=zeros(1,K*L);
P_non_derive_K_L_y=zeros(1,K*L);
P_derive_M_N_y=zeros(1,M*N);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_derive_R_S_y=zeros(1,R*S);
B=zeros(12,(I*J+P*Q+M*N+K*L+R*S));
% theta=[0 90 90 0];
% thickness_of_each_ply=0.01*ones(1,length(theta));
hope=zeros((I*J+P*Q+M*N+K*L+R*S),(I*J+P*Q+M*N+K*L+R*S));
spar_position_I_J=linspace(-1,1,I*J);
spar_position_M_N=linspace(-1,1,M*N);
spar_position_P_Q=linspace(-1,1,P*Q);
spar_position_R_S=linspace(-1,1,R*S);
%kg=zeros(((I*J)+(K*L)+(P*Q)+(R*S))*dof,
((I*J)+(K*L)+(P*Q)+(R*S))*dof);
for
 i=1:I*J%*length(icon_KL(:,1))*length(icon_KL(1,:))*length(icon_PQ(:,1))*length(ic
%     n11_IJ=icon_IJ(i,1);
%     n22_IJ=icon_IJ(i,2);
%     n33_IJ=icon_IJ(i,3);
%     n44_IJ=icon_IJ(i,4);
%     %n55_IJ=icon_IJ(i,5);
%     n11_KL=icon_KL(i,1);
%     n22_KL=icon_KL(i,2);
%     n33_KL=icon_KL(i,3);
%     n44_KL=icon_KL(i,4);
%     %n55_KL=icon_KL(i,5);
%     n11_PQ=icon_PQ(i,1);
%     n22_PQ=icon_PQ(i,2);
%     n33_PQ=icon_PQ(i,3);
%     n44_PQ=icon_PQ(i,4);
%     %n55_PQ=icon_PQ(i,5);
%     n11_RS=icon_RS(i,1);
%     n22_RS=icon_RS(i,2);
%     n33_RS=icon_RS(i,3);
%     n44_RS=icon_RS(i,4);
    %n55_RS=icon_RS(i,5);
    for op=1:length(gauss_points_y)
        c=0.5*cr*(1-
gauss_points_y(op))+0.5*tip_chord*(1+gauss_points_y(op));
    for j=1:length(gauss_points_x)
P_non_derive_I_J_x(1,1)=1;
P_non_derive_I_J_x(1,2)=(t_2/
c)*gauss_points_x(j)+spar_position_M_N(j);
P_derive_I_J_x(1,1)=0;
P_derive_I_J_x(1,2)=(t_2/c);
P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=(t_2/
c)*gauss_points_x(j)+spar_position_M_N(j);
```

```
P_derive_M_N_x(1,1)=0;
P_derive_M_N_x(1,2)=(t_2/c);
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=(t_2/
c)*gauss_points_x(j)+spar_position_M_N(j);
P_derive_M_N_x(1,1)=0;
P_derive_M_N_x(1,2)=(t_2/c);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=(t_2/
c)*gauss_points_x(j)+spar_position_M_N(j);
P_derive_P_Q_x(1,1)=0;
P_derive_P_Q_x(1,2)=(t_2/c);
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=(t_2/
c)*gauss_points_x(j)+spar_position_M_N(j);
P_derive_R_S_x(1,1)=0;
P_derive_R_S_x(1,2)=(t_2/c);
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=(t_2/
c)*gauss_points_y(op)+spar_position_M_N(op);
P_derive_I_J_y(1,1)=0;
P_derive_I_J_y(1,2)=(t_2/c);
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=(t_2/
c)*gauss_points_y(op)+spar_position_M_N(op);
P_derive_M_N_y(1,1)=0;
P_derive_M_N_y(1,2)=(t_2/c);
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=(t_2/
c)*gauss_points_y(op)+spar_position_M_N(op);
P_derive_M_N_y(1,1)=0;
P_derive_M_N_y(1,2)=(t_2/c);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=(t_2/
c)*gauss_points_y(op)+spar_position_M_N(op);
P_derive_P_Q_y(1,1)=0;
P_derive_P_Q_y(1,2)=(t_2/c);
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=(t_2/
c)*gauss_points_y(op)+spar_position_M_N(op);
P_derive_R_S_y(1,1)=0;
P_derive_R_S_y(1,2)=(t_2/c);
        for k=3:number_of_nodes_I_J
           P_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_I_J_x(1,k-1)-(k/(k
+1)*P_derive_I_J_x(1,k-2)));
           P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2)));
        end
        for k=3:number_of_nodes_K_L
           P_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_K_L_x(1,k-1)-(k/(k
+1)*P_derive_K_L_x(1,k-2)));
```

```
            P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
        end
        for k=3:number_of_nodes_M_N
            P_derive_M_N_x(1,k)=((((2*k+1)/(k+1))*(t_2/
c)*gauss_points_x(j)+spar_position_M_N(1,k-1))*P_derive_M_N_x(1,k-1)-
(k/(k+1)*P_derive_M_N_x(1,k-2)));
            P_non_derive_M_N_x(1,k)=((((2*k+1)/(k+1))*(t_2/
c)*gauss_points_x(j)+spar_position_M_N(1,k-1))*P_non_derive_M_N_x(1,k-1)-
(k/(k+1)*P_non_derive_M_N_x(1,k-2)));
        end
        for k=3:number_of_nodes_P_Q
            P_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_derive_P_Q_x(1,k-2)));
            P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
        end
        for k=3:number_of_nodes_R_S
            P_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_R_S_x(1,k-1)-(k/(k
+1)*P_derive_R_S_x(1,k-2)));
            P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
        end
        for k=3:number_of_nodes_I_J
            P_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(j))*P_derive_I_J_y(1,k-1)-(k/(k
+1)*P_derive_I_J_y(1,k-2)));
            P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
        end
        for k=3:number_of_nodes_K_L
            P_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(j))*P_derive_K_L_x(1,k-1)-(k/(k
+1)*P_derive_K_L_x(1,k-2)));
            P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
        end
        for k=3:number_of_nodes_M_N
            P_derive_M_N_y(1,k)=((((2*k+1)/(k+1))*(t_2/
c)*gauss_points_y(op)+spar_position_M_N(1,k-1))*P_derive_M_N_y(1,k-1)-
(k/(k+1)*P_derive_M_N_y(1,k-2)));
            P_non_derive_M_N_y(1,k)=((((2*k+1)/(k+1))*(t_2/
c)*gauss_points_y(op)+spar_position_M_N(1,k-1))*P_non_derive_M_N_y(1,k-1)-
(k/(k+1)*P_non_derive_M_N_y(1,k-2)));
        end
        for k=3:number_of_nodes_P_Q
```

```
            P_derive_P_Q_y(1,k)=((((2*k+1)/(k+1))*(t_2/
c)*gauss_points_y(op)+spar_position_P_Q(1,k-1))*P_derive_P_Q_y(1,k-1)-
(k/(k+1)*P_derive_P_Q_y(1,k-2)));
            P_non_derive_P_Q_y(1,k)=((((2*k+1)/(k+1))*(t_2/
c)*gauss_points_y(op)+spar_position_P_Q(1,k-1))*P_non_derive_P_Q_y(1,k-1)-
(k/(k+1)*P_non_derive_P_Q_y(1,k-2)));
        end
        for k=3:number_of_nodes_R_S
            P_derive_R_S_y(1,k)=((((2*k+1)/(k+1))*(t_2/
c)*gauss_points_y(op)+spar_position_R_S(1,k-1))*P_derive_R_S_y(1,k-1)-
(k/(k+1)*P_derive_R_S_y(1,k-2)));
            P_non_derive_R_S_y(1,k)=((((2*k+1)/(k+1))*(t_2/
c)*gauss_points_y(op)+spar_position_R_S(1,k-1))*P_non_derive_R_S_y(1,k-1)-
(k/(k+1)*P_non_derive_R_S_y(1,k-2)));
        end

 [detJacobian,inverse]=coordinate_transformation_Jacobian(sweep_angle,cr,b,gauss_p
        inverse_trans=inverse';
%         final_IJ_x=P_derive_I_J_x.*P_non_derive_I_J_x;
%         final_KL_x=P_derive_K_L_x.*P_non_derive_K_L_x;
        final_MN_x=P_derive_M_N_x.*P_non_derive_M_N_x;
%         final_PQ_x=P_derive_P_Q_x.*P_non_derive_P_Q_x;
%         final_RS_x=P_derive_R_S_x.*P_non_derive_R_S_x;
%         final_IJ_y=P_derive_I_J_y.*P_non_derive_I_J_y;
%         final_KL_y=P_derive_K_L_y.*P_non_derive_K_L_y;
        final_MN_y=P_derive_M_N_y.*P_non_derive_M_N_y;
%         final_PQ_y=P_derive_P_Q_y.*P_non_derive_P_Q_y;
%         final_RS_y=P_derive_R_S_y.*P_non_derive_R_S_y;
final_PQ=P_non_derive_P_Q_x.*P_non_derive_P_Q_y;
final_RS=P_non_derive_R_S_x.*P_non_derive_R_S_y;
        boom=5;
        idk=1;
%         final_IJ=[final_IJ_x;final_IJ_y];
%         final_KL=[final_KL_x;final_KL_y];
%         final_MN=[final_MN_x;final_MN_y];
%         final_PQ=[final_PQ_x;final_PQ_y];
%         final_RS=[final_RS_x;final_RS_y];
        for pi=1:length(final_MN_x)
            B(1:12,idk:boom)=B(1:12,idk:boom)+[0 0 0 0 0;0 0 0 0 0;0 0
 0 0 0;0 0 0 0 0;0 0 final_MN_x(1,pi) 0 0;0 0 final_MN_y(1,pi) 0 0;0
 0 0 0 0;0 0 0 0 0;0 0 0 0 0;0 0 0 0 0;0 0 0 final_PQ(1,pi) 0;0 0 0 0
 final_RS(1,pi)];
            idk=idk+5;
            boom=boom+5;
        end

 D=D_matrix_general_trans_plane(sweep_angle,cr,b,theta,thickness_of_each_ply,gauss
+D_matrix_upper_airfoil_trans_plane(sweep_angle,cr,b,theta,thickness_of_each_ply,g
        hope=hope+(t_2/
c)*B'*D*B*weights_x(j)*weights_y(op)*detJacobian;
    end
    end
%     iv_0=[5*n11_IJ-4 5*n11_IJ-3 5*n11_IJ-2 5*n11_IJ-1 5*n11_IJ
 5*n22_IJ-4 5*n22_IJ-3 5*n22_IJ-2 5*n22_IJ-1 5*n22_IJ 5*n33_IJ-4
```

```
 5*n33_IJ-3 5*n33_IJ-2 5*n33_IJ-1 5*n33_IJ 5*n44_IJ-4 5*n44_IJ-3
 5*n44_IJ-2 5*n44_IJ-1 5*n44_IJ];
%     iv_1=[5*n11_KL-4 5*n11_KL-3 5*n11_KL-2 5*n11_KL-1 5*n11_KL
 5*n22_KL-4 5*n22_KL-3 5*n22_KL-2 5*n22_KL-1 5*n22_KL 5*n33_KL-4
 5*n33_KL-3 5*n33_KL-2 5*n33_KL-1 5*n33_KL 5*n44_KL-4 5*n44_KL-3
 5*n44_KL-2 5*n44_KL-1 5*n44_KL];
%     iv_2=[5*n11_PQ-4 5*n11_PQ-3 5*n11_PQ-2 5*n11_PQ-1 5*n11_PQ
 5*n22_PQ-4 5*n22_PQ-3 5*n22_PQ-2 5*n22_PQ-1 5*n22_PQ 5*n33_PQ-4
 5*n33_PQ-3 5*n33_PQ-2 5*n33_PQ-1 5*n33_PQ 5*n44_PQ-4 5*n44_PQ-3
 5*n44_PQ-2 5*n44_PQ-1 5*n44_PQ];
%     iv_3=[5*n11_RS-4 5*n11_RS-3 5*n11_RS-2 5*n11_RS-1 5*n11_RS
 5*n22_RS-4 5*n22_RS-3 5*n22_RS-2 5*n22_RS-1 5*n22_RS 5*n33_RS-4
 5*n33_RS-3 5*n33_RS-2 5*n33_RS-1 5*n33_RS 5*n44_RS-4 5*n44_RS-3
 5*n44_RS-2 5*n44_RS-1 5*n44_RS];
%     iv=[iv_0 iv_1 iv_2 iv_3];
    %kg(iv,iv)=kg(iv,iv)+hope;
end
kg_trans_plane=hope;
end
```

```
Not enough input arguments.

Error in B_matrix_trans_plane_rib_web (line 9)
sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 D_matrix_1=D_matrix_general_in_plane(sweep,cr,b,theta,thickness_of_each_ply,gauss
% coeff_of_z_lower=Lower_fitting_a_curve();
% coeff_of_z_upper=Upper_fitting_a_curve();
% sweep=30;
% cr=10*10^-3;
% b=20*10^-3;
thickness_lower=0.01;
Z=zeros(5,12);
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
z_2=0;
z_derive_1_lower=0;
z_derive_2_lower=0;
z_derive_3_lower=0;
z_derive_1_upper=0;
z_derive_2_upper=0;
z_derive_3_upper=0;
n=10;
z_lower=[0.000000  0.000000;
  0.002000 -0.009300;
  0.005000 -0.016000;
  0.010000 -0.022100;
  0.020000 -0.029500;
  0.030000 -0.034400;
  0.040000 -0.038100;
  0.050000 -0.041200;
  0.070000 -0.046200;
  0.100000 -0.051700;
  0.120000 -0.054700;
  0.150000 -0.058500;
  0.170000 -0.060600;
  0.200000 -0.063300;
  0.220000 -0.064700;
  0.250000 -0.066600;
  0.280000 -0.068000;
  0.300000 -0.068700;
  0.320000 -0.069200;
  0.350000 -0.069600;
  0.370000 -0.069600;
  0.400000 -0.069200;
  0.420000 -0.068800;
  0.450000 -0.067600;
  0.480000 -0.065700;
  0.500000 -0.064400;
  0.530000 -0.061400;
  0.550000 -0.058800;
  0.580000 -0.054300;
  0.600000 -0.050900;
  0.630000 -0.045100;
  0.650000 -0.041000;
```

```
  0.680000 -0.034600;
  0.700000 -0.030200;
  0.730000 -0.023500;
  0.750000 -0.019200;
  0.770000 -0.015000;
  0.800000 -0.009300;
  0.830000 -0.004800;
  0.850000 -0.002400;
  0.870000 -0.001300;
  0.890000 -0.000800;
  0.920000 -0.001600;
  0.940000 -0.003500;
  0.950000 -0.004900;
  0.960000 -0.006600;
  0.970000 -0.008500;
  0.980000 -0.010900;
  0.990000 -0.013700;
  1.000000 -0.016300];
z_upper_curve=[0.000000  0.000000;
  0.002000  0.009500;
  0.005000  0.015800;
  0.010000  0.021900;
  0.020000  0.029300;
  0.030000  0.034300;
  0.040000  0.038100;
  0.050000  0.041100;
  0.070000  0.046200;
  0.100000  0.051800;
  0.120000  0.054800;
  0.150000  0.058500;
  0.170000  0.060600;
  0.200000  0.063200;
  0.220000  0.064600;
  0.250000  0.066400;
  0.270000  0.067300;
  0.300000  0.068500;
  0.330000  0.069200;
  0.350000  0.069600;
  0.380000  0.069800;
  0.400000  0.069700;
  0.430000  0.069500;
  0.450000  0.069200;
  0.480000  0.068400;
  0.500000  0.067800;
  0.530000  0.066600;
  0.550000  0.065600;
  0.570000  0.064500;
  0.600000  0.062500;
  0.620000  0.061000;
  0.650000  0.058500;
  0.680000  0.055500;
  0.700000  0.053300;
  0.720000  0.050900;
  0.750000  0.046900;
```

```
  0.770000  0.043900;
  0.800000  0.038900;
  0.820000  0.035300;
  0.850000  0.029400;
  0.870000  0.025100;
  0.900000  0.018100;
  0.920000  0.013100;
  0.950000  0.004900;
  0.970000 -0.000900;
  0.980000 -0.003900;
  0.990000 -0.007100;
  1.000000 -0.010400;
  ];
x=z_lower(1:end,1);
y=z_upper_curve(1:end,1);
y=y';
z_lower_1=z_lower(1:end,1);
for po=1:length(coeff_of_z_lower)
    z_derive_1_lower=z_derive_1_lower
+(coeff_of_z_lower(po)*(((factorial(length(coeff_of_z_lower))/
(factorial(length(coeff_of_z_lower)-
po)*factorial(po))))*po*(length(coeff_of_z_lower)-
po)*((z_lower_1(po))^(po-1)))*((z_lower_1(po))^0.75)*((1-
z_lower(po))^0.75));
    z_derive_2_lower=z_derive_2_lower
+(coeff_of_z_lower(po))*(factorial(length(coeff_of_z_lower))/
((factorial(length(coeff_of_z_lower)-
po)*factorial(po))))*(0.75)*(z_lower_1(po))^(0.75-1)*(1-
z_lower_1(po))^0.75;
     z_derive_3_lower=z_derive_3_lower
+(coeff_of_z_lower(po)*(((factorial(length(coeff_of_z_lower))/
(factorial(length(coeff_of_z_lower))*(0.75*(1-
z_lower_1(po))^(0.75-1)*(z_lower_1(po))^(0.75)))))));
end
for po=1:length(coeff_of_z_upper)
    z_derive_1_upper=z_derive_1_upper
+(coeff_of_z_upper(po)*(((factorial(length(coeff_of_z_upper))/
(factorial(length(coeff_of_z_upper)-
po)*factorial(po))))*po*(length(coeff_of_z_upper)-
po)*((z_upper_curve(po))^(po-1)))*((z_upper_curve(po))^0.75)*((1-
z_upper_curve(po))^0.75));
    z_derive_2_upper=z_derive_2_upper
+(coeff_of_z_upper(po))*(factorial(length(coeff_of_z_upper))/
((factorial(length(coeff_of_z_upper)-
po)*factorial(po))))*(0.75)*(z_upper_curve(po))^(0.75-1)*(1-
z_upper_curve(po))^0.75;
     z_derive_3_upper=z_derive_3_upper
+(coeff_of_z_upper(po)*(((factorial(length(coeff_of_z_upper))/
(factorial(length(coeff_of_z_upper))*(0.75*(1-
z_upper_curve(po))^(0.75-1)*(z_upper_curve(po))^(0.75)))))));
end
z_derive_lower=z_derive_1_lower+z_derive_2_lower+z_derive_3_lower;
y_lower=atan(z_derive_lower);
y_lower=(y_lower*180/pi);
```

```
thickness_lower_1=thickness_lower*sqrt(1+(tan(y_lower))^2);
z_derive_upper=z_derive_1_upper+z_derive_2_upper+z_derive_3_upper;
y_upper=atan(z_derive_upper);
y_upper=(y_upper*180/pi);
thickness_upper_1=thickness_lower*sqrt(1+(tan(y_upper))^2);
%step_size=thickness_lower_1/n;
%for k=1:length(coeff_of_z_lower)
%theta=[0 45 45 0];
% thickness_of_each_ply=0.1*(ones(1,4));
z_1=(((x).^0.75).*(1-x).^(0.75));
z_2=(((y).^0.75).*(1-y).^(0.75));
z_2_lower=0;
z_2_upper=0;
x=x';
for k=1:length(x)
z_2_lower=z_2_lower+(factorial(length(coeff_of_z_lower))/
(factorial(length(coeff_of_z_lower)-
k)*factorial(k)))*coeff_of_z_lower*(((x).^k).*(ones(1,length(x))-
x).^(length(coeff_of_z_lower)-k))';
end
for k=1:length(coeff_of_z_upper)
z_2_upper=z_2_upper+(factorial(length(coeff_of_z_upper))/
(factorial(length(coeff_of_z_upper)-
k)*factorial(k)))*coeff_of_z_upper*(((y).^k).*(ones(1,length(coeff_of_z_upper))-
y).^(length(coeff_of_z_upper)-k))';
end
z_1=z_1';
z_lower_hope=z_1+z_2_lower;
z_upper_hope=z_2+z_2_upper;
z_upper=z_upper_hope-ones(1,length(y))*(thickness_upper_1/2)-
ones(1,length(y))*h_1;
z_l_lower=z_lower_hope
+(ones(1,length(x))*(thickness_lower_1/2))+ones(1,length(x))*h_1;
z_upper=[z_upper 0 0];
h=(z_upper-z_l_lower)./n;
integral_final=letsintegrate(z_upper,z_l_lower,sweep,cr,b,gauss_points_x,gauss_poi
for i=1:length(n)
    k=z_l_lower+i*h;
    for lo=1:length(z_l_lower)
    if rem(k(lo),3)==0
        integral_final=integral_final
+2*(function_1(k(lo),sweep,cr,b,gauss_points_x,gauss_points_y));
    else
        integral_final=integral_final
+3*(function_1(k(lo),sweep,cr,b,gauss_points_x,gauss_points_y));
    end
    end
    integral_final=integral_final*3*h(1)/8;
end
[stiffness]=abd_matrix(theta,thickness_of_each_ply);
D_matrix_1=integral_final'*stiffness*integral_final;
function func=function_1(k,sweep,cr,b,gauss_points_x,gauss_points_y)
% sweep=30;
% cr=10;
```

```
% b=20;
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
inverse_trans=inverse';
func=[(inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0
 0 0 k*(inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0
 0 0;0 0 0 (inverse_trans(1,2)+inverse_trans(2,1)) 0
 0 0 0 0 (k)*(inverse_trans(1,2)+inverse_trans(2,2))
 0 0;0 (inverse_trans(1,2)+inverse_trans(2,1))
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0
 0 k*(inverse_trans(2,2)+inverse_trans(2,1))
 k*(inverse_trans(1,1)+inverse_trans(1,2)) 0 0 0];
end
function
 integral=letsintegrate(z_pp,z_l_lower,sweep,cr,b,gauss_points_x,gauss_points_y)
% sweep=30;
% cr=10*10^-3;
% b=20*10^-3;
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
inverse_trans=inverse';
integration_1=zeros(3,12);
integration_2=zeros(3,12);
for i=1:length(z_pp)
integration_1=integration_1+[(inverse_trans(1,1)+inverse_trans(2,1))
 0 0 0 0 0 z_pp(i)*(inverse_trans(1,1)+inverse_trans(2,1))
 0 0 0 0 0;0 0 0 (inverse_trans(1,2)+inverse_trans(2,1)) 0
 0 0 0 0 (z_pp(i))*(inverse_trans(1,2)+inverse_trans(2,2))
 0 0;0 (inverse_trans(1,2)+inverse_trans(2,1))
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0 0
 z_pp(i)*(inverse_trans(2,2)+inverse_trans(2,1))
 z_pp(i)*(inverse_trans(1,1)+inverse_trans(1,2)) 0 0 0];
integration_2=integration_2+[(inverse_trans(1,1)+inverse_trans(2,1))
 0 0 0 0 0 z_l_lower(i)*(inverse_trans(1,1)+inverse_trans(2,1))
 0 0 0 0 0;0 0 0 (inverse_trans(1,2)+inverse_trans(2,1)) 0 0
 0 0 0 (z_l_lower(i))*(inverse_trans(1,2)+inverse_trans(2,2))
 0 0;0 (inverse_trans(1,2)+inverse_trans(2,1))
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0 0
 z_l_lower(i)*(inverse_trans(2,2)+inverse_trans(2,1))
 z_l_lower(i)*(inverse_trans(1,1)+inverse_trans(1,2)) 0 0 0];
end
integral=integration_1+integration_2;
end
end
```

```
Not enough input arguments.

Error in D_matrix_general_in_plane (line 11)
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
```

Published with MATLAB® R2020a

```
function
 D_matrix_trans_1=D_matrix_general_trans_plane(sweep,cr,b,theta,thickness_of_each_
% coeff_of_z_lower=Lower_fitting_a_curve();
% coeff_of_z_upper=Upper_fitting_a_curve();
% sweep=30;
% cr=10*10^-3;
% b=20*10^-3;
% h_1=1;
% h=1;
thickness_lower=0.01;
Z=zeros(5,12);
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
z_2=0;
z_derive_l_lower=0;
z_derive_2_lower=0;
z_derive_3_lower=0;
z_derive_l_upper=0;
z_derive_2_upper=0;
z_derive_3_upper=0;
n=50;
z_lower_l=[0.000000  0.000000;
  0.002000 -0.009300;
  0.005000 -0.016000;
  0.010000 -0.022100;
  0.020000 -0.029500;
  0.030000 -0.034400;
  0.040000 -0.038100;
  0.050000 -0.041200;
  0.070000 -0.046200;
  0.100000 -0.051700;
  0.120000 -0.054700;
  0.150000 -0.058500;
  0.170000 -0.060600;
  0.200000 -0.063300;
  0.220000 -0.064700;
  0.250000 -0.066600;
  0.280000 -0.068000;
  0.300000 -0.068700;
  0.320000 -0.069200;
  0.350000 -0.069600;
  0.370000 -0.069600;
  0.400000 -0.069200;
  0.420000 -0.068800;
  0.450000 -0.067600;
  0.480000 -0.065700;
  0.500000 -0.064400;
  0.530000 -0.061400;
  0.550000 -0.058800;
  0.580000 -0.054300;
  0.600000 -0.050900;
```

```
  0.630000 -0.045100;
  0.650000 -0.041000;
  0.680000 -0.034600;
  0.700000 -0.030200;
  0.730000 -0.023500;
  0.750000 -0.019200;
  0.770000 -0.015000;
  0.800000 -0.009300;
  0.830000 -0.004800;
  0.850000 -0.002400;
  0.870000 -0.001300;
  0.890000 -0.000800;
  0.920000 -0.001600;
  0.940000 -0.003500;
  0.950000 -0.004900;
  0.960000 -0.006600;
  0.970000 -0.008500;
  0.980000 -0.010900;
  0.990000 -0.013700;
  1.000000 -0.016300];
z_upper_curve=[0.000000  0.000000;
  0.002000  0.009500;
  0.005000  0.015800;
  0.010000  0.021900;
  0.020000  0.029300;
  0.030000  0.034300;
  0.040000  0.038100;
  0.050000  0.041100;
  0.070000  0.046200;
  0.100000  0.051800;
  0.120000  0.054800;
  0.150000  0.058500;
  0.170000  0.060600;
  0.200000  0.063200;
  0.220000  0.064600;
  0.250000  0.066400;
  0.270000  0.067300;
  0.300000  0.068500;
  0.330000  0.069200;
  0.350000  0.069600;
  0.380000  0.069800;
  0.400000  0.069700;
  0.430000  0.069500;
  0.450000  0.069200;
  0.480000  0.068400;
  0.500000  0.067800;
  0.530000  0.066600;
  0.550000  0.065600;
  0.570000  0.064500;
  0.600000  0.062500;
  0.620000  0.061000;
  0.650000  0.058500;
  0.680000  0.055500;
  0.700000  0.053300;
```

```
  0.720000  0.050900;
  0.750000  0.046900;
  0.770000  0.043900;
  0.800000  0.038900;
  0.820000  0.035300;
  0.850000  0.029400;
  0.870000  0.025100;
  0.900000  0.018100;
  0.920000  0.013100;
  0.950000  0.004900;
  0.970000 -0.000900;
  0.980000 -0.003900;
  0.990000 -0.007100;
  1.000000 -0.010400;
  ];
x=z_lower_l(1:end,1);
y_1=z_upper_curve(1:end,1);
%z_lower_1=z_lower(1:end,1);
for po=1:length(coeff_of_z_lower)
    z_derive_l_lower=z_derive_l_lower
+(coeff_of_z_lower(po)*(((factorial(length(coeff_of_z_lower))/
(factorial(length(coeff_of_z_lower)-
po)*factorial(po))))*po*(length(coeff_of_z_lower)-
po)*((z_lower_l(po))^(po-1)))*((z_lower_l(po))^0.75)*((1-
z_lower_l(po))^0.75));
    z_derive_2_lower=z_derive_2_lower
+(coeff_of_z_lower(po))*(factorial(length(coeff_of_z_lower))/
((factorial(length(coeff_of_z_lower)-
po)*factorial(po))))*(0.75)*(z_lower_l(po))^(0.75-1)*(1-
z_lower_l(po))^0.75;
     z_derive_3_lower=z_derive_3_lower
+(coeff_of_z_lower(po)*(((factorial(length(coeff_of_z_lower))/
(factorial(length(coeff_of_z_lower))*(0.75*(1-
z_lower_l(po))^(0.75-1)*(z_lower_l(po))^(0.75)))))));
end
for po=1:length(coeff_of_z_upper)
    z_derive_l_upper=z_derive_l_upper
+(coeff_of_z_upper(po)*(((factorial(length(coeff_of_z_upper))/
(factorial(length(coeff_of_z_upper)-
po)*factorial(po))))*po*(length(coeff_of_z_upper)-
po)*((z_upper_curve(po))^(po-1)))*((z_upper_curve(po))^0.75)*((1-
z_upper_curve(po))^0.75));
    z_derive_2_upper=z_derive_2_upper
+(coeff_of_z_upper(po))*(factorial(length(coeff_of_z_upper))/
((factorial(length(coeff_of_z_upper)-
po)*factorial(po))))*(0.75)*(z_upper_curve(po))^(0.75-1)*(1-
z_upper_curve(po))^0.75;
     z_derive_3_upper=z_derive_3_upper
+(coeff_of_z_upper(po)*(((factorial(length(coeff_of_z_upper))/
(factorial(length(coeff_of_z_upper))*(0.75*(1-
z_upper_curve(po))^(0.75-1)*(z_upper_curve(po))^(0.75)))))));
end
z_derive_lower=z_derive_l_lower+z_derive_2_lower+z_derive_3_lower;
y=atan(z_derive_lower);
```

```
y=(y*180/pi);
thickness_lower_1=thickness_lower*sqrt(1+(tan(y))^2);
z_derive_upper=z_derive_l_upper+z_derive_2_upper+z_derive_3_upper;
y_upper=atan(z_derive_upper);
y_upper=(y_upper*180/pi);
thickness_upper_1=thickness_lower*sqrt(1+(tan(y_upper))^2);
%step_size=thickness_lower_1/n;
%for k=1:length(coeff_of_z_lower)
% theta=[0 90 90 0];
% thickness_of_each_ply=0.1*(ones(1,4));
z_1=(((x).^0.75).*(1-x).^(0.75));
z_2=(((y_1).^0.75).*(1-y_1).^(0.75));
z_2_lower=0;
z_2_upper=0;
x=x';
for k=1:length(x)
z_2_lower=z_2_lower+(factorial(length(coeff_of_z_lower))/
(factorial(length(coeff_of_z_lower)-
k)*factorial(k)))*coeff_of_z_lower*(((x).^k).*(ones(1,length(x))-
x).^(length(coeff_of_z_lower)-k))';
end
for k=1:length(coeff_of_z_upper)
z_2_upper=z_2_upper+(factorial(length(coeff_of_z_upper))/
(factorial(length(coeff_of_z_upper)-
k)*factorial(k)))*coeff_of_z_upper*(((y_1).^k).*(ones(1,length(coeff_of_z_upper))-
y_1).^(length(coeff_of_z_upper)-k))';
end
z_1=z_1';
z_2=z_2';
z_lower_hope=z_1+z_2_lower;
z_upper_hope=z_2+z_2_upper;
z_upper=z_upper_hope-ones(1,length(y))*(thickness_upper_1/2)-
ones(1,length(y))*h_2;
z_l_lower=z_lower_hope
+(ones(1,length(x))*(thickness_lower_1/2))+ones(1,length(x))*h;
z_upper=z_upper;
z_upper=[z_upper 0 0];
h_2=abs((z_upper-z_l_lower))./n;
integral_final=letsintegrate(z_upper,z_l_lower,sweep,cr,b,gauss_points_x,gauss_poi
for i=1:length(n)
    k=z_l_lower+i*h_2;
    for lo=1:length(z_l_lower)
    if rem(k(lo),3)==0
        integral_final=integral_final
+2*(function_1(k(lo),sweep,cr,b,gauss_points_x,gauss_points_y));
    else
        integral_final=integral_final
+3*(function_1(k(lo),sweep,cr,b,gauss_points_x,gauss_points_y));
    end
    end
    integral_final=integral_final*3*h_2(1)/8;
end
[stiffness]=abd_matrix_trans(theta,thickness_of_each_ply);
D_matrix_trans_1=integral_final'*stiffness*integral_final;
```

```
function func=function_1(k,sweep,cr,b,gauss_points_x,gauss_points_y)
% sweep=30;
% cr=10;
% b=20;
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
inverse_trans=inverse';
func=[0 0 0 0 0 (inverse_trans(1,2)+inverse_trans(2,2)) 0 0 0 0 0 1;0
 0 0 0 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0 0 0 1 0];
end
function
 integral=letsintegrate(z_pp,z_l_lower,sweep,cr,b,gauss_points_x,gauss_points_y)
% sweep=30;
% cr=10*10^-3;
% b=20*10^-3;
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
inverse_trans=inverse';
integration_1=zeros(2,12);
integration_2=zeros(2,12);
for i=1:length(z_pp)
integration_1=integration_1+[0 0 0 0 0
 (inverse_trans(1,2)+inverse_trans(2,2)) 0 0 0 0 0 1;0 0 0 0
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0 0 0 1 0];
integration_2=integration_2+[0 0 0 0 0
 (inverse_trans(1,2)+inverse_trans(2,2)) 0 0 0 0 0 1;0 0 0 0
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0 0 0 1 0];
end
integral=integration_1+integration_2;
end
end
```

```
Not enough input arguments.

Error in D_matrix_general_trans_plane (line 13)
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
```

Published with MATLAB® R2020a

```
function
 Stiffness_rib_web=FEM_Main_file_rib_web(gauss_points_in_x,gauss_points_in_y,gauss
% gauss_points_in_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_trans_x=0;
% gauss_points_in_y=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_trans_y=0;
% weights_in_x=[1 1];
% weights_in_y=[1 1];
% weights_trans_x=2;
% weights_trans_y=2;
% sweep=30;%degree
% b=10;
% cr=5;
% I=4;
% J=4;
% K=4;
% L=4;
% M=4;
% N=4;
% P=4;
% Q=4;
% R=4;
% S=4;
% l_1=1;
% %c=1;
% h=1;
% h_2=1;
% t_2=1;
% theta=[0 90 90 0];
% thickness_of_each_ply=1*ones(1,length(theta));
B_in_plane=B_matrix_in_plane_rib_web(gauss_points_in_x,gauss_points_in_y,sweep,b,c
B_trans_plane=B_matrix_trans_plane_rib_web(gauss_points_trans_x,gauss_points_trans
Stiffness_rib_web=B_in_plane+B_trans_plane;
end
```

```
Not enough input arguments.

Error in FEM_Main_file_rib_web (line 30)
B_in_plane=B_matrix_in_plane_rib_web(gauss_points_in_x,gauss_points_in_y,sweep,b,c
```

Published with MATLAB® R2020a

```
function curve_fit_Lower=Lower_fitting_a_curve()
z_lower=[0.000000  0.000000;
  0.002000 -0.009300;
  0.005000 -0.016000;
  0.010000 -0.022100;
  0.020000 -0.029500;
  0.030000 -0.034400;
  0.040000 -0.038100;
  0.050000 -0.041200;
  0.070000 -0.046200;
  0.100000 -0.051700;
  0.120000 -0.054700;
  0.150000 -0.058500;
  0.170000 -0.060600;
  0.200000 -0.063300;
  0.220000 -0.064700;
  0.250000 -0.066600;
  0.280000 -0.068000;
  0.300000 -0.068700;
  0.320000 -0.069200;
  0.350000 -0.069600;
  0.370000 -0.069600;
  0.400000 -0.069200;
  0.420000 -0.068800;
  0.450000 -0.067600;
  0.480000 -0.065700;
  0.500000 -0.064400;
  0.530000 -0.061400;
  0.550000 -0.058800;
  0.580000 -0.054300;
  0.600000 -0.050900;
  0.630000 -0.045100;
  0.650000 -0.041000;
  0.680000 -0.034600;
  0.700000 -0.030200;
  0.730000 -0.023500;
  0.750000 -0.019200;
  0.770000 -0.015000;
  0.800000 -0.009300;
  0.830000 -0.004800;
  0.850000 -0.002400;
  0.870000 -0.001300;
  0.890000 -0.000800;
  0.920000 -0.001600;
  0.940000 -0.003500;
  0.950000 -0.004900;
  0.960000 -0.006600;
  0.970000 -0.008500;
  0.980000 -0.010900;
  0.990000 -0.013700;
  1.000000 -0.016300];
x_guess=ones(1,length(z_lower));
```

```
airfoil_equation_lower_curve_2=@airfoil_equation_lower_curve;
curve_fit_Lower=lsqcurvefit(airfoil_equation_lower_curve_2,x_guess,z_lower(1:end,1
z_predict=z_lower(1:end,1)';
thickness_ratio_lower_surface=0.01;
S_predict=zeros(length(z_lower),length(z_lower));
for i=1:length(z_predict)
   S_predict(i,1:length(z_predict))=(factorial(length(z_lower))/
(factorial(length(z_lower)-
i)*factorial(i)))*(((z_predict).^i).*((ones(1,length(z_predict))-
z_predict).^(length(z_predict)-i)));
end
% predicited_graph=((z_predict).^0.75).*((ones(1,length(z_predict))-
z_predict).^0.75).*(curve_fit_Lower*S_predict)+ones(1,length(z_predict))*thickness
% plot(z_lower(1:end,1)',predicited_graph,"b")
% hold on
% plot(z_lower(1:end,1),z_lower(1:end,2),"bo")
% hold off
function z=airfoil_equation_lower_curve(x,x_data)
x_data=x_data';
thickness_ratio_lower_surface=0.01;
S=zeros(length(x_data),length(x_data));
for i=1:length(x_data)
   S(i,1:length(x_data))=(factorial(length(x_data))/
(factorial(length(x_data)-
i)*factorial(i)))*(((x_data).^i).*((ones(1,length(x_data))-
x_data).^(length(x_data)-i)));
end
matrix=zeros(1,length(x_data));
for i=1:length(x_data)
    matrix(1,i)=x(i);
end
z=((x_data).^0.75).*((ones(1,length(x_data))-
x_data).^0.75).*(matrix*S)+ones(1,length(x_data))*thickness_ratio_lower_surface;
end
end
```

```
Local minimum found.

Optimization completed because the size of the gradient is less than
the value of the optimality tolerance.


ans =

  Columns 1 through 7

   -6.8476   16.4317  -26.3682   19.0586    1.1947  -12.2233   -0.0915

  Columns 8 through 14

    7.5818    1.1945   -5.4205   -3.4790    2.0104    3.4490    0.2375

  Columns 15 through 21
```

```
  -2.8556   -2.4605    0.2447    1.9473    1.0842   -0.9910   -1.9690

Columns 22 through 28

  -1.0438    0.5741    1.1811    0.3165   -0.9685   -1.3421   -0.5335

Columns 29 through 35

   0.5210    0.7394   -0.0024   -0.8314   -0.8664   -0.1462    0.4839

Columns 36 through 42

   0.3620   -0.2620   -0.5966   -0.3027    0.1513    0.2203    0.0051

Columns 43 through 49

  -0.1396   -0.2629   -0.2347    0.4094    0.2315   -1.3475    1.2399

Column 50

  -1.7220
```

Published with MATLAB® R2020a

```
function C=ques_1_compliance()
% prompt="Enter the value of E";
% prompt_1="Enter the value of poisson's ratio";
% prompt_2="Enter the value of G";
E1=70;
E2=70;
%E3=input(prompt);
poisson_ratio_12=0.1;
%poisson_ratio_13=input(prompt_1);
%poisson_ratio_23=input(prompt_1);
G12=5;
%G13=input(prompt_2);
%G23=input(prompt_2);
S=zeros(3,3);
S(1,1)=1/E1;
S(1,2)=-poisson_ratio_12/E2;
%S(1,3)=poisson_ratio_13/E3;
S(2,1)=-poisson_ratio_12/E1;
S(2,2)=1/E2;
%S(2,3)=poisson_ratio_23/E3;
%S(3,1)=poisson_ratio_13/E1;
%S(3,2)=poisson_ratio_23/E2;
%S(3,3)=1/E3;
%S(4,4)=1/G23;
%S(5,5)=1/G13;
S(3,3)=1/G12;
C=inv(S);
end
```

```
ans =

   70.7071    7.0707         0
    7.0707   70.7071         0
         0         0    5.0000
```

Published with MATLAB® R2020a

```
function C=ques_1_compliance_trans_plane()
% prompt="Enter the value of E";
% prompt_1="Enter the value of poisson's ratio";
% prompt_2="Enter the value of G";
%E1=15;
%E2=20;
%E3=input(prompt);
%poisson_ratio_12=0.3;
%poisson_ratio_13=input(prompt_1);
%poisson_ratio_23=input(prompt_1);
%G12=5;
G13=2;
G23=2;
S=zeros(2,2);
S(1,1)=1/G13;
S(1,2)=0;
%S(1,3)=poisson_ratio_13/E3;
S(2,1)=0;
S(2,2)=1/G23;
%S(2,3)=poisson_ratio_23/E3;
%S(3,1)=poisson_ratio_13/E1;
%S(3,2)=poisson_ratio_23/E2;
%S(3,3)=1/E3;
%S(4,4)=1/G23;
%S(5,5)=1/G13;
%S(3,3)=1/G12;
C=inv(S);
end
```

```
ans =

     2     0
     0     2
```

Published with MATLAB® R2020a

```
function Q=ques_2_Q_matrix_in_plane()
Stiffness=ques_1_compliance_in_plane();
Q=zeros(3,3);
Q(1,1)=Stiffness(1,1);%-(Stiffness(1,3)^2/Stiffness(3,3));
Q(1,2)=Stiffness(1,2);%-((Stiffness(1,3)*Stiffness(2,3))/
Stiffness(3,3));
Q(2,2)=Stiffness(2,2);%-(Stiffness(2,3)^2/Stiffness(3,3));
Q(3,3)=Stiffness(3,3);
Q(2,1)=Q(1,2);
end
```

```
ans =

   70.7071    7.0707         0
    7.0707   70.7071         0
         0         0    5.0000
```

Published with MATLAB® R2020a

```
function Q=ques_2_Q_matrix_trans_plane()
Stiffness=ques_1_compliance_trans_plane();
Q=Stiffness;
% Q(1,1)=Stiffness(1,1);%-(Stiffness(1,3)^2/Stiffness(3,3));
% Q(1,2)=Stiffness(1,2);%-((Stiffness(1,3)*Stiffness(2,3))/
Stiffness(3,3));
% Q(2,2)=Stiffness(2,2);%-(Stiffness(2,3)^2/Stiffness(3,3));
% Q(3,3)=Stiffness(3,3);
% Q(2,1)=Q(1,2);
end
```

```
ans =

     2     0
     0     2
```

Published with MATLAB® R2020a

```
function C=ques_3_with_arguments_transformed(theta)
%prompt="Enter the value of theta in degree";
%theta=input(prompt);
theta=(pi*theta)/180;
C=zeros(3,3);
C(1,1)=cos(theta)^2;
C(1,2)=sin(theta)^2;
C(1,3)=2*cos(theta)*sin(theta);
C(2,1)=sin(theta)^2;
C(2,2)=cos(theta)^2;
C(2,3)=-2*cos(theta)*sin(theta);
C(3,1)=-cos(theta)*sin(theta);
C(3,2)=cos(theta)*sin(theta);
C(3,3)=cos(theta)^2-sin(theta)^2;
end
```

```
Not enough input arguments.

Error in ques_3_with_arguments_transformed_in_plane (line 4)
theta=(pi*theta)/180;
```

Published with MATLAB® R2020a

```
function C=ques_3_with_arguments_transformed_trans_plane(theta)
%prompt="Enter the value of theta in degree";
%theta=input(prompt);
%theta=30;
theta=(pi*theta)/180;
C=zeros(2,2);
C(1,1)=cos(theta);
C(1,2)=-sin(theta);
%C(1,3)=2*cos(theta)*sin(theta);
C(2,1)=-sin(theta);
C(2,2)=cos(theta);
% C(2,3)=-2*cos(theta)*sin(theta);
% C(3,1)=-cos(theta)*sin(theta);
% C(3,2)=cos(theta)*sin(theta);
% C(3,3)=cos(theta)^2-sin(theta)^2;
end
```

```
Not enough input arguments.

Error in ques_3_with_arguments_transformed_trans_plane (line 5)
theta=(pi*theta)/180;
```

Published with MATLAB® R2020a

```
function curve_fit_upper=Upper_fitting_a_curve()
z_upper=[0.000000  0.000000;
  0.002000  0.009500;
  0.005000  0.015800;
  0.010000  0.021900;
  0.020000  0.029300;
  0.030000  0.034300;
  0.040000  0.038100;
  0.050000  0.041100;
  0.070000  0.046200;
  0.100000  0.051800;
  0.120000  0.054800;
  0.150000  0.058500;
  0.170000  0.060600;
  0.200000  0.063200;
  0.220000  0.064600;
  0.250000  0.066400;
  0.270000  0.067300;
  0.300000  0.068500;
  0.330000  0.069200;
  0.350000  0.069600;
  0.380000  0.069800;
  0.400000  0.069700;
  0.430000  0.069500;
  0.450000  0.069200;
  0.480000  0.068400;
  0.500000  0.067800;
  0.530000  0.066600;
  0.550000  0.065600;
  0.570000  0.064500;
  0.600000  0.062500;
  0.620000  0.061000;
  0.650000  0.058500;
  0.680000  0.055500;
  0.700000  0.053300;
  0.720000  0.050900;
  0.750000  0.046900;
  0.770000  0.043900;
  0.800000  0.038900;
  0.820000  0.035300;
  0.850000  0.029400;
  0.870000  0.025100;
  0.900000  0.018100;
  0.920000  0.013100;
  0.950000  0.004900;
  0.970000 -0.000900;
  0.980000 -0.003900;
  0.990000 -0.007100;
  1.000000 -0.010400;
  ];
x_guess=ones(1,length(z_upper));
airfoil_equation_lower_curve_2=@airfoil_equation_lower_curve;
```

```
curve_fit_upper=lsqcurvefit(airfoil_equation_lower_curve_2,x_guess,z_upper(1:end,1
z_predict=z_upper(1:end,1)';
thickness_ratio_lower_surface=0.01;
S_predict=zeros(length(z_upper),length(z_upper));
for i=1:length(z_predict)
   S_predict(i,1:length(z_predict))=(factorial(length(z_upper))/
(factorial(length(z_upper)-
i)*factorial(i)))*(((z_predict).^i).*((ones(1,length(z_predict))-
z_predict).^(length(z_predict)-i)));
end
% predicited_graph=((z_predict).^0.75).*((ones(1,length(z_predict))-
z_predict).^0.75).*(curve_fit_upper*S_predict)+ones(1,length(z_predict))*thickness
% plot(z_upper(1:end,1)',predicited_graph,"b")
% hold on
% plot(z_upper(1:end,1),z_upper(1:end,2),"bo")
% hold off
function z=airfoil_equation_lower_curve(x,x_data)
x_data=x_data';
thickness_ratio_lower_surface=0.01;
S=zeros(length(x_data),length(x_data));
for i=1:length(x_data)
   S(i,1:length(x_data))=(factorial(length(x_data))/
(factorial(length(x_data)-
i)*factorial(i)))*(((x_data).^i).*((ones(1,length(x_data))-
x_data).^(length(x_data)-i)));
end
matrix=zeros(1,length(x_data));
for i=1:length(x_data)
    matrix(1,i)=x(i);
end
z=((x_data).^0.75).*((ones(1,length(x_data))-
x_data).^0.75).*(matrix*S)+ones(1,length(x_data))*thickness_ratio_lower_surface;
end
end
```

Local minimum found.

Optimization completed because the size of the gradient is less than the value of the optimality tolerance.

ans =

Columns 1 through 7

1.4012 -1.1876 1.2572 0.4493 -0.2935 0.0879 0.5228

Columns 8 through 14

0.4273 0.1085 -0.0065 0.1345 0.3086 0.3318 0.2166

Columns 15 through 21

```
    0.0980    0.0795    0.1532    0.2356    0.2548    0.2051    0.1366

  Columns 22 through 28

    0.1040    0.1243    0.1720    0.2059    0.2026    0.1689    0.1313

  Columns 29 through 35

    0.1141    0.1246    0.1522    0.1777    0.1829    0.1583    0.1120

  Columns 36 through 42

    0.0756    0.0893    0.1579    0.2111    0.1484   -0.0132   -0.0575

  Columns 43 through 48

    0.1703    0.3062   -0.1698   -0.3775    0.6231   -1.1358
```

Published with MATLAB® R2020a

5.5 Rib Cap

```
function stiffness=abd_matrix(theta_of_each_ply,~)
%theta_of_each_ply=[0 45 45 0];
%thickness_of_each_ply=ones(1,length(theta_of_each_ply));
% height_of_lamina=sum(thickness_of_each_ply);
% ords1=zeros(1,(length(thickness_of_each_ply)+1));
% ords1(1,1)=-height_of_lamina/2;
% for i=2:(length(thickness_of_each_ply))
%     ords1(1,i)=ords1(1,i-1)+thickness_of_each_ply(1,i);
% end
% ords1(1,end)=abs(ords1(1,1));
Q_transformed_1=zeros(3,3);
for i=1:length(theta_of_each_ply)

 Q_transformed=inv(ques_3_with_arguments_transformed_in_plane(theta_of_each_ply(i)
%     abd(1:3,1:3)=abd(1:3,1:3)+Q_transformed*((ords1(1,i+1))-
ords1(1,i));
%     abd(4:end,1:3)=abd(4:end,1:3)+0.5*Q_transformed*((ords1(1,i
+1))^2-ords1(1,i)^2);
%     abd(1:3,4:6)=abd(1:3,4:6)+0.5*Q_transformed*((ords1(1,i+1))^2-
ords1(1,i)^2);
%     abd(4:6,4:6)=abd(4:6,4:6)+(1/3)*Q_transformed*((ords1(1,i+1))^3-
ords1(1,i)^3);
Q_transformed_1=Q_transformed_1+Q_transformed;
end
% z_k=0;
% z_k_2=0;
% for i=1:length(theta_of_each_ply)
% z_k=z_k+(ords1(1,i+1)-ords1(1,i));
% z_k_2=z_k_2+0.5*((ords1(1,i+1))^2-(ords1(1,i))^2);
% end
% complaince=[1 0 0 1 0 0;0 1 0 0 1 0;0 0 1 0 0 1]*inv(abd)*[z_k
 0 0;0 z_k 0;0 0 z_k;(((z_k_2)^2)/2) 0 0;0 (((z_k_2)^2)/2) 0;0 0
 (((z_k_2)^2)/2)];
stiffness=inv(Q_transformed_1);
end
```

```
Not enough input arguments.

Error in abd_matrix (line 12)
for i=1:length(theta_of_each_ply)
```

Published with MATLAB® R2020a

```
function stiffness=abd_matrix_trans(theta_of_each_ply,~)
%theta_of_each_ply=[0 90 90 0];
%DISCLAIMNER 45 DEGREE OR -45 DEGREE LEADS TO SINGULARITY OR ITS
 FACTORS IN
%Q2 Q3 OR Q4
%thickness_of_each_ply=ones(1,length(theta_of_each_ply));
% height_of_lamina=sum(thickness_of_each_ply);
% ords1=zeros(1,(length(thickness_of_each_ply)+1));
% ords1(1,1)=-height_of_lamina/2;
% for i=2:(length(thickness_of_each_ply))
%     ords1(1,i)=ords1(1,i-1)+thickness_of_each_ply(1,i);
% end
% ords1(1,end)=abs(ords1(1,1));
Q_transformed_1=zeros(2,2);
for i=1:length(theta_of_each_ply)

 Q_transformed=inv(ques_3_with_arguments_transformed_trans_plane(theta_of_each_ply
%     abd(1:2,1:2)=abd(1:2,1:2)+Q_transformed*((ords1(1,i+1))-
ords1(1,i));
%     abd(3:end,1:2)=abd(3:end,1:2)+0.5*Q_transformed*((ords1(1,i
+1))^2-ords1(1,i)^2);
%     abd(1:2,3:4)=abd(1:2,3:4)+0.5*Q_transformed*((ords1(1,i+1))^2-
ords1(1,i)^2);
%     abd(3:4,3:4)=abd(3:4,3:4)+(1/3)*Q_transformed*((ords1(1,i+1))^3-
ords1(1,i)^3);
Q_transformed_1=Q_transformed_1+Q_transformed;
end
% z_k=0;
% z_k_2=0;
% for i=1:length(theta_of_each_ply)
% z_k=z_k+(ords1(1,i+1)-ords1(1,i));
% z_k_2=z_k_2+0.5*((ords1(1,i+1))^2-(ords1(1,i))^2);
% end
% complaince=[1 0 1 0;0 1 0 1]*inv(abd)*[z_k 0;0 z_k;(z_k_2) 0;0
 z_k_2];
stiffness=inv(Q_transformed_1);
end
```

```
Not enough input arguments.

Error in abd_matrix_trans (line 14)
for i=1:length(theta_of_each_ply)
```

Published with MATLAB® R2020a

```
function
 kg_in_plane=B_matrix_in_plane(gauss_points_x,gauss_points_y,sweep_angle,b,cr,I,J,
%gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
%gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
% weights_x=[1 1];
% weights_y=[1 1];
%sweep_angle=30;%degree
%b=10;
%l_1=1;
%c=1;
%h=1;
%h_1=1;
pi=3.1416;
sweep_angle_in_radians=sweep_angle*pi/180;
%cr=5;
tip_chord=cr-((b/2)*tan(sweep_angle_in_radians));
%Linear nodes per element
% I=5;
% J=5;
% K=5;
% L=5;
% M=5;
% N=5;
% P=5;
% Q=5;
% R=5;
% S=5;
% dof=(5);
% icon_IJ=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_KL=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_PQ=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
% icon_RS=Assignment_1_Anirudh_K_ME22MTECH14017_case_3_question_2(2);
coeff_of_z_lower=Lower_fitting_a_curve();
coeff_of_z_upper=Upper_fitting_a_curve();
number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
%number_of_elements=number_of_nodes-1;
P_non_derive_I_J_x=zeros(1,I*J);
P_derive_I_J_x=zeros(1,I*J);
P_derive_K_L_x=zeros(1,K*L);
P_non_derive_K_L_x=zeros(1,K*L);
P_derive_M_N_x=zeros(1,M*N);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_derive_I_J_y=zeros(1,I*J);
```

```
P_derive_K_L_y=zeros(1,K*L);
P_non_derive_K_L_y=zeros(1,K*L);
P_derive_M_N_y=zeros(1,M*N);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_derive_R_S_y=zeros(1,R*S);
B=zeros(12,(I*J+P*Q+M*N+K*L+R*S));
% theta=[0 90 90 0];
% thickness_of_each_ply=0.01*ones(1,length(theta));
hope=zeros((I*J+P*Q+M*N+K*L+R*S),(I*J+P*Q+M*N+K*L+R*S));
spar_position_I_J=linspace(-1,1,I*J);
spar_position_K_L=linspace(-1,1,K*L);
spar_position_P_Q=linspace(-1,1,P*Q);
spar_position_M_N=linspace(-1,1,M*N);
spar_position_R_S=linspace(-1,1,R*S);
%kg=zeros(((I*J)+(K*L)+(P*Q)+(R*S))*dof,
((I*J)+(K*L)+(P*Q)+(R*S))*dof);
for
 i=1:I*J%*length(icon_KL(:,1))*length(icon_KL(1,:))*length(icon_PQ(:,1))*length(ic
%     n11_IJ=icon_IJ(i,1);
%     n22_IJ=icon_IJ(i,2);
%     n33_IJ=icon_IJ(i,3);
%     n44_IJ=icon_IJ(i,4);
%     %n55_IJ=icon_IJ(i,5);
%     n11_KL=icon_KL(i,1);
%     n22_KL=icon_KL(i,2);
%     n33_KL=icon_KL(i,3);
%     n44_KL=icon_KL(i,4);
%     %n55_KL=icon_KL(i,5);
%     n11_PQ=icon_PQ(i,1);
%     n22_PQ=icon_PQ(i,2);
%     n33_PQ=icon_PQ(i,3);
%     n44_PQ=icon_PQ(i,4);
%     %n55_PQ=icon_PQ(i,5);
%     n11_RS=icon_RS(i,1);
%     n22_RS=icon_RS(i,2);
%     n33_RS=icon_RS(i,3);
%     n44_RS=icon_RS(i,4);
    %n55_RS=icon_RS(i,5);
    for op=1:length(gauss_points_y)
        c=0.5*cr*(1-
gauss_points_y(op))+0.5*tip_chord*(1+gauss_points_y(op));
    for j=1:length(gauss_points_x)
P_non_derive_I_J_x(1,1)=1;
P_non_derive_I_J_x(1,2)=((l_1/
c)*gauss_points_x(j)+spar_position_I_J(1));
P_derive_I_J_x(1,1)=0;
P_derive_I_J_x(1,2)=1;
P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=((l_1/
c)*gauss_points_x(j)+spar_position_K_L(1));
P_derive_M_N_x(1,1)=0;
```

```
P_derive_M_N_x(1,2)=1;
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_derive_M_N_x(1,1)=0;
P_derive_M_N_x(1,2)=1;
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=((l_1/
c)*gauss_points_x(j)+spar_position_P_Q(1));
P_derive_P_Q_x(1,1)=0;
P_derive_P_Q_x(1,2)=1;
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=((l_1/
c)*gauss_points_x(j)+spar_position_R_S(1));
P_derive_R_S_x(1,1)=0;
P_derive_R_S_x(1,2)=1;
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=((l_1/
c)*gauss_points_y(op)+spar_position_I_J(1));
P_derive_I_J_y(1,1)=0;
P_derive_I_J_y(1,2)=1;
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=((l_1/
c)*gauss_points_y(op)+spar_position_K_L(1));
P_derive_M_N_y(1,1)=0;
P_derive_M_N_y(1,2)=1;
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_derive_M_N_y(1,1)=0;
P_derive_M_N_y(1,2)=1;
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=((l_1/
c)*gauss_points_y(op)+spar_position_P_Q(1));
P_derive_P_Q_y(1,1)=0;
P_derive_P_Q_y(1,2)=1;
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=((l_1/
c)*gauss_points_y(op)+spar_position_R_S(1));
P_derive_R_S_y(1,1)=0;
P_derive_R_S_y(1,2)=1;
        for k=3:number_of_nodes_I_J
           P_derive_I_J_x(1,k)=((((2*k+1)/(k+1))*((l_1/
c)*gauss_points_x(j)+spar_position_I_J(1,k-1)))*P_derive_I_J_x(1,k-1)-
(k/(k+1)*P_derive_I_J_x(1,k-2)));
           P_non_derive_I_J_x(1,k)=((((2*k+1)/(k+1))*((l_1/
c)*gauss_points_x(j)+spar_position_I_J(1,k-1)))*P_non_derive_I_J_x(1,k-1)-
(k/(k+1)*P_non_derive_I_J_x(1,k-2)));
        end
        for k=3:number_of_nodes_K_L
           P_derive_K_L_x(1,k)=((((2*k+1)/(k+1))*((l_1/
c)*gauss_points_x(j)+spar_position_K_L(1,k-1)))*P_derive_K_L_x(1,k-1)-
(k/(k+1)*P_derive_K_L_x(1,k-2)));
           P_non_derive_K_L_x(1,k)=((((2*k+1)/(k+1))*((l_1/
c)*gauss_points_x(j)+spar_position_K_L(1,k-1)))*P_non_derive_K_L_x(1,k-1)-
(k/(k+1)*P_non_derive_K_L_x(1,k-2)));
```

```
        end
%         for k=3:number_of_nodes_M_N
%            P_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_M_N_x(1,k-1)-(k/(k
+1)*P_derive_M_N_x(1,k-2)));
%            P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
%         end
        for k=3:number_of_nodes_P_Q
           P_derive_P_Q_x(1,k)=((((2*k+1)/(k+1))*((l_1/
c)*gauss_points_x(j)+spar_position_P_Q(1,k-1)))*P_derive_P_Q_x(1,k-1)-
(k/(k+1)*P_derive_P_Q_x(1,k-2)));
           P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k+1))*((l_1/
c)*gauss_points_x(j)+spar_position_P_Q(1,k-1)))*P_non_derive_P_Q_x(1,k-1)-
(k/(k+1)*P_non_derive_P_Q_x(1,k-2)));
        end
        for k=3:number_of_nodes_R_S
           P_derive_R_S_x(1,k)=((((2*k+1)/(k+1))*(l_1/
c)*(gauss_points_x(j)+spar_position_R_S(1,k-1)))*P_derive_R_S_x(1,k-1)-
(k/(k+1)*P_derive_R_S_x(1,k-2)));
           P_non_derive_R_S_x(1,k)=((((2*k+1)/(k+1))*((l_1/
c)*gauss_points_x(j)+spar_position_R_S(1,k-1)))*P_non_derive_R_S_x(1,k-1)-
(k/(k+1)*P_non_derive_R_S_x(1,k-2)));
        end
        for k=3:number_of_nodes_I_J
           P_derive_I_J_y(1,k)=((((2*k+1)/(k+1))*((l_1/
c)*gauss_points_y(op)+spar_position_I_J(1,k-1)))*P_derive_I_J_y(1,k-1)-
(k/(k+1)*P_derive_I_J_y(1,k-2)));
           P_non_derive_I_J_y(1,k)=((((2*k+1)/(k+1))*((l_1/
c)*gauss_points_y(op)+spar_position_I_J(1,k-1)))*P_non_derive_I_J_y(1,k-1)-
(k/(k+1)*P_non_derive_I_J_y(1,k-2)));
        end
        for k=3:number_of_nodes_K_L
           P_derive_K_L_y(1,k)=((((2*k+1)/(k+1))*((l_1/
c)*gauss_points_y(op)+spar_position_K_L(1,k-1)))*P_derive_K_L_x(1,k-1)-
(k/(k+1)*P_derive_K_L_x(1,k-2)));
           P_non_derive_K_L_y(1,k)=((((2*k+1)/(k+1))*((l_1/
c)*gauss_points_y(op)+spar_position_K_L(1,k-1)))*P_non_derive_K_L_y(1,k-1)-
(k/(k+1)*P_non_derive_K_L_y(1,k-2)));
        end
%         for k=3:number_of_nodes_M_N
%            P_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(j))*P_derive_M_N_y(1,k-1)-(k/(k
+1)*P_derive_M_N_y(1,k-2)));
%            P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
%         end
        for k=3:number_of_nodes_P_Q
           P_derive_P_Q_y(1,k)=((((2*k+1)/(k+1))*((l_1/
c)*gauss_points_y(op)+spar_position_P_Q(1,k-1)))*P_derive_P_Q_y(1,k-1)-
(k/(k+1)*P_derive_P_Q_y(1,k-2)));
```

```
            P_non_derive_P_Q_y(1,k)=((((2*k+1)/(k+1))*((l_1/
c)*gauss_points_y(op)+spar_position_P_Q(1,k-1)))*P_non_derive_P_Q_y(1,k-1)-
(k/(k+1)*P_non_derive_P_Q_y(1,k-2)));
        end
        for k=3:number_of_nodes_R_S
            P_derive_R_S_y(1,k)=((((2*k+1)/(k+1))*((l_1/
c)*gauss_points_y(op)+spar_position_R_S(1,k-1)))*P_derive_R_S_y(1,k-1)-
(k/(k+1)*P_derive_R_S_y(1,k-2)));
            P_non_derive_R_S_y(1,k)=((((2*k+1)/(k+1))*((l_1/
c)*gauss_points_y(op)+spar_position_R_S(1,k-1)))*P_non_derive_R_S_y(1,k-1)-
(k/(k+1)*P_non_derive_R_S_y(1,k-2)));
        end

 [detJacobian,inverse]=coordinate_transformation_Jacobian(sweep_angle,cr,b,gauss_p
        inverse_trans=inverse';
        final_IJ_x=P_derive_I_J_x.*P_non_derive_I_J_x;
        final_KL_x=P_derive_K_L_x.*P_non_derive_K_L_x;
       % final_MN_x=P_derive_M_N_x.*P_non_derive_M_N_x;
        final_PQ_x=P_derive_P_Q_x.*P_non_derive_P_Q_x;
        final_RS_x=P_derive_R_S_x.*P_non_derive_R_S_x;
        final_IJ_y=P_derive_I_J_y.*P_non_derive_I_J_y;
        final_KL_y=P_derive_K_L_y.*P_non_derive_K_L_y;
        %final_MN_y=P_derive_M_N_y.*P_non_derive_M_N_y;
        final_PQ_y=P_derive_P_Q_y.*P_non_derive_P_Q_y;
        final_RS_y=P_derive_R_S_y.*P_non_derive_R_S_y;
        boom=5;
        idk=1;
%         final_IJ=[final_IJ_x;final_IJ_y];
%         final_KL=[final_KL_x;final_KL_y];
%         final_MN=[final_MN_x;final_MN_y];
%         final_PQ=[final_PQ_x;final_PQ_y];
%         final_RS=[final_RS_x;final_RS_y];
        for pi=1:length(final_IJ_x)
            B(1:12,idk:boom)=B(1:12,idk:boom)+[final_IJ_x(1,pi)
 0 0 0 0;final_IJ_y(1,pi) 0 0 0 0;0 final_KL_x(1,pi) 0 0 0;0
 final_KL_y(1,pi) 0 0 0;0 0 0 0 0;0 0 0 0 0;0 0 0 final_PQ_x(1,pi)
 0;0 0 0 final_PQ_y(1,pi) 0;0 0 0 0 final_RS_x(1,pi);0 0 0 0
 final_RS_y(1,pi);0 0 0 0 0;0 0 0 0 0];
            idk=idk+5;
            boom=boom+5;
        end

 D=D_matrix_lower_airfoil_in_plane(sweep_angle,cr,b,theta,thickness_of_each_ply,ga
        hope=hope+(l_1/
c)*B'*D*B*weights_x(j)*weights_y(op)*detJacobian;
    end
    end
%     iv_0=[5*n11_IJ-4 5*n11_IJ-3 5*n11_IJ-2 5*n11_IJ-1 5*n11_IJ
 5*n22_IJ-4 5*n22_IJ-3 5*n22_IJ-2 5*n22_IJ-1 5*n22_IJ 5*n33_IJ-4
 5*n33_IJ-3 5*n33_IJ-2 5*n33_IJ-1 5*n33_IJ 5*n44_IJ-4 5*n44_IJ-3
 5*n44_IJ-2 5*n44_IJ-1 5*n44_IJ];
%     iv_1=[5*n11_KL-4 5*n11_KL-3 5*n11_KL-2 5*n11_KL-1 5*n11_KL
 5*n22_KL-4 5*n22_KL-3 5*n22_KL-2 5*n22_KL-1 5*n22_KL 5*n33_KL-4
```

```
 5*n33_KL-3 5*n33_KL-2 5*n33_KL-1 5*n33_KL 5*n44_KL-4 5*n44_KL-3
 5*n44_KL-2 5*n44_KL-1 5*n44_KL];
%     iv_2=[5*n11_PQ-4 5*n11_PQ-3 5*n11_PQ-2 5*n11_PQ-1 5*n11_PQ
 5*n22_PQ-4 5*n22_PQ-3 5*n22_PQ-2 5*n22_PQ-1 5*n22_PQ 5*n33_PQ-4
 5*n33_PQ-3 5*n33_PQ-2 5*n33_PQ-1 5*n33_PQ 5*n44_PQ-4 5*n44_PQ-3
 5*n44_PQ-2 5*n44_PQ-1 5*n44_PQ];
%     iv_3=[5*n11_RS-4 5*n11_RS-3 5*n11_RS-2 5*n11_RS-1 5*n11_RS
 5*n22_RS-4 5*n22_RS-3 5*n22_RS-2 5*n22_RS-1 5*n22_RS 5*n33_RS-4
 5*n33_RS-3 5*n33_RS-2 5*n33_RS-1 5*n33_RS 5*n44_RS-4 5*n44_RS-3
 5*n44_RS-2 5*n44_RS-1 5*n44_RS];
%     iv=[iv_0 iv_1 iv_2 iv_3];
    %kg(iv,iv)=kg(iv,iv)+hope;
end
kg_in_plane=hope;
end
```

```
Not enough input arguments.

Error in B_matrix_in_plane_rib_cap (line 13)
sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 D_matrix_1=D_matrix_lower_airfoil_in_plane(sweep,cr,b,theta,thickness_of_each_ply
% coeff_of_z_lower=Lower_fitting_a_curve();
% coeff_of_z_upper=Upper_fitting_a_curve();
% sweep=30;
% cr=10;
% b=20;
% h=1;
thickness_lower=0.01;
Z=zeros(5,12);
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
z_2=0;
z_derive_1=0;
z_derive_2=0;
z_derive_3=0;
n=10;
z_lower=[0.000000  0.000000;
  0.002000 -0.009300;
  0.005000 -0.016000;
  0.010000 -0.022100;
  0.020000 -0.029500;
  0.030000 -0.034400;
  0.040000 -0.038100;
  0.050000 -0.041200;
  0.070000 -0.046200;
  0.100000 -0.051700;
  0.120000 -0.054700;
  0.150000 -0.058500;
  0.170000 -0.060600;
  0.200000 -0.063300;
  0.220000 -0.064700;
  0.250000 -0.066600;
  0.280000 -0.068000;
  0.300000 -0.068700;
  0.320000 -0.069200;
  0.350000 -0.069600;
  0.370000 -0.069600;
  0.400000 -0.069200;
  0.420000 -0.068800;
  0.450000 -0.067600;
  0.480000 -0.065700;
  0.500000 -0.064400;
  0.530000 -0.061400;
  0.550000 -0.058800;
  0.580000 -0.054300;
  0.600000 -0.050900;
  0.630000 -0.045100;
  0.650000 -0.041000;
  0.680000 -0.034600;
  0.700000 -0.030200;
```

```
  0.730000 -0.023500;
  0.750000 -0.019200;
  0.770000 -0.015000;
  0.800000 -0.009300;
  0.830000 -0.004800;
  0.850000 -0.002400;
  0.870000 -0.001300;
  0.890000 -0.000800;
  0.920000 -0.001600;
  0.940000 -0.003500;
  0.950000 -0.004900;
  0.960000 -0.006600;
  0.970000 -0.008500;
  0.980000 -0.010900;
  0.990000 -0.013700;
  1.000000 -0.016300];
x=z_lower(1:end,1);
z_lower_1=z_lower(1:end,1);
for po=1:length(coeff_of_z_lower)

 z_derive_1=z_derive_1+(coeff_of_z_lower(po)*(((factorial(length(coeff_of_z_lower)
(factorial(length(coeff_of_z_lower)-
po)*factorial(po))))*po*(length(coeff_of_z_lower)-
po)*((z_lower_1(po))^(po-1)))*((z_lower_1(po))^0.75)*((1-
z_lower(po))^0.75));

 z_derive_2=z_derive_2+(coeff_of_z_lower(po))*(factorial(length(coeff_of_z_lower))
((factorial(length(coeff_of_z_lower)-
po)*factorial(po))))*(0.75)*(z_lower_1(po))^(0.75-1)*(1-
z_lower_1(po))^0.75;

 z_derive_3=z_derive_3+(coeff_of_z_lower(po)*(((factorial(length(coeff_of_z_lower)
(factorial(length(coeff_of_z_lower))*(0.75*(1-
z_lower_1(po))^(0.75-1)*(z_lower_1(po))^(0.75)))))));
end
z_derive=z_derive_1+z_derive_2+z_derive_3;
y=atan(z_derive);
y=(y*180/pi);
thickness_lower_1=thickness_lower*sqrt(1+(tan(y)^2));
%step_size=thickness_lower_1/n;
%for k=1:length(coeff_of_z_lower)
% theta=[0 90 90 0];
%  thickness_of_each_ply=0.1*(ones(1,4));
z_1=(((x).^0.75).*(1-x).^(0.75));
for k=1:length(x)
z_2=z_2+(factorial(length(coeff_of_z_lower))/
(factorial(length(coeff_of_z_lower)-
k)*factorial(k)))*coeff_of_z_lower*(((x).^k).*(ones(1,length(x))-
x).^(length(coeff_of_z_lower)-k))';
end
z_1=z_1';
z=z_1+z_2;
z_upper=z+ones(1,length(x))*(thickness_lower_1/2)+ones(1,length(x))*h;
z_l_lower=z-(ones(1,length(x))*(thickness_lower_1/2));
```

```
h=(z_upper-z_l_lower)./n;
integral_final=letsintegrate(z_upper,z_l_lower,sweep,cr,b,gauss_points_x,gauss_poi
for i=1:length(n)
    k=z_l_lower+i*h;
    for lo=1:length(z_l_lower)
    if rem(k(lo),3)==0
        integral_final=integral_final
+2*(function_1(k(lo),sweep,cr,b,gauss_points_x,gauss_points_y));
    else
        integral_final=integral_final
+3*(function_1(k(lo),sweep,cr,b,gauss_points_x,gauss_points_y));
    end
    end
    integral_final=integral_final*3*h(1)/8;
end
[stiffness]=abd_matrix(theta,thickness_of_each_ply);
D_matrix_1=integral_final'*stiffness*integral_final;
function func=function_1(k,sweep,cr,b,gauss_points_x,gauss_points_y)
% sweep=30;
% cr=10;
% b=20;
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
inverse_trans=inverse';
func=[(inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0
 0 0 k*(inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0
 0 0;0 0 0 (inverse_trans(1,2)+inverse_trans(2,1)) 0
 0 0 0 0 (k)*(inverse_trans(1,2)+inverse_trans(2,2))
 0 0;0 (inverse_trans(1,2)+inverse_trans(2,1))
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0
 0 k*(inverse_trans(2,2)+inverse_trans(2,1))
 k*(inverse_trans(1,1)+inverse_trans(1,2)) 0 0 0];
end
function
 integral=letsintegrate(z_pp,z_l_lower,sweep,cr,b,gauss_points_x,gauss_points_y)
% sweep=30;
% cr=10*10^-3;
% b=20*10^-3;
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
inverse_trans=inverse';
integration_1=zeros(3,12);
integration_2=zeros(3,12);
for i=1:length(z_pp)
integration_1=integration_1+[(inverse_trans(1,1)+inverse_trans(2,1))
 0 0 0 0 0 z_pp(i)*(inverse_trans(1,1)+inverse_trans(2,1))
 0 0 0 0 0;0 0 0 (inverse_trans(1,2)+inverse_trans(2,1)) 0
 0 0 0 0 (z_pp(i))*(inverse_trans(1,2)+inverse_trans(2,2))
 0 0;0 (inverse_trans(1,2)+inverse_trans(2,1))
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0 0
 z_pp(i)*(inverse_trans(2,2)+inverse_trans(2,1))
 z_pp(i)*(inverse_trans(1,1)+inverse_trans(1,2)) 0 0 0];
```

```
integration_2=integration_2+[(inverse_trans(1,1)+inverse_trans(2,1))
 0 0 0 0 0 z_l_lower(i)*(inverse_trans(1,1)+inverse_trans(2,1))
 0 0 0 0 0;0 0 0 (inverse_trans(1,2)+inverse_trans(2,1)) 0 0
 0 0 0 (z_l_lower(i))*(inverse_trans(1,2)+inverse_trans(2,2))
 0 0;0 (inverse_trans(1,2)+inverse_trans(2,1))
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0 0
 z_l_lower(i)*(inverse_trans(2,2)+inverse_trans(2,1))
 z_l_lower(i)*(inverse_trans(1,1)+inverse_trans(1,2)) 0 0 0];
end
integral=integration_1+integration_2;
end
end
```

```
Not enough input arguments.

Error in D_matrix_lower_airfoil_in_plane (line 12)
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
```

Published with MATLAB® R2020a

```
function
 D_matrix_2=D_matrix_upper_airfoil_in_plane(sweep,cr,b,theta,thickness_of_each_ply
% coeff_of_z_lower=Lower_fitting_a_curve();
% coeff_of_z_upper=Upper_fitting_a_curve();
% sweep=30;
% cr=10*10^-3;
% b=20*10^-3;
% h_1=1;
gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
thickness_lower=0.01;
Z=zeros(5,12);
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
z_2=0;
z_derive_1=0;
z_derive_2=0;
z_derive_3=0;
n=10;
z_uppe_1=[0.000000  0.000000;
  0.002000  0.009500;
  0.005000  0.015800;
  0.010000  0.021900;
  0.020000  0.029300;
  0.030000  0.034300;
  0.040000  0.038100;
  0.050000  0.041100;
  0.070000  0.046200;
  0.100000  0.051800;
  0.120000  0.054800;
  0.150000  0.058500;
  0.170000  0.060600;
  0.200000  0.063200;
  0.220000  0.064600;
  0.250000  0.066400;
  0.270000  0.067300;
  0.300000  0.068500;
  0.330000  0.069200;
  0.350000  0.069600;
  0.380000  0.069800;
  0.400000  0.069700;
  0.430000  0.069500;
  0.450000  0.069200;
  0.480000  0.068400;
  0.500000  0.067800;
  0.530000  0.066600;
  0.550000  0.065600;
  0.570000  0.064500;
  0.600000  0.062500;
  0.620000  0.061000;
  0.650000  0.058500;
  0.680000  0.055500;
  0.700000  0.053300;
```

```
  0.720000  0.050900;
  0.750000  0.046900;
  0.770000  0.043900;
  0.800000  0.038900;
  0.820000  0.035300;
  0.850000  0.029400;
  0.870000  0.025100;
  0.900000  0.018100;
  0.920000  0.013100;
  0.950000  0.004900;
  0.970000 -0.000900;
  0.980000 -0.003900;
  0.990000 -0.007100;
  1.000000 -0.010400;
  ];
x=z_uppe_1(1:end,1);
z_lower_actual=z_uppe_1(1:end,1);
for po=1:length(coeff_of_z_upper)

 z_derive_1=z_derive_1+(coeff_of_z_upper(po)*(((factorial(length(coeff_of_z_upper)
(factorial(length(coeff_of_z_upper)-
po)*factorial(po))))*po*(length(coeff_of_z_upper)-
po)*((z_lower_actual(po))^(po-1)))*((z_lower_actual(po))^0.75)*((1-
z_lower_actual(po))^0.75));

 z_derive_2=z_derive_2+(coeff_of_z_upper(po))*(factorial(length(coeff_of_z_upper))
((factorial(length(coeff_of_z_upper)-
po)*factorial(po))))*(0.75)*(z_lower_actual(po))^(0.75-1)*(1-
z_lower_actual(po))^0.75;

 z_derive_3=z_derive_3+(coeff_of_z_upper(po)*(((factorial(length(coeff_of_z_upper)
(factorial(length(coeff_of_z_upper))*(0.75*(1-
z_lower_actual(po))^(0.75-1)*(z_lower_actual(po))^(0.75)))))));
end
z_derive=z_derive_1+z_derive_2+z_derive_3;
y=atan(z_derive);
y=(y*180/pi);
thickness_lower_1=thickness_lower*sqrt(1+(tan(y))^2);
%step_size=thickness_lower_1/n;
%for k=1:length(coeff_of_z_lower)
% theta=[0 90 90 0];
% thickness_of_each_ply=0.1*(ones(1,4));
z_1=(((x).^0.75).*(1-x).^(0.75));
for k=1:length(x)
z_2=z_2+(factorial(length(coeff_of_z_upper))/
(factorial(length(coeff_of_z_upper)-
k)*factorial(k)))*coeff_of_z_upper*(((x).^k).*(ones(1,length(x))-
x).^(length(coeff_of_z_upper)-k))';
end
z_1=z_1';
z=z_1+z_2;
z_upper=z-ones(1,length(x))*(thickness_lower_1/2);
z_l_lower=z-(ones(1,length(x))*(thickness_lower_1/2))-
(ones(1,length(x))*h_1);
```

```
h=(z_upper-z_l_lower)./n;
integral_final=letsintegrate(z_upper,z_l_lower,sweep,cr,b,gauss_points_x,gauss_poi
for i=1:length(n)
    k=z_l_lower+i*h;
    for lo=1:length(z_l_lower)
    if rem(k(lo),3)==0
        integral_final=integral_final
+2*(function_1(k(lo),sweep,cr,b));
    else
        integral_final=integral_final
+3*(function_1(k(lo),sweep,cr,b));
    end
    end
    integral_final=integral_final*3*h(1)/8;
end
[stiffness]=abd_matrix(theta,thickness_of_each_ply);
D_matrix_2=integral_final'*stiffness*integral_final;
function func=function_1(k,sweep,cr,b)
% sweep=30;
% cr=10;
% b=20;
gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
inverse_trans=inverse';
func=[(inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0
 0 0 k*(inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0
 0 0;0 0 0 (inverse_trans(1,2)+inverse_trans(2,1)) 0
 0 0 0 0 (k)*(inverse_trans(1,2)+inverse_trans(2,2))
 0 0;0 (inverse_trans(1,2)+inverse_trans(2,1))
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0
 0 k*(inverse_trans(2,2)+inverse_trans(2,1))
 k*(inverse_trans(1,1)+inverse_trans(1,2)) 0 0 0];
end
function
 integral=letsintegrate(z_pp,z_l_lower,sweep,cr,b,gauss_points_x,gauss_points_y)
% sweep=30;
% cr=10*10^-3;
% b=20*10^-3;
gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
inverse_trans=inverse';
integration_1=zeros(3,12);
integration_2=zeros(3,12);
for i=1:length(z_pp)
integration_1=integration_1+[(inverse_trans(1,1)+inverse_trans(2,1))
 0 0 0 0 0 z_pp(i)*(inverse_trans(1,1)+inverse_trans(2,1))
 0 0 0 0 0;0 0 0 (inverse_trans(1,2)+inverse_trans(2,1)) 0
 0 0 0 0 (z_pp(i))*(inverse_trans(1,2)+inverse_trans(2,2))
 0 0;0 (inverse_trans(1,2)+inverse_trans(2,1))
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0 0
 z_pp(i)*(inverse_trans(2,2)+inverse_trans(2,1))
 z_pp(i)*(inverse_trans(1,1)+inverse_trans(1,2)) 0 0 0];
```

```
integration_2=integration_2+[(inverse_trans(1,1)+inverse_trans(2,1))
 0 0 0 0 0 z_l_lower(i)*(inverse_trans(1,1)+inverse_trans(2,1))
 0 0 0 0 0;0 0 0 (inverse_trans(1,2)+inverse_trans(2,1)) 0 0
 0 0 0 (z_l_lower(i))*(inverse_trans(1,2)+inverse_trans(2,2))
 0 0;0 (inverse_trans(1,2)+inverse_trans(2,1))
 (inverse_trans(1,1)+inverse_trans(2,1)) 0 0 0 0
 z_l_lower(i)*(inverse_trans(2,2)+inverse_trans(2,1))
 z_l_lower(i)*(inverse_trans(1,1)+inverse_trans(1,2)) 0 0 0];
end
integral=integration_1+integration_2;
end
end
```

```
Not enough input arguments.

Error in D_matrix_upper_airfoil_in_plane (line 12)
[detJacobian,inverse]=coordinate_transformation_Jacobian(sweep,cr,b,gauss_points_x
```

Published with MATLAB® R2020a

```
function
 Stiffness_rib_cap=FEM_Main_file_ribs_cap(gauss_points_in_x,gauss_points_in_y,gaus
% gauss_points_in_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_trans_x=0;
% gauss_points_in_y=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_trans_y=0;
% weights_in_x=[1 1];
% weights_in_y=[1 1];
% weights_trans_x=2;
% weights_trans_y=2;
% sweep=30;%degree
% b=10;
% cr=5;
% I=4;
% J=4;
% K=4;
% L=4;
% M=4;
% N=4;
% P=4;
% Q=4;
% R=4;
% S=4;
% l_1=1;
% %c=1;
% h=1;
% h_2=1;
B_in_plane=B_matrix_in_plane_rib_cap(gauss_points_in_x,gauss_points_in_y,sweep,b,c
B_trans_plane=B_matrix_trans_plane_rib_cap(gauss_points_trans_x,gauss_points_trans
Stiffness_rib_cap=B_in_plane+B_trans_plane;
end
```

```
Not enough input arguments.

Error in FEM_Main_file_ribs_cap (line 27)
B_in_plane=B_matrix_in_plane_rib_cap(gauss_points_in_x,gauss_points_in_y,sweep,b,c
```

Published with MATLAB® R2020a

```
function curve_fit_Lower=Lower_fitting_a_curve()
z_lower=[0.000000  0.000000;
  0.002000 -0.009300;
  0.005000 -0.016000;
  0.010000 -0.022100;
  0.020000 -0.029500;
  0.030000 -0.034400;
  0.040000 -0.038100;
  0.050000 -0.041200;
  0.070000 -0.046200;
  0.100000 -0.051700;
  0.120000 -0.054700;
  0.150000 -0.058500;
  0.170000 -0.060600;
  0.200000 -0.063300;
  0.220000 -0.064700;
  0.250000 -0.066600;
  0.280000 -0.068000;
  0.300000 -0.068700;
  0.320000 -0.069200;
  0.350000 -0.069600;
  0.370000 -0.069600;
  0.400000 -0.069200;
  0.420000 -0.068800;
  0.450000 -0.067600;
  0.480000 -0.065700;
  0.500000 -0.064400;
  0.530000 -0.061400;
  0.550000 -0.058800;
  0.580000 -0.054300;
  0.600000 -0.050900;
  0.630000 -0.045100;
  0.650000 -0.041000;
  0.680000 -0.034600;
  0.700000 -0.030200;
  0.730000 -0.023500;
  0.750000 -0.019200;
  0.770000 -0.015000;
  0.800000 -0.009300;
  0.830000 -0.004800;
  0.850000 -0.002400;
  0.870000 -0.001300;
  0.890000 -0.000800;
  0.920000 -0.001600;
  0.940000 -0.003500;
  0.950000 -0.004900;
  0.960000 -0.006600;
  0.970000 -0.008500;
  0.980000 -0.010900;
  0.990000 -0.013700;
  1.000000 -0.016300];
x_guess=ones(1,length(z_lower));
```

```
airfoil_equation_lower_curve_2=@airfoil_equation_lower_curve;
curve_fit_Lower=lsqcurvefit(airfoil_equation_lower_curve_2,x_guess,z_lower(1:end,1
z_predict=z_lower(1:end,1)';
thickness_ratio_lower_surface=0.01;
S_predict=zeros(length(z_lower),length(z_lower));
for i=1:length(z_predict)
   S_predict(i,1:length(z_predict))=(factorial(length(z_lower))/
(factorial(length(z_lower)-
i)*factorial(i)))*(((z_predict).^i).*((ones(1,length(z_predict))-
z_predict).^(length(z_predict)-i)));
end
% predicited_graph=((z_predict).^0.75).*((ones(1,length(z_predict))-
z_predict).^0.75).*(curve_fit_Lower*S_predict)+ones(1,length(z_predict))*thickness
% plot(z_lower(1:end,1)',predicited_graph,"b")
% hold on
% plot(z_lower(1:end,1),z_lower(1:end,2),"bo")
% hold off
function z=airfoil_equation_lower_curve(x,x_data)
x_data=x_data';
thickness_ratio_lower_surface=0.01;
S=zeros(length(x_data),length(x_data));
for i=1:length(x_data)
   S(i,1:length(x_data))=(factorial(length(x_data))/
(factorial(length(x_data)-
i)*factorial(i)))*(((x_data).^i).*((ones(1,length(x_data))-
x_data).^(length(x_data)-i)));
end
matrix=zeros(1,length(x_data));
for i=1:length(x_data)
    matrix(1,i)=x(i);
end
z=((x_data).^0.75).*((ones(1,length(x_data))-
x_data).^0.75).*(matrix*S)+ones(1,length(x_data))*thickness_ratio_lower_surface;
end
end
```

```
Local minimum found.

Optimization completed because the size of the gradient is less than
the value of the optimality tolerance.


ans =

  Columns 1 through 7

   -6.8476   16.4317  -26.3682   19.0586    1.1947  -12.2233   -0.0915

  Columns 8 through 14

    7.5818    1.1945   -5.4205   -3.4790    2.0104    3.4490    0.2375

  Columns 15 through 21
```

```
  -2.8556   -2.4605    0.2447    1.9473    1.0842   -0.9910   -1.9690

Columns 22 through 28

  -1.0438    0.5741    1.1811    0.3165   -0.9685   -1.3421   -0.5335

Columns 29 through 35

   0.5210    0.7394   -0.0024   -0.8314   -0.8664   -0.1462    0.4839

Columns 36 through 42

   0.3620   -0.2620   -0.5966   -0.3027    0.1513    0.2203    0.0051

Columns 43 through 49

  -0.1396   -0.2629   -0.2347    0.4094    0.2315   -1.3475    1.2399

Column 50

  -1.7220
```

Published with MATLAB® R2020a

```
function C=ques_1_compliance()
% prompt="Enter the value of E";
% prompt_1="Enter the value of poisson's ratio";
% prompt_2="Enter the value of G";
E1=70.7;
E2=70.7;
%E3=input(prompt);
poisson_ratio_12=0.1;
%poisson_ratio_13=input(prompt_1);
%poisson_ratio_23=input(prompt_1);
G12=5;
%G13=input(prompt_2);
%G23=input(prompt_2);
S=zeros(3,3);
S(1,1)=1/E1;
S(1,2)=-poisson_ratio_12/E2;
%S(1,3)=poisson_ratio_13/E3;
S(2,1)=-poisson_ratio_12/E1;
S(2,2)=1/E2;
%S(2,3)=poisson_ratio_23/E3;
%S(3,1)=poisson_ratio_13/E1;
%S(3,2)=poisson_ratio_23/E2;
%S(3,3)=1/E3;
%S(4,4)=1/G23;
%S(5,5)=1/G13;
S(3,3)=1/G12;
C=inv(S);
end
```

```
ans =

   71.4141    7.1414         0
    7.1414   71.4141         0
         0         0    5.0000
```

Published with MATLAB® R2020a

```
function Q=ques_2_Q_matrix_in_plane()
Stiffness=ques_1_compliance_in_plane();
Q=zeros(3,3);
Q(1,1)=Stiffness(1,1);%-(Stiffness(1,3)^2/Stiffness(3,3));
Q(1,2)=Stiffness(1,2);%-((Stiffness(1,3)*Stiffness(2,3))/
Stiffness(3,3));
Q(2,2)=Stiffness(2,2);%-(Stiffness(2,3)^2/Stiffness(3,3));
Q(3,3)=Stiffness(3,3);
Q(2,1)=Q(1,2);
end
```

```
ans =

   71.4141    7.1414         0
    7.1414   71.4141         0
         0         0    5.0000
```

Published with MATLAB® R2020a

```
function Q=ques_2_Q_matrix_trans_plane()
Stiffness=ques_1_compliance_trans_plane();
Q=Stiffness;
% Q(1,1)=Stiffness(1,1);%-(Stiffness(1,3)^2/Stiffness(3,3));
% Q(1,2)=Stiffness(1,2);%-((Stiffness(1,3)*Stiffness(2,3))/
Stiffness(3,3));
% Q(2,2)=Stiffness(2,2);%-(Stiffness(2,3)^2/Stiffness(3,3));
% Q(3,3)=Stiffness(3,3);
% Q(2,1)=Q(1,2);
end
```

```
ans =

     5     0
     0     5
```

Published with MATLAB® R2020a

```
function C=ques_3_with_arguments_transformed(theta)
%prompt="Enter the value of theta in degree";
%theta=input(prompt);
theta=(pi*theta)/180;
C=zeros(3,3);
C(1,1)=cos(theta)^2;
C(1,2)=sin(theta)^2;
C(1,3)=2*cos(theta)*sin(theta);
C(2,1)=sin(theta)^2;
C(2,2)=cos(theta)^2;
C(2,3)=-2*cos(theta)*sin(theta);
C(3,1)=-cos(theta)*sin(theta);
C(3,2)=cos(theta)*sin(theta);
C(3,3)=cos(theta)^2-sin(theta)^2;
end
```

```
Not enough input arguments.

Error in ques_3_with_arguments_transformed_in_plane (line 4)
theta=(pi*theta)/180;
```

Published with MATLAB® R2020a

```
function C=ques_3_with_arguments_transformed_trans_plane(theta)
%prompt="Enter the value of theta in degree";
%theta=input(prompt);
%theta=30;
theta=(pi*theta)/180;
C=zeros(2,2);
C(1,1)=cos(theta);
C(1,2)=-sin(theta);
%C(1,3)=2*cos(theta)*sin(theta);
C(2,1)=-sin(theta);
C(2,2)=cos(theta);
% C(2,3)=-2*cos(theta)*sin(theta);
% C(3,1)=-cos(theta)*sin(theta);
% C(3,2)=cos(theta)*sin(theta);
% C(3,3)=cos(theta)^2-sin(theta)^2;
end
```

```
Not enough input arguments.

Error in ques_3_with_arguments_transformed_trans_plane (line 5)
theta=(pi*theta)/180;
```

Published with MATLAB® R2020a

```
function curve_fit_upper=Upper_fitting_a_curve()
z_upper=[0.000000  0.000000;
  0.002000  0.009500;
  0.005000  0.015800;
  0.010000  0.021900;
  0.020000  0.029300;
  0.030000  0.034300;
  0.040000  0.038100;
  0.050000  0.041100;
  0.070000  0.046200;
  0.100000  0.051800;
  0.120000  0.054800;
  0.150000  0.058500;
  0.170000  0.060600;
  0.200000  0.063200;
  0.220000  0.064600;
  0.250000  0.066400;
  0.270000  0.067300;
  0.300000  0.068500;
  0.330000  0.069200;
  0.350000  0.069600;
  0.380000  0.069800;
  0.400000  0.069700;
  0.430000  0.069500;
  0.450000  0.069200;
  0.480000  0.068400;
  0.500000  0.067800;
  0.530000  0.066600;
  0.550000  0.065600;
  0.570000  0.064500;
  0.600000  0.062500;
  0.620000  0.061000;
  0.650000  0.058500;
  0.680000  0.055500;
  0.700000  0.053300;
  0.720000  0.050900;
  0.750000  0.046900;
  0.770000  0.043900;
  0.800000  0.038900;
  0.820000  0.035300;
  0.850000  0.029400;
  0.870000  0.025100;
  0.900000  0.018100;
  0.920000  0.013100;
  0.950000  0.004900;
  0.970000 -0.000900;
  0.980000 -0.003900;
  0.990000 -0.007100;
  1.000000 -0.010400;
  ];
x_guess=ones(1,length(z_upper));
airfoil_equation_lower_curve_2=@airfoil_equation_lower_curve;
```

```
curve_fit_upper=lsqcurvefit(airfoil_equation_lower_curve_2,x_guess,z_upper(1:end,1
z_predict=z_upper(1:end,1)';
thickness_ratio_lower_surface=0.01;
S_predict=zeros(length(z_upper),length(z_upper));
for i=1:length(z_predict)
   S_predict(i,1:length(z_predict))=(factorial(length(z_upper))/
(factorial(length(z_upper)-
i)*factorial(i)))*(((z_predict).^i).*((ones(1,length(z_predict))-
z_predict).^(length(z_predict)-i)));
end
% predicited_graph=((z_predict).^0.75).*((ones(1,length(z_predict))-
z_predict).^0.75).*(curve_fit_upper*S_predict)+ones(1,length(z_predict))*thickness
% plot(z_upper(1:end,1)',predicited_graph,"b")
% hold on
% plot(z_upper(1:end,1),z_upper(1:end,2),"bo")
% hold off
function z=airfoil_equation_lower_curve(x,x_data)
x_data=x_data';
thickness_ratio_lower_surface=0.01;
S=zeros(length(x_data),length(x_data));
for i=1:length(x_data)
   S(i,1:length(x_data))=(factorial(length(x_data))/
(factorial(length(x_data)-
i)*factorial(i)))*(((x_data).^i).*((ones(1,length(x_data))-
x_data).^(length(x_data)-i)));
end
matrix=zeros(1,length(x_data));
for i=1:length(x_data)
    matrix(1,i)=x(i);
end
z=((x_data).^0.75).*((ones(1,length(x_data))-
x_data).^0.75).*(matrix*S)+ones(1,length(x_data))*thickness_ratio_lower_surface;
end
end
```

Local minimum found.

Optimization completed because the size of the gradient is less than the value of the optimality tolerance.

ans =

Columns 1 through 7

1.4012 -1.1876 1.2572 0.4493 -0.2935 0.0879 0.5228

Columns 8 through 14

0.4273 0.1085 -0.0065 0.1345 0.3086 0.3318 0.2166

Columns 15 through 21

```
    0.0980    0.0795    0.1532    0.2356    0.2548    0.2051    0.1366

  Columns 22 through 28

    0.1040    0.1243    0.1720    0.2059    0.2026    0.1689    0.1313

  Columns 29 through 35

    0.1141    0.1246    0.1522    0.1777    0.1829    0.1583    0.1120

  Columns 36 through 42

    0.0756    0.0893    0.1579    0.2111    0.1484   -0.0132   -0.0575

  Columns 43 through 48

    0.1703    0.3062   -0.1698   -0.3775    0.6231   -1.1358
```

Published with MATLAB® R2020a

5.6 Force Vector

```
function
 fg=force_vector(sweep_angle,root_of_chord,span_of_the_entire_wing,I,J,K,L,M,N,P,Q

% sweep_angle=30;%degree
% span_of_the_entire_wing=4;
%sweep_angle_in_radians=sweep_angle*pi/180;
% root_of_chord=5;
%tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
% area=0.5*(span_of_the_entire_wing/2)*(root_of_chord+tip_chord);
% aspect=span_of_the_entire_wing/area;
%  coordinates_left_top_corner=[0 root_of_chord];
%  coordinates_left_bottom_corner=[0 0];
%  coordinates_right_bottom_corner=[span_of_the_entire_wing/2 0];
%  coordinates_right_top_corner=[span_of_the_entire_wing/2 tip_chord];
%  I=5;
%  J=5;
%  K=5;
%  L=5;
%  M=5;
%  N=5;
%  P=5;
%  Q=5;
%  R=5;
%  S=5;
 number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
% weights_x=[1 1];
% weights_y=[1 1];

Not enough input arguments.

Error in force_vector (line 23)
 number_of_nodes_I_J=((I*J));
```

Transformed coordinates

```
%lagrange shape functions
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_z=[-1/sqrt(3) 1/sqrt(3)];
% coordinate_trans_x=zeros(1,length(coordinates_x));
% coordinate_trans_y=zeros(1,length(coordinates_y));
% for i=1:length(gauss_points_x)
%     for j=1:length(gauss_points_y)
%         B_1=0.25*(1-gauss_points_y(j))*(1-gauss_points_x(i));
%         B_2=0.25*(1+gauss_points_y(j))*(1-gauss_points_x(i));
%         B_3=0.25*(1+gauss_points_y(j))*(1+gauss_points_x(i));
%         B_4=0.25*(1-gauss_points_y(j))*(1+gauss_points_x(i));
```

```
%         B_matrix=[B_1 B_2 B_3 B_4];
%         for op=1:length(x)
%
 coordinate_trans_x(1,op)=(B_1+B_1+B_1+B_1)*coordinates_x(1,op);
%
 coordinate_trans_y(1,op)=(B_1+B_1+B_1+B_1)*coordinates_y(1,op);
%         end
%     end
% end
%hope=zeros((I*J+K*L+M*N+P*Q+R*S),1);
P_non_derive_I_J_x=zeros(1,I*J);
P_derive_I_J_x=zeros(1,I*J);
P_derive_K_L_x=zeros(1,K*L);
P_non_derive_K_L_x=zeros(1,K*L);
P_derive_M_N_x=zeros(1,M*N);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_derive_I_J_y=zeros(1,I*J);
P_derive_K_L_y=zeros(1,K*L);
P_non_derive_K_L_y=zeros(1,K*L);
P_derive_M_N_y=zeros(1,M*N);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_derive_R_S_y=zeros(1,R*S);
hope_1=zeros(5,(I*J+K*L+M*N+P*Q+R*S));
hope_local=zeros((I*J+K*L+M*N+P*Q+R*S),1);
%mn=1:1:I*J;
for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
        for ki=1:length(gauss_points_z)
P_non_derive_I_J_x(1,1)=1;
P_non_derive_I_J_x(1,2)=gauss_points_x(j);
P_derive_I_J_x(1,1)=0;
P_derive_I_J_x(1,2)=1;
P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=gauss_points_x(j);
P_derive_M_N_x(1,1)=0;
P_derive_M_N_x(1,2)=1;
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_derive_M_N_x(1,1)=0;
P_derive_M_N_x(1,2)=1;
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=gauss_points_x(j);
P_derive_P_Q_x(1,1)=0;
P_derive_P_Q_x(1,2)=1;
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=gauss_points_x(j);
```

```
P_derive_R_S_x(1,1)=0;
P_derive_R_S_x(1,2)=1;
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=gauss_points_y(op);
P_derive_I_J_y(1,1)=0;
P_derive_I_J_y(1,2)=1;
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=gauss_points_y(op);
P_derive_M_N_y(1,1)=0;
P_derive_M_N_y(1,2)=1;
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_derive_M_N_y(1,1)=0;
P_derive_M_N_y(1,2)=1;
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_derive_P_Q_y(1,1)=0;
P_derive_P_Q_y(1,2)=1;
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=gauss_points_y(op);
P_derive_R_S_y(1,1)=0;
P_derive_R_S_y(1,2)=1;
[detJacob,inerse]=coordinate_transformation_Jacobian(sweep_angle,root_of_chord,spa
        for k=3:number_of_nodes_I_J
           P_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_I_J_x(1,k-1)-(k/(k
+1)*P_derive_I_J_x(1,k-2)));
           P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2)));
        end
        for k=3:number_of_nodes_K_L
           P_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_K_L_x(1,k-1)-(k/(k
+1)*P_derive_K_L_x(1,k-2)));
           P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
        end
        for k=3:number_of_nodes_M_N
            P_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_M_N_x(1,k-1)-(k/(k
+1)*P_derive_M_N_x(1,k-2)));
            P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
         end
        for k=3:number_of_nodes_P_Q
           P_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_derive_P_Q_x(1,k-2)));
           P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
```

```
        end
        for k=3:number_of_nodes_R_S
           P_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(j))*P_derive_R_S_x(1,k-1)-(k/(k
+1)*P_derive_R_S_x(1,k-2)));
           P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
        end
        for k=3:number_of_nodes_I_J
           P_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(j))*P_derive_I_J_y(1,k-1)-(k/(k
+1)*P_derive_I_J_y(1,k-2)));
           P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
        end
        for k=3:number_of_nodes_K_L
           P_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(j))*P_derive_K_L_x(1,k-1)-(k/(k
+1)*P_derive_K_L_x(1,k-2)));
           P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
        end
         for k=3:number_of_nodes_M_N
            P_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_derive_M_N_y(1,k-1)-(k/(k
+1)*P_derive_M_N_y(1,k-2)));
            P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
         end
        for k=3:number_of_nodes_P_Q
           P_derive_P_Q_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(j))*P_derive_P_Q_y(1,k-1)-(k/(k
+1)*P_derive_P_Q_y(1,k-2)));
           P_non_derive_P_Q_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_P_Q_y(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y(1,k-2)));
        end
        for k=3:number_of_nodes_R_S
           P_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(j))*P_derive_R_S_y(1,k-1)-(k/(k
+1)*P_derive_R_S_y(1,k-2)));
           P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
        end
        final_IJ=P_non_derive_I_J_x.*P_non_derive_I_J_y;
        final_KL=P_non_derive_K_L_x.*P_non_derive_K_L_y;
        final_MN=P_non_derive_M_N_x.*P_non_derive_M_N_y;
        final_PQ=P_non_derive_P_Q_x.*P_non_derive_P_Q_y;
        final_RS=P_non_derive_R_S_x.*P_non_derive_R_S_y;
```

```
        opq=1;
        pew=1;
        omega=1;
        yt=1;
        Ltd=10;
        while opq<=(I*J+K*L+M*N+P*Q+R*S)
            final=[final_IJ(1,yt) final_KL(1,yt) final_MN(1,yt)
 final_PQ(1,yt) final_RS(1,yt)];
            hope_1(pew,opq)=final(omega);
            yt=yt+1;
            if rem(opq,I*J)==0
                pew=pew+1;
                yt=1;
                omega=omega+1;
            end
                opq=opq+1;
        end
        %hope(u,I*J)=[final_IJ(u) zeros(1,length(mn));zeros(1,m_n)
 final_IJ(u) zeros(1,length)]

  %force_local=[0;0;1;gauss_points_z(ki)*0;gauss_points_z(ki)*0];
        hope_local=hope_local
+hope_1'*force_local*detJacob*weights_x(j)*weights_y(op);
    end

end
end
fg=hope_local;

end
```

Published with MATLAB® R2020a

```
function
 [det_jacobian,Jacobian_inverse]=coordinate_transformation_Jacobian(sweep_angle,ro
%sweep_angle=30;%degree
%span_of_the_entire_wing=10;
sweep_angle_in_radians=sweep_angle*pi/180;
%root_of_chord=5;
tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
% coordinates_left_top_corner=[0 root_of_chord];
% coordinates_left_bottom_corner=[0 0];
% coordinates_right_bottom_corner=[span_of_the_entire_wing/2 0];
% coordinates_right_top_corner=[span_of_the_entire_wing/2 tip_chord];
coordinates_x=[0 root_of_chord root_of_chord root_of_chord-
tip_chord]';
coordinates_y=[0 0 span_of_the_entire_wing/2
 span_of_the_entire_wing/2]';
% gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
% coordinates_transformation_along_x=0;
% coordinates_transformation_along_y=0;
coordinates=[coordinates_x coordinates_y];
for i=1:length(gauss_points_x)
    for j=1:length(gauss_points_y)
%N_1=0.25*(1-gauss_points_x(i))*(1-gauss_points_y(j));
%N_2=0.25*(1+gauss_points_x(i))*(1-gauss_points_y(j));
%N_3=0.25*(1+gauss_points_x(i))*(1+gauss_points_y(j));
%N_4=0.25*(1-gauss_points_x(i))*(1+gauss_points_y(j));
B_1_x=-0.25*(1-gauss_points_y(j));
B_1_y=-0.25*(1-gauss_points_x(i));
B_2_x=0.25*(1-gauss_points_y(j));
B_2_y=-0.25*(1+gauss_points_x(i));
B_3_x=0.25*(1+gauss_points_x(i));
B_3_y=0.25*(1+gauss_points_y(j));
B_4_x=-0.25*(1+gauss_points_y(j));
B_4_y=0.25*(1+gauss_points_x(i));
Jacobian=[B_1_x B_2_x B_3_x B_4_x;B_1_y B_2_y B_3_y
 B_4_y]*coordinates;
    end
end
det_jacobian=det(Jacobian);
Jacobian_inverse=inv(Jacobian);
end
```

```
Not enough input arguments.

Error in coordinate_transformation_Jacobian (line 4)
sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

5.7 Artificial Stiffness

```
function
 K_artificial_global=Artificial_Stiffness(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span_of_
%   I=4;
%  J=4;
%  K=4;
%  L=4;
%   M=4;
%    N=4;
%   P=4;
%   Q=4;
%    R=4;
%    S=4;
%    sweep_angle=30;%degree
%   span_of_the_entire_wing=4;
%   root_of_chord=5;
  %tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
  % gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
%   gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%   weights_x=[1 1];
%   weights_y=[1 1];
% gauss_points_trans_x=0;
% gauss_points_trans_y=0;
% weights_trans_x=2;
% weights_trans_y=2;
  dummy_1=zeros(I*J,I*J);
  dummy_2=zeros(K*L,K*L);
  dummy_3=zeros(M*N,M*N);
  dummy_4=zeros(P*Q,P*Q);
K_artificial_global_1=Artificial_stiffness_K_11(I,J,K,L,M,N,P,Q,R,S,sweep_angle,sp
K_artificial_global_2=Artificial_stiffness_K_22(I,J,K,L,M,N,P,Q,R,S,sweep_angle,sp
K_artificial_global_3=Artificial_stiffness_K_33(I,J,K,L,M,N,P,Q,R,S,sweep_angle,sp
K_artificial_global_4=Artificial_stiffness_K_44(I,J,K,L,M,N,P,Q,R,S,sweep_angle,sp
K_artificial_global_5=Artificial_stiffness_K_55(I,J,K,L,M,N,P,Q,R,S,sweep_angle,sp
K_artificial_global=[K_artificial_global_1 dummy_1 dummy_2 dummy_3
 dummy_4;dummy_1 K_artificial_global_2 dummy_2 dummy_3 dummy_4;dummy_1
 dummy_2 K_artificial_global_3 dummy_3 dummy_4;dummy_1 dummy_2 dummy_3
 K_artificial_global_4 dummy_4;dummy_1 dummy_2 dummy_3 dummy_4
 K_artificial_global_5];
end
```

```
Not enough input arguments.

Error in Artificial_Stiffness (line 24)
  dummy_1=zeros(I*J,I*J);
```

Published with MATLAB® R2020a

```
function
 K_11=Artificial_stiffness_K_11(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span_of_the_entire
%  I=4;
%  J=4;
%  K=4;
%   L=4;
%  M=4;
%   N=4;
%  P=4;
%  Q=4;
%   R=4;
%   S=4;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
%  root_of_chord=5;
%  %tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%   gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
%  gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%   weights_x=[1 1];
%  weights_y=[1 1];
%  gauss_points_trans_x=0;
%  gauss_points_trans_y=0;
%  weights_trans_x=2;
%  weights_trans_y=2;
K_11_1=Artificial_stiffness_leading_edge_K_11(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span
K_11_2=Artificial_stiffness_root_K_11(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span_of_the_
K_11_3=Artificial_stiffness_tip_K_11(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span_of_the_e
K_11_4=Artificial_stiffness_trailing_edge_K_11(I,J,K,L,M,N,P,Q,R,S,sweep_angle,spa
K_11=K_11_1+K_11_2+K_11_3+K_11_4;
end
```

```
Not enough input arguments.

Error in Artificial_stiffness_K_11 (line 24)
K_11_1=Artificial_stiffness_leading_edge_K_11(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span
```

Published with MATLAB® R2020a

```
function
 K_11_alpha_le_trans=Artificial_stiffness_leading_edge(I,J,K,L,M,N,P,Q,R,S,sweep_a
%  I=5;
% J=5;
% K=5;
%  L=5;
%  M=5;
%  N=5;
%  P=5;
%  Q=5;
%  R=5;
%  S=5;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
  sweep_angle_in_radians=sweep_angle*pi/180;
%  root_of_chord=5;
  tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%  gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%  weights_x=[1 1];
% weights_y=[1 1];
 u_o=zeros(I,1);
 P_non_derive_I_J_x=zeros(1,I*J);
P_non_derive_K_L_x=zeros(1,K*L);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_non_derive_K_L_y=zeros(1,K*L);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_non_derive_I_J_x_le=zeros(1,I*J);
P_non_derive_K_L_x_le=zeros(1,K*L);
P_non_derive_M_N_x_le=zeros(1,M*N);
P_non_derive_P_Q_x_le=zeros(1,P*Q);
P_non_derive_R_S_x_le=zeros(1,R*S);
P_non_derive_I_J_y_le=zeros(1,I*J);
P_non_derive_K_L_y_le=zeros(1,K*L);
P_non_derive_M_N_y_le=zeros(1,M*N);
P_non_derive_P_Q_y_le=zeros(1,P*Q);
 P_non_derive_R_S_y_le=zeros(1,R*S);
number_of_nodes_I_J_le_x=I*J;
number_of_nodes_K_L_le_x=K*L;
number_of_nodes_M_N_le_x=M*N;
number_of_nodes_P_Q_le_x=P*Q;
number_of_nodes_R_S_le_x=R*S;
number_of_nodes_I_J_le_y=I*J;
number_of_nodes_K_L_le_y=K*L;
number_of_nodes_M_N_le_y=M*N;
number_of_nodes_R_S_le_y=R*S;
```

```
 number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
 number_of_nodes_I_J_le=((I*J));
number_of_nodes_K_L_le=((K*L));
number_of_nodes_M_N_le=((M*N));
number_of_nodes_P_Q_le=((P*Q));
number_of_nodes_R_S_le=((R*S));
coordinates_x=[0 root_of_chord root_of_chord root_of_chord-
tip_chord]';
coordinates_y=[0 0 span_of_the_entire_wing/2
 span_of_the_entire_wing/2]';
x_3=coordinates_x(3);
y_3=coordinates_y(3);
x_4=coordinates_x(4);
y_4=coordinates_y(4);
K_11_alpha_le_trans=zeros(1,(I*J));
for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
        P_non_derive_I_J_x(1,1)=1;
        P_non_derive_I_J_x(1,2)=(gauss_points_x(j));
        P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=(gauss_points_x(j));
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=(gauss_points_y(op));
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=(gauss_points_y(op));
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=(gauss_points_y(op));
P_non_derive_I_J_x_le(1,1)=1;
P_non_derive_I_J_x_le(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y_le(1,1)=1;
P_non_derive_I_J_y_le(1,2)=(gauss_points_y(op));
 P_non_derive_K_L_x_le(1,1)=1;
P_non_derive_K_L_x_le(1,2)=(gauss_points_x(j));
P_non_derive_K_L_y_le(1,1)=1;
P_non_derive_K_L_y_le(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x_le(1,1)=1;
P_non_derive_M_N_x_le(1,2)=gauss_points_x(j);
P_non_derive_M_N_y_le(1,1)=1;
P_non_derive_M_N_y_le(1,2)=gauss_points_y(op);
P_non_derive_P_Q_x_le(1,1)=1;
```

```
P_non_derive_P_Q_x_le(1,2)=(gauss_points_x(j));
P_non_derive_P_Q_y_le(1,1)=1;
P_non_derive_P_Q_y_le(1,2)=gauss_points_y(op);
P_non_derive_R_S_x_le(1,1)=1;
P_non_derive_R_S_x_le(1,2)=(gauss_points_y(op));
P_non_derive_R_S_y_le(1,1)=1;
P_non_derive_R_S_y_le(1,2)=(gauss_points_y(op));
for k=3:number_of_nodes_I_J
    P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2))));
end
for k=3:number_of_nodes_K_L
    P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
end
for k=3:number_of_nodes_M_N
     P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
end
 for k=3:number_of_nodes_P_Q
      P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
 end
  for k=3:number_of_nodes_R_S
       P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
  end
   for k=3:number_of_nodes_I_J
       P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
   end
   for k=3:number_of_nodes_K_L
       P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
   end
    for k=3:number_of_nodes_M_N
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
     for k=3:number_of_nodes_R_S
         P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
     end
     for k=3:number_of_nodes_I_J_le_x
```

```
    P_non_derive_I_J_x_le(1,k)=((((2*k+1)/(k
+1))*(-1)*P_non_derive_I_J_x_le(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x_le(1,k-2))));
end
for k=3:number_of_nodes_K_L_le_x
    P_non_derive_K_L_x_le(1,k)=((((2*k+1)/(k
+1))*(-1))*P_non_derive_K_L_x_le(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x_le(1,k-2)));
end
for k=3:number_of_nodes_M_N_le_x
     P_non_derive_M_N_x_le(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_M_N_x_le(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x_le(1,k-2)));
end
 for k=3:number_of_nodes_P_Q_le_x
      P_non_derive_P_Q_x_le(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_P_Q_x_le(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x_le(1,k-2)));
 end
  for k=3:number_of_nodes_R_S_le_x
       P_non_derive_R_S_x_le(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_R_S_x_le(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x_le(1,k-2)));
  end
   for k=3:number_of_nodes_I_J_le_y
       P_non_derive_I_J_y_le(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y_le(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y_le(1,k-2)));
   end
   for k=3:number_of_nodes_K_L_le_y
       P_non_derive_K_L_y_le(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y_le(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y_le(1,k-2)));
   end
    for k=3:number_of_nodes_M_N_le_y
        P_non_derive_M_N_y_le(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y_le(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y_le(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y_le(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_P_Q_y_le(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y_le(1,k-2)));
    end
     for k=3:number_of_nodes_R_S_le_y
         P_non_derive_R_S_y_le(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y_le(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y_le(1,k-2)));
     end
     final_IJ=P_non_derive_I_J_x.*P_non_derive_I_J_y;
        final_KL=P_non_derive_K_L_x.*P_non_derive_K_L_y;
        final_MN=P_non_derive_M_N_x.*P_non_derive_M_N_y;
        final_PQ=P_non_derive_P_Q_x.*P_non_derive_P_Q_y;
        final_RS=P_non_derive_R_S_x.*P_non_derive_R_S_y;
```

```
        final_IJ_le=P_non_derive_I_J_x_le.*P_non_derive_I_J_y_le;
        final_KL_le=P_non_derive_K_L_x_le.*P_non_derive_K_L_y_le;
        final_MN_le=P_non_derive_M_N_x_le.*P_non_derive_M_N_y_le;
        final_PQ_le=P_non_derive_P_Q_x_le.*P_non_derive_P_Q_y_le;
        final_RS_le= P_non_derive_R_S_x_le.*P_non_derive_R_S_y_le;
        K_11_alpha_le_trans=K_11_alpha_le_trans
+(sin(sweep_angle_in_radians)*(x_3-
x_4)+cos(sweep_angle_in_radians)*((y_3-
y_4)/4))*(final_IJ_le.*final_KL_le.*final_PQ_le.*final_MN_le.*final_RS_le)'*(final
    end
end
end
```

Not enough input arguments.

Error in Artificial_stiffness_leading_edge_K_11 (line 14)
* sweep_angle_in_radians=sweep_angle*pi/180;*

Published with MATLAB® R2020a

```
function
 K_11_alpha_le_trans=Artificial_stiffness_leading_edge(I,J,K,L,M,N,P,Q,R,S,sweep_a
%  I=5;
% J=5;
% K=5;
%  L=5;
%  M=5;
%  N=5;
%  P=5;
%  Q=5;
%  R=5;
%  S=5;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
  sweep_angle_in_radians=sweep_angle*pi/180;
%  root_of_chord=5;
  tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%  gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%  weights_x=[1 1];
% weights_y=[1 1];
 u_o=zeros(I,1);
 P_non_derive_I_J_x=zeros(1,I*J);
P_non_derive_K_L_x=zeros(1,K*L);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_non_derive_K_L_y=zeros(1,K*L);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_non_derive_I_J_x_le=zeros(1,I*J);
P_non_derive_K_L_x_le=zeros(1,K*L);
P_non_derive_M_N_x_le=zeros(1,M*N);
P_non_derive_P_Q_x_le=zeros(1,P*Q);
P_non_derive_R_S_x_le=zeros(1,R*S);
P_non_derive_I_J_y_le=zeros(1,I*J);
P_non_derive_K_L_y_le=zeros(1,K*L);
P_non_derive_M_N_y_le=zeros(1,M*N);
P_non_derive_P_Q_y_le=zeros(1,P*Q);
 P_non_derive_R_S_y_le=zeros(1,R*S);
number_of_nodes_I_J_le_x=I*J;
number_of_nodes_K_L_le_x=K*L;
number_of_nodes_M_N_le_x=M*N;
number_of_nodes_P_Q_le_x=P*Q;
number_of_nodes_R_S_le_x=R*S;
number_of_nodes_I_J_le_y=I*J;
number_of_nodes_K_L_le_y=K*L;
number_of_nodes_M_N_le_y=M*N;
number_of_nodes_R_S_le_y=R*S;
```

```
 number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
 number_of_nodes_I_J_le=((I*J));
number_of_nodes_K_L_le=((K*L));
number_of_nodes_M_N_le=((M*N));
number_of_nodes_P_Q_le=((P*Q));
number_of_nodes_R_S_le=((R*S));
coordinates_x=[0 root_of_chord root_of_chord root_of_chord-
tip_chord]';
coordinates_y=[0 0 span_of_the_entire_wing/2
 span_of_the_entire_wing/2]';
x_3=coordinates_x(3);
y_3=coordinates_y(3);
x_4=coordinates_x(4);
y_4=coordinates_y(4);
K_11_alpha_le_trans=zeros(1,(I*J));
for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
        P_non_derive_I_J_x(1,1)=1;
        P_non_derive_I_J_x(1,2)=(gauss_points_x(j));
        P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=(gauss_points_x(j));
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=(gauss_points_y(op));
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=(gauss_points_y(op));
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=(gauss_points_y(op));
P_non_derive_I_J_x_le(1,1)=1;
P_non_derive_I_J_x_le(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y_le(1,1)=1;
P_non_derive_I_J_y_le(1,2)=(gauss_points_y(op));
 P_non_derive_K_L_x_le(1,1)=1;
P_non_derive_K_L_x_le(1,2)=(gauss_points_x(j));
P_non_derive_K_L_y_le(1,1)=1;
P_non_derive_K_L_y_le(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x_le(1,1)=1;
P_non_derive_M_N_x_le(1,2)=gauss_points_x(j);
P_non_derive_M_N_y_le(1,1)=1;
P_non_derive_M_N_y_le(1,2)=gauss_points_y(op);
P_non_derive_P_Q_x_le(1,1)=1;
```

```
P_non_derive_P_Q_x_le(1,2)=(gauss_points_x(j));
P_non_derive_P_Q_y_le(1,1)=1;
P_non_derive_P_Q_y_le(1,2)=gauss_points_y(op);
P_non_derive_R_S_x_le(1,1)=1;
P_non_derive_R_S_x_le(1,2)=(gauss_points_y(op));
P_non_derive_R_S_y_le(1,1)=1;
P_non_derive_R_S_y_le(1,2)=(gauss_points_y(op));
for k=3:number_of_nodes_I_J
    P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2))));
end
for k=3:number_of_nodes_K_L
    P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
end
for k=3:number_of_nodes_M_N
     P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
end
 for k=3:number_of_nodes_P_Q
      P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
 end
  for k=3:number_of_nodes_R_S
       P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
  end
   for k=3:number_of_nodes_I_J
       P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
   end
   for k=3:number_of_nodes_K_L
       P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
   end
    for k=3:number_of_nodes_M_N
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
     for k=3:number_of_nodes_R_S
         P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
     end
     for k=3:number_of_nodes_I_J_le_x
```

```
    P_non_derive_I_J_x_le(1,k)=((((2*k+1)/(k
+1))*(-1)*P_non_derive_I_J_x_le(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x_le(1,k-2))));
end
for k=3:number_of_nodes_K_L_le_x
    P_non_derive_K_L_x_le(1,k)=((((2*k+1)/(k
+1))*(-1))*P_non_derive_K_L_x_le(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x_le(1,k-2)));
end
for k=3:number_of_nodes_M_N_le_x
     P_non_derive_M_N_x_le(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_M_N_x_le(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x_le(1,k-2)));
end
 for k=3:number_of_nodes_P_Q_le_x
      P_non_derive_P_Q_x_le(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_P_Q_x_le(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x_le(1,k-2)));
 end
  for k=3:number_of_nodes_R_S_le_x
       P_non_derive_R_S_x_le(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_R_S_x_le(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x_le(1,k-2)));
  end
   for k=3:number_of_nodes_I_J_le_y
       P_non_derive_I_J_y_le(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y_le(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y_le(1,k-2)));
   end
   for k=3:number_of_nodes_K_L_le_y
       P_non_derive_K_L_y_le(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y_le(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y_le(1,k-2)));
   end
    for k=3:number_of_nodes_M_N_le_y
        P_non_derive_M_N_y_le(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y_le(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y_le(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y_le(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_P_Q_y_le(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y_le(1,k-2)));
    end
     for k=3:number_of_nodes_R_S_le_y
         P_non_derive_R_S_y_le(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y_le(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y_le(1,k-2)));
     end
     final_IJ=P_non_derive_I_J_x.*P_non_derive_I_J_y;
        final_KL=P_non_derive_K_L_x.*P_non_derive_K_L_y;
        final_MN=P_non_derive_M_N_x.*P_non_derive_M_N_y;
        final_PQ=P_non_derive_P_Q_x.*P_non_derive_P_Q_y;
        final_RS=P_non_derive_R_S_x.*P_non_derive_R_S_y;
```

```
        final_IJ_le=P_non_derive_I_J_x_le.*P_non_derive_I_J_y_le;
        final_KL_le=P_non_derive_K_L_x_le.*P_non_derive_K_L_y_le;
        final_MN_le=P_non_derive_M_N_x_le.*P_non_derive_M_N_y_le;
        final_PQ_le=P_non_derive_P_Q_x_le.*P_non_derive_P_Q_y_le;
        final_RS_le= P_non_derive_R_S_x_le.*P_non_derive_R_S_y_le;
        K_11_alpha_le_trans=K_11_alpha_le_trans
+(sin(sweep_angle_in_radians)*(x_3-
x_4)+cos(sweep_angle_in_radians)*((y_3-
y_4)/4))*(final_IJ_le.*final_KL_le.*final_PQ_le.*final_MN_le.*final_RS_le)'*(final
    end
end
end
```

Not enough input arguments.

Error in Artificial_stiffness_leading_edge_K_11 (line 14)
* sweep_angle_in_radians=sweep_angle*pi/180;*

Published with MATLAB® R2020a

```
function
 K_11_alpha_tip_trans=Artificial_stiffness_tip_K_11(I,J,K,L,M,N,P,Q,R,S,sweep_angl
%  I=2;
% J=2;
% K=2;
%  L=2;
%  M=2;
%  N=2;
%  P=2;
%  Q=2;
%  R=2;
%  S=2;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
 sweep_angle_in_radians=sweep_angle*pi/180;
%  root_of_chord=5;
 tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%  gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%  weights_x=[1 1];
% weights_y=[1 1];
 u_o=zeros(I,1);
 P_non_derive_I_J_x=zeros(1,I*J);
P_non_derive_K_L_x=zeros(1,K*L);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_non_derive_K_L_y=zeros(1,K*L);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_non_derive_I_J_x_root=zeros(1,I*J);
P_non_derive_K_L_x_root=zeros(1,K*L);
P_non_derive_M_N_x_root=zeros(1,M*N);
P_non_derive_P_Q_x_root=zeros(1,P*Q);
P_non_derive_R_S_x_root=zeros(1,R*S);
P_non_derive_I_J_y_root=zeros(1,I*J);
P_non_derive_K_L_y_root=zeros(1,K*L);
P_non_derive_M_N_y_root=zeros(1,M*N);
P_non_derive_P_Q_y_root=zeros(1,P*Q);
 P_non_derive_R_S_y_root=zeros(1,R*S);
number_of_nodes_I_J_le_x=I*J;
number_of_nodes_K_L_le_x=K*L;
number_of_nodes_M_N_le_x=M*N;
number_of_nodes_P_Q_le_x=P*Q;
number_of_nodes_R_S_le_x=R*S;
number_of_nodes_I_J_le_y=I*J;
number_of_nodes_K_L_le_y=K*L;
number_of_nodes_M_N_le_y=M*N;
number_of_nodes_R_S_le_y=R*S;
```

```
 number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
 number_of_nodes_I_J_le=((I*J));
number_of_nodes_K_L_le=((K*L));
number_of_nodes_M_N_le=((M*N));
number_of_nodes_P_Q_le=((P*Q));
number_of_nodes_R_S_le=((R*S));
coordinates_x=[0 root_of_chord root_of_chord root_of_chord-
tip_chord]';
coordinates_y=[0 0 span_of_the_entire_wing/2
 span_of_the_entire_wing/2]';
x_3=coordinates_x(3);
y_3=coordinates_y(3);
x_4=coordinates_x(4);
y_4=coordinates_y(4);
K_11_alpha_tip_trans=zeros(1,I*J);
for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
        P_non_derive_I_J_x(1,1)=1;
        P_non_derive_I_J_x(1,2)=(gauss_points_x(j));
        P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=(gauss_points_x(j));
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=(gauss_points_y(op));
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=(gauss_points_y(op));
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=(gauss_points_y(op));
P_non_derive_I_J_x_root(1,1)=1;
P_non_derive_I_J_x_root(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y_root(1,1)=1;
P_non_derive_I_J_y_root(1,2)=(gauss_points_y(op));
 P_non_derive_K_L_x_root(1,1)=1;
P_non_derive_K_L_x_root(1,2)=(gauss_points_x(j));
P_non_derive_K_L_y_root(1,1)=1;
P_non_derive_K_L_y_root(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x_root(1,1)=1;
P_non_derive_M_N_x_root(1,2)=gauss_points_x(j);
P_non_derive_M_N_y_root(1,1)=1;
P_non_derive_M_N_y_root(1,2)=gauss_points_y(op);
P_non_derive_P_Q_x_root(1,1)=1;
```

```
P_non_derive_P_Q_x_root(1,2)=(gauss_points_x(j));
P_non_derive_P_Q_y_root(1,1)=1;
P_non_derive_P_Q_y_root(1,2)=gauss_points_y(op);
P_non_derive_R_S_x_root(1,1)=1;
P_non_derive_R_S_x_root(1,2)=(gauss_points_y(op));
P_non_derive_R_S_y_root(1,1)=1;
P_non_derive_R_S_y_root(1,2)=(gauss_points_y(op));
for k=3:number_of_nodes_I_J
    P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2))));
end
for k=3:number_of_nodes_K_L
    P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
end
for k=3:number_of_nodes_M_N
     P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
end
 for k=3:number_of_nodes_P_Q
      P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
 end
  for k=3:number_of_nodes_R_S
       P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
  end
   for k=3:number_of_nodes_I_J
       P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
   end
   for k=3:number_of_nodes_K_L
       P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
   end
    for k=3:number_of_nodes_M_N
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
     for k=3:number_of_nodes_R_S
         P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
     end
     for k=3:number_of_nodes_I_J_le_x
```

```
    P_non_derive_I_J_x_root(1,k)=((((2*k+1)/(k
+1))*(1)*P_non_derive_I_J_x_root(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x_root(1,k-2))));
end
for k=3:number_of_nodes_K_L_le_x
    P_non_derive_K_L_x_root(1,k)=((((2*k+1)/(k
+1))*(1))*P_non_derive_K_L_x_root(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x_root(1,k-2)));
end
for k=3:number_of_nodes_M_N_le_x
     P_non_derive_M_N_x_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_M_N_x_root(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x_root(1,k-2)));
end
 for k=3:number_of_nodes_P_Q_le_x
      P_non_derive_P_Q_x_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_P_Q_x_root(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x_root(1,k-2)));
 end
  for k=3:number_of_nodes_R_S_le_x
       P_non_derive_R_S_x_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_R_S_x_root(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x_root(1,k-2)));
  end
   for k=3:number_of_nodes_I_J_le_y
       P_non_derive_I_J_y_root(1,k)=((((2*k+1)/
(k+1))*(1))*P_non_derive_I_J_y_root(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y_root(1,k-2)));
   end
   for k=3:number_of_nodes_K_L_le_y
       P_non_derive_K_L_y_root(1,k)=((((2*k+1)/
(k+1))*(1))*P_non_derive_K_L_y_root(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y_root(1,k-2)));
   end
    for k=3:number_of_nodes_M_N_le_y
        P_non_derive_M_N_y_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_M_N_y_root(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y_root(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_P_Q_y_root(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y_root(1,k-2)));
    end
     for k=3:number_of_nodes_R_S_le_y
         P_non_derive_R_S_y_root(1,k)=((((2*k+1)/
(k+1))*(1))*P_non_derive_R_S_y_root(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y_root(1,k-2)));
     end
     final_IJ=P_non_derive_I_J_x.*P_non_derive_I_J_y;
        final_KL=P_non_derive_K_L_x.*P_non_derive_K_L_y;
        final_MN=P_non_derive_M_N_x.*P_non_derive_M_N_y;
        final_PQ=P_non_derive_P_Q_x.*P_non_derive_P_Q_y;
        final_RS=P_non_derive_R_S_x.*P_non_derive_R_S_y;
```

```
 final_IJ_root=P_non_derive_I_J_x_root.*P_non_derive_I_J_y_root;

 final_KL_root=P_non_derive_K_L_x_root.*P_non_derive_K_L_y_root;

 final_MN_root=P_non_derive_M_N_x_root.*P_non_derive_M_N_y_root;

 final_PQ_root=P_non_derive_P_Q_x_root.*P_non_derive_P_Q_y_root;
        final_RS_root=
 P_non_derive_R_S_x_root.*P_non_derive_R_S_y_root;
        K_11_alpha_tip_trans=K_11_alpha_tip_trans+((x_3-
x_4)/4)*(final_IJ_root.*final_KL_root.*final_PQ_root.*final_MN_root.*final_RS_root
    end
end
```

```
Not enough input arguments.

Error in Artificial_stiffness_tip_K_11 (line 14)
 sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 K_11_alpha_te_trans=Artificial_stiffness_trailing_edge_K_11(I,J,K,L,M,N,P,Q,R,S,s
%  I=2;
% J=2;
% K=2;
%  L=2;
%  M=2;
%  N=2;
%  P=2;
%  Q=2;
%  R=2;
%  S=2;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
 sweep_angle_in_radians=sweep_angle*pi/180;
%  root_of_chord=5;
 tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%  gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%  weights_x=[1 1];
% weights_y=[1 1];
 u_o=zeros(I,1);
 P_non_derive_I_J_x=zeros(1,I*J);
P_non_derive_K_L_x=zeros(1,K*L);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_non_derive_K_L_y=zeros(1,K*L);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_non_derive_I_J_x_te=zeros(1,I*J);
P_non_derive_K_L_x_te=zeros(1,K*L);
P_non_derive_M_N_x_te=zeros(1,M*N);
P_non_derive_P_Q_x_te=zeros(1,P*Q);
P_non_derive_R_S_x_te=zeros(1,R*S);
P_non_derive_I_J_y_te=zeros(1,I*J);
P_non_derive_K_L_y_te=zeros(1,K*L);
P_non_derive_M_N_y_te=zeros(1,M*N);
P_non_derive_P_Q_y_te=zeros(1,P*Q);
 P_non_derive_R_S_y_te=zeros(1,R*S);
number_of_nodes_I_J_te_x=I*J;
number_of_nodes_K_L_te_x=K*L;
number_of_nodes_M_N_te_x=M*N;
number_of_nodes_P_Q_te_x=P*Q;
number_of_nodes_R_S_te_x=R*S;
number_of_nodes_I_J_te_y=I*J;
number_of_nodes_K_L_te_y=K*L;
number_of_nodes_M_N_te_y=M*N;
number_of_nodes_R_S_te_y=R*S;
```

```
 number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
 number_of_nodes_I_J_le=((I*J));
number_of_nodes_K_L_le=((K*L));
number_of_nodes_M_N_le=((M*N));
number_of_nodes_P_Q_le=((P*Q));
number_of_nodes_R_S_le=((R*S));
coordinates_x=[0 root_of_chord root_of_chord root_of_chord-
tip_chord]';
coordinates_y=[0 0 span_of_the_entire_wing/2
 span_of_the_entire_wing/2]';
x_3=coordinates_x(3);
y_3=coordinates_y(3);
x_2=coordinates_x(2);
y_2=coordinates_y(2);
K_11_alpha_te_trans=zeros(1,I*J);
for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
        P_non_derive_I_J_x(1,1)=1;
        P_non_derive_I_J_x(1,2)=(gauss_points_x(j));
        P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=(gauss_points_x(j));
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=(gauss_points_y(op));
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=(gauss_points_y(op));
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=(gauss_points_y(op));
P_non_derive_I_J_x_te(1,1)=1;
P_non_derive_I_J_x_te(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y_te(1,1)=1;
P_non_derive_I_J_y_te(1,2)=(gauss_points_y(op));
 P_non_derive_K_L_x_te(1,1)=1;
P_non_derive_K_L_x_te(1,2)=(gauss_points_x(j));
P_non_derive_K_L_y_te(1,1)=1;
P_non_derive_K_L_y_te(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x_te(1,1)=1;
P_non_derive_M_N_x_te(1,2)=gauss_points_x(j);
P_non_derive_M_N_y_te(1,1)=1;
P_non_derive_M_N_y_te(1,2)=gauss_points_y(op);
P_non_derive_P_Q_x_te(1,1)=1;
```

```
P_non_derive_P_Q_x_te(1,2)=(gauss_points_x(j));
P_non_derive_P_Q_y_te(1,1)=1;
P_non_derive_P_Q_y_te(1,2)=gauss_points_y(op);
P_non_derive_R_S_x_te(1,1)=1;
P_non_derive_R_S_x_te(1,2)=(gauss_points_y(op));
P_non_derive_R_S_y_te(1,1)=1;
P_non_derive_R_S_y_te(1,2)=(gauss_points_y(op));
for k=3:number_of_nodes_I_J
    P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2))));
end
for k=3:number_of_nodes_K_L
    P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
end
for k=3:number_of_nodes_M_N
     P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
end
 for k=3:number_of_nodes_P_Q
      P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
 end
  for k=3:number_of_nodes_R_S
       P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
  end
   for k=3:number_of_nodes_I_J
       P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
   end
   for k=3:number_of_nodes_K_L
       P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
   end
    for k=3:number_of_nodes_M_N
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
     for k=3:number_of_nodes_R_S
         P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
     end
     for k=3:number_of_nodes_I_J_te_x
```

```
    P_non_derive_I_J_x_te(1,k)=((((2*k+1)/(k
+1))*(1)*P_non_derive_I_J_x_te(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x_te(1,k-2))));
end
for k=3:number_of_nodes_K_L_te_x
    P_non_derive_K_L_x_te(1,k)=((((2*k+1)/(k
+1))*(1))*P_non_derive_K_L_x_te(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x_te(1,k-2)));
end
for k=3:number_of_nodes_M_N_te_x
     P_non_derive_M_N_x_te(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_M_N_x_te(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x_te(1,k-2)));
end
 for k=3:number_of_nodes_P_Q_te_x
      P_non_derive_P_Q_x_te(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_P_Q_x_te(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x_te(1,k-2)));
 end
  for k=3:number_of_nodes_R_S_te_x
       P_non_derive_R_S_x_te(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_R_S_x_te(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x_te(1,k-2)));
  end
   for k=3:number_of_nodes_I_J_te_y
       P_non_derive_I_J_y_te(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y_te(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y_te(1,k-2)));
   end
   for k=3:number_of_nodes_K_L_te_y
       P_non_derive_K_L_y_te(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y_te(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y_te(1,k-2)));
   end
    for k=3:number_of_nodes_M_N_te_y
        P_non_derive_M_N_y_te(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y_te(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y_te(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y_te(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_P_Q_y_te(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y_te(1,k-2)));
    end
     for k=3:number_of_nodes_R_S_te_y
         P_non_derive_R_S_y_te(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y_te(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y_te(1,k-2)));
     end
     final_IJ=P_non_derive_I_J_x.*P_non_derive_I_J_y;
        final_KL=P_non_derive_K_L_x.*P_non_derive_K_L_y;
        final_MN=P_non_derive_M_N_x.*P_non_derive_M_N_y;
        final_PQ=P_non_derive_P_Q_x.*P_non_derive_P_Q_y;
        final_RS=P_non_derive_R_S_x.*P_non_derive_R_S_y;
```

```
        final_IJ_le=P_non_derive_I_J_x_te.*P_non_derive_I_J_y_te;
        final_KL_le=P_non_derive_K_L_x_te.*P_non_derive_K_L_y_te;
        final_MN_le=P_non_derive_M_N_x_te.*P_non_derive_M_N_y_te;
        final_PQ_le=P_non_derive_P_Q_x_te.*P_non_derive_P_Q_y_te;
        final_RS_le= P_non_derive_R_S_x_te.*P_non_derive_R_S_y_te;
        K_11_alpha_te_trans=K_11_alpha_te_trans+(sin(0)*(x_3-
x_2)+cos(0)*((y_3-
y_2))/4)*(final_IJ_le.*final_KL_le.*final_PQ_le.*final_MN_le.*final_RS_le)'*(final
    end
end
```

```
Not enough input arguments.

Error in Artificial_stiffness_trailing_edge_K_11 (line 14)
 sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 K_22=Artificial_stiffness_K_11(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span_of_the_entire
%  I=4;
%  J=4;
%  K=4;
%   L=4;
%  M=4;
%   N=4;
%  P=4;
%  Q=4;
%   R=4;
%   S=4;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
%  root_of_chord=5;
%  %tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%   gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
%  gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%   weights_x=[1 1];
%  weights_y=[1 1];
% gauss_points_trans_x=0;
% gauss_points_trans_y=0;
% weights_trans_x=2;
% weights_trans_y=2;
K_11_1=Artificial_stiffness_leading_edge_K_22(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span
K_11_2=Artificial_stiffness_root_K_22(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span_of_the_
K_11_3=Artificial_stiffness_tip_K_22(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span_of_the_e
K_11_4=Artificial_stiffness_trailing_edge_K_22(I,J,K,L,M,N,P,Q,R,S,sweep_angle,spa
K_22=K_11_1+K_11_2+K_11_3+K_11_4;
end
```

```
Not enough input arguments.

Error in Artificial_stiffness_K_22 (line 24)
K_11_1=Artificial_stiffness_leading_edge_K_22(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span
```

Published with MATLAB® R2020a

```
function
 K_22_alpha_le_trans=Artificial_stiffness_leading_edge(I,J,K,L,M,N,P,Q,R,S,sweep_a
%  I=2;
% J=2;
% K=2;
%  L=2;
%  M=2;
%  N=2;
%  P=2;
%  Q=2;
%  R=2;
%  S=2;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
 sweep_angle_in_radians=sweep_angle*pi/180;
%  root_of_chord=5;
 tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%  gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%  weights_x=[1 1];
% weights_y=[1 1];
 u_o=zeros(I,1);
 P_non_derive_I_J_x=zeros(1,I*J);
P_non_derive_K_L_x=zeros(1,K*L);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_non_derive_K_L_y=zeros(1,K*L);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_non_derive_I_J_x_le=zeros(1,I*J);
P_non_derive_K_L_x_le=zeros(1,K*L);
P_non_derive_M_N_x_le=zeros(1,M*N);
P_non_derive_P_Q_x_le=zeros(1,P*Q);
P_non_derive_R_S_x_le=zeros(1,R*S);
P_non_derive_I_J_y_le=zeros(1,I*J);
P_non_derive_K_L_y_le=zeros(1,K*L);
P_non_derive_M_N_y_le=zeros(1,M*N);
P_non_derive_P_Q_y_le=zeros(1,P*Q);
 P_non_derive_R_S_y_le=zeros(1,R*S);
number_of_nodes_I_J_le_x=I*J;
number_of_nodes_K_L_le_x=K*L;
number_of_nodes_M_N_le_x=M*N;
number_of_nodes_P_Q_le_x=P*Q;
number_of_nodes_R_S_le_x=R*S;
number_of_nodes_I_J_le_y=I*J;
number_of_nodes_K_L_le_y=K*L;
number_of_nodes_M_N_le_y=M*N;
number_of_nodes_R_S_le_y=R*S;
```

```
 number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
 number_of_nodes_I_J_le=((I*J));
number_of_nodes_K_L_le=((K*L));
number_of_nodes_M_N_le=((M*N));
number_of_nodes_P_Q_le=((P*Q));
number_of_nodes_R_S_le=((R*S));
coordinates_x=[0 root_of_chord root_of_chord root_of_chord-
tip_chord]';
coordinates_y=[0 0 span_of_the_entire_wing/2
 span_of_the_entire_wing/2]';
x_3=coordinates_x(3);
y_3=coordinates_y(3);
x_4=coordinates_x(4);
y_4=coordinates_y(4);
K_22_alpha_le_trans=zeros(1,I*J);
for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
        P_non_derive_I_J_x(1,1)=1;
        P_non_derive_I_J_x(1,2)=(gauss_points_x(j));
        P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=(gauss_points_x(j));
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=(gauss_points_y(op));
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=(gauss_points_y(op));
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=(gauss_points_y(op));
P_non_derive_I_J_x_le(1,1)=1;
P_non_derive_I_J_x_le(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y_le(1,1)=1;
P_non_derive_I_J_y_le(1,2)=(gauss_points_y(op));
 P_non_derive_K_L_x_le(1,1)=1;
P_non_derive_K_L_x_le(1,2)=(gauss_points_x(j));
P_non_derive_K_L_y_le(1,1)=1;
P_non_derive_K_L_y_le(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x_le(1,1)=1;
P_non_derive_M_N_x_le(1,2)=gauss_points_x(j);
P_non_derive_M_N_y_le(1,1)=1;
P_non_derive_M_N_y_le(1,2)=gauss_points_y(op);
P_non_derive_P_Q_x_le(1,1)=1;
```

```
P_non_derive_P_Q_x_le(1,2)=(gauss_points_x(j));
P_non_derive_P_Q_y_le(1,1)=1;
P_non_derive_P_Q_y_le(1,2)=gauss_points_y(op);
P_non_derive_R_S_x_le(1,1)=1;
P_non_derive_R_S_x_le(1,2)=(gauss_points_y(op));
P_non_derive_R_S_y_le(1,1)=1;
P_non_derive_R_S_y_le(1,2)=(gauss_points_y(op));
for k=3:number_of_nodes_I_J
    P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2))));
end
for k=3:number_of_nodes_K_L
    P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
end
for k=3:number_of_nodes_M_N
     P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
end
 for k=3:number_of_nodes_P_Q
      P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
 end
  for k=3:number_of_nodes_R_S
       P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
  end
   for k=3:number_of_nodes_I_J
       P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
   end
   for k=3:number_of_nodes_K_L
       P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
   end
    for k=3:number_of_nodes_M_N
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
    for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_P_Q_y(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y(1,k-2)));
    end
     for k=3:number_of_nodes_R_S
```

```
         P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
     end
     for k=3:number_of_nodes_I_J_le_x
    P_non_derive_I_J_x_le(1,k)=((((2*k+1)/(k
+1))*(-1)*P_non_derive_I_J_x_le(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x_le(1,k-2))));
end
for k=3:number_of_nodes_K_L_le_x
    P_non_derive_K_L_x_le(1,k)=((((2*k+1)/(k
+1))*(-1))*P_non_derive_K_L_x_le(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x_le(1,k-2)));
end
for k=3:number_of_nodes_M_N_le_x
     P_non_derive_M_N_x_le(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_M_N_x_le(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x_le(1,k-2)));
end
 for k=3:number_of_nodes_P_Q_le_x
      P_non_derive_P_Q_x_le(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_P_Q_x_le(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x_le(1,k-2)));
 end
  for k=3:number_of_nodes_R_S_le_x
       P_non_derive_R_S_x_le(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_R_S_x_le(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x_le(1,k-2)));
  end
   for k=3:number_of_nodes_I_J_le_y
       P_non_derive_I_J_y_le(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y_le(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y_le(1,k-2)));
   end
   for k=3:number_of_nodes_K_L_le_y
       P_non_derive_K_L_y_le(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y_le(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y_le(1,k-2)));
   end
    for k=3:number_of_nodes_M_N_le_y
        P_non_derive_M_N_y_le(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y_le(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y_le(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y_le(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_P_Q_y_le(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y_le(1,k-2)));
    end
     for k=3:number_of_nodes_R_S_le_y
         P_non_derive_R_S_y_le(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y_le(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y_le(1,k-2)));
     end
```

```
     final_IJ=P_non_derive_I_J_x'*P_non_derive_I_J_y;
        final_KL=P_non_derive_K_L_x'*P_non_derive_K_L_y;
        final_MN=P_non_derive_M_N_x'*P_non_derive_M_N_y;
        final_PQ=P_non_derive_P_Q_x'*P_non_derive_P_Q_y;
        final_RS=P_non_derive_R_S_x'*P_non_derive_R_S_y;
        final_IJ_le=P_non_derive_I_J_x_le'*P_non_derive_I_J_y_le;
        final_KL_le=P_non_derive_K_L_x_le'*P_non_derive_K_L_y_le;
        final_MN_le=P_non_derive_M_N_x_le'*P_non_derive_M_N_y_le;
        final_PQ_le=P_non_derive_P_Q_x_le'*P_non_derive_P_Q_y_le;
        final_RS_le= P_non_derive_R_S_x_le'*P_non_derive_R_S_y_le;
        K_22_alpha_le_trans=K_22_alpha_le_trans
+(sin(sweep_angle_in_radians)*(x_3-
x_4)+cos(sweep_angle_in_radians)*((y_3-
y_4)/4))*(final_KL)'*(final_IJ_le.*final_KL_le.*final_PQ_le.*final_MN_le.*final_RS
    end
end
```

```
Not enough input arguments.

Error in Artificial_stiffness_leading_edge_K_22 (line 14)
 sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 K_22_alpha_root_trans=Artificial_stiffness_root(I,J,K,L,M,N,P,Q,R,S,sweep_angle,s
%  I=2;
% J=2;
% K=2;
%  L=2;
%  M=2;
%  N=2;
%  P=2;
%  Q=2;
%  R=2;
%  S=2;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
 sweep_angle_in_radians=sweep_angle*pi/180;
%  root_of_chord=5;
tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%  gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%  weights_x=[1 1];
% weights_y=[1 1];
 u_o=zeros(I,1);
 P_non_derive_I_J_x=zeros(1,I*J);
P_non_derive_K_L_x=zeros(1,K*L);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_non_derive_K_L_y=zeros(1,K*L);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_non_derive_I_J_x_root=zeros(1,I*J);
P_non_derive_K_L_x_root=zeros(1,K*L);
P_non_derive_M_N_x_root=zeros(1,M*N);
P_non_derive_P_Q_x_root=zeros(1,P*Q);
P_non_derive_R_S_x_root=zeros(1,R*S);
P_non_derive_I_J_y_root=zeros(1,I*J);
P_non_derive_K_L_y_root=zeros(1,K*L);
P_non_derive_M_N_y_root=zeros(1,M*N);
P_non_derive_P_Q_y_root=zeros(1,P*Q);
 P_non_derive_R_S_y_root=zeros(1,R*S);
number_of_nodes_I_J_le_x=I*J;
number_of_nodes_K_L_le_x=K*L;
number_of_nodes_M_N_le_x=M*N;
number_of_nodes_P_Q_le_x=P*Q;
number_of_nodes_R_S_le_x=R*S;
number_of_nodes_I_J_le_y=I*J;
number_of_nodes_K_L_le_y=K*L;
number_of_nodes_M_N_le_y=M*N;
number_of_nodes_R_S_le_y=R*S;
```

```
 number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
 number_of_nodes_I_J_le=((I*J));
number_of_nodes_K_L_le=((K*L));
number_of_nodes_M_N_le=((M*N));
number_of_nodes_P_Q_le=((P*Q));
number_of_nodes_R_S_le=((R*S));
coordinates_x=[0 root_of_chord root_of_chord root_of_chord-
tip_chord]';
coordinates_y=[0 0 span_of_the_entire_wing/2
 span_of_the_entire_wing/2]';
x_1=coordinates_x(1);
y_1=coordinates_y(1);
x_4=coordinates_x(1);
y_1=coordinates_y(1);
K_22_alpha_root_trans=zeros(1,I*J);
for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
        P_non_derive_I_J_x(1,1)=1;
        P_non_derive_I_J_x(1,2)=(gauss_points_x(j));
        P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=(gauss_points_x(j));
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=(gauss_points_y(op));
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=(gauss_points_y(op));
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=(gauss_points_y(op));
P_non_derive_I_J_x_root(1,1)=1;
P_non_derive_I_J_x_root(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y_root(1,1)=1;
P_non_derive_I_J_y_root(1,2)=(gauss_points_y(op));
 P_non_derive_K_L_x_root(1,1)=1;
P_non_derive_K_L_x_root(1,2)=(gauss_points_x(j));
P_non_derive_K_L_y_root(1,1)=1;
P_non_derive_K_L_y_root(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x_root(1,1)=1;
P_non_derive_M_N_x_root(1,2)=gauss_points_x(j);
P_non_derive_M_N_y_root(1,1)=1;
P_non_derive_M_N_y_root(1,2)=gauss_points_y(op);
P_non_derive_P_Q_x_root(1,1)=1;
```

```
P_non_derive_P_Q_x_root(1,2)=(gauss_points_x(j));
P_non_derive_P_Q_y_root(1,1)=1;
P_non_derive_P_Q_y_root(1,2)=gauss_points_y(op);
P_non_derive_R_S_x_root(1,1)=1;
P_non_derive_R_S_x_root(1,2)=(gauss_points_y(op));
P_non_derive_R_S_y_root(1,1)=1;
P_non_derive_R_S_y_root(1,2)=(gauss_points_y(op));
for k=3:number_of_nodes_I_J
    P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2))));
end
for k=3:number_of_nodes_K_L
    P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
end
for k=3:number_of_nodes_M_N
     P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
end
 for k=3:number_of_nodes_P_Q
      P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
 end
  for k=3:number_of_nodes_R_S
       P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
  end
   for k=3:number_of_nodes_I_J
       P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
   end
   for k=3:number_of_nodes_K_L
       P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
   end
    for k=3:number_of_nodes_M_N
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_P_Q_y(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y(1,k-2)));
    end
     for k=3:number_of_nodes_R_S
```

```
        P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
     end
     for k=3:number_of_nodes_I_J_le_x
    P_non_derive_I_J_x_root(1,k)=((((2*k+1)/(k
+1))*(-1)*P_non_derive_I_J_x_root(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x_root(1,k-2))));
end
for k=3:number_of_nodes_K_L_le_x
    P_non_derive_K_L_x_root(1,k)=((((2*k+1)/(k
+1))*(-1))*P_non_derive_K_L_x_root(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x_root(1,k-2)));
end
for k=3:number_of_nodes_M_N_le_x
     P_non_derive_M_N_x_root(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_M_N_x_root(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x_root(1,k-2)));
end
 for k=3:number_of_nodes_P_Q_le_x
      P_non_derive_P_Q_x_root(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_P_Q_x_root(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x_root(1,k-2)));
 end
  for k=3:number_of_nodes_R_S_le_x
       P_non_derive_R_S_x_root(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_R_S_x_root(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x_root(1,k-2)));
  end
   for k=3:number_of_nodes_I_J_le_y
       P_non_derive_I_J_y_root(1,k)=((((2*k+1)/
(k+1))*(-1))*P_non_derive_I_J_y_root(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y_root(1,k-2)));
   end
   for k=3:number_of_nodes_K_L_le_y
       P_non_derive_K_L_y_root(1,k)=((((2*k+1)/
(k+1))*(-1))*P_non_derive_K_L_y_root(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y_root(1,k-2)));
   end
    for k=3:number_of_nodes_M_N_le_y
        P_non_derive_M_N_y_root(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_M_N_y_root(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y_root(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y_root(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_P_Q_y_root(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y_root(1,k-2)));
    end
     for k=3:number_of_nodes_R_S_le_y
         P_non_derive_R_S_y_root(1,k)=((((2*k+1)/
(k+1))*(-1))*P_non_derive_R_S_y_root(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y_root(1,k-2)));
     end
```

```
     final_IJ=P_non_derive_I_J_x'*P_non_derive_I_J_y;
        final_KL=P_non_derive_K_L_x'*P_non_derive_K_L_y;
        final_MN=P_non_derive_M_N_x'*P_non_derive_M_N_y;
        final_PQ=P_non_derive_P_Q_x'*P_non_derive_P_Q_y;
        final_RS=P_non_derive_R_S_x'*P_non_derive_R_S_y;

 final_IJ_root=P_non_derive_I_J_x_root'*P_non_derive_I_J_y_root;

 final_KL_root=P_non_derive_K_L_x_root'*P_non_derive_K_L_y_root;

 final_MN_root=P_non_derive_M_N_x_root'*P_non_derive_M_N_y_root;

 final_PQ_root=P_non_derive_P_Q_x_root'*P_non_derive_P_Q_y_root;
        final_RS_root=
 P_non_derive_R_S_x_root'*P_non_derive_R_S_y_root;
        K_22_alpha_root_trans=K_22_alpha_root_trans+((x_1-
x_4)/4)*(final_KL)'*(final_IJ_root.*final_KL_root.*final_PQ_root.*final_MN_root.*f
    end
end
```

```
Not enough input arguments.

Error in Artificial_stiffness_root_K_22 (line 14)
 sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 K_22_alpha_tip_trans=Artificial_stiffness_tip_K_11(I,J,K,L,M,N,P,Q,R,S,sweep_angl
%  I=2;
% J=2;
% K=2;
%  L=2;
%  M=2;
%  N=2;
%  P=2;
%  Q=2;
%  R=2;
%  S=2;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
 sweep_angle_in_radians=sweep_angle*pi/180;
%  root_of_chord=5;
 tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%  gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%  weights_x=[1 1];
% weights_y=[1 1];
 u_o=zeros(I,1);
 P_non_derive_I_J_x=zeros(1,I*J);
P_non_derive_K_L_x=zeros(1,K*L);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_non_derive_K_L_y=zeros(1,K*L);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_non_derive_I_J_x_root=zeros(1,I*J);
P_non_derive_K_L_x_root=zeros(1,K*L);
P_non_derive_M_N_x_root=zeros(1,M*N);
P_non_derive_P_Q_x_root=zeros(1,P*Q);
P_non_derive_R_S_x_root=zeros(1,R*S);
P_non_derive_I_J_y_root=zeros(1,I*J);
P_non_derive_K_L_y_root=zeros(1,K*L);
P_non_derive_M_N_y_root=zeros(1,M*N);
P_non_derive_P_Q_y_root=zeros(1,P*Q);
 P_non_derive_R_S_y_root=zeros(1,R*S);
number_of_nodes_I_J_le_x=I*J;
number_of_nodes_K_L_le_x=K*L;
number_of_nodes_M_N_le_x=M*N;
number_of_nodes_P_Q_le_x=P*Q;
number_of_nodes_R_S_le_x=R*S;
number_of_nodes_I_J_le_y=I*J;
number_of_nodes_K_L_le_y=K*L;
number_of_nodes_M_N_le_y=M*N;
number_of_nodes_R_S_le_y=R*S;
```

```
 number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
 number_of_nodes_I_J_le=((I*J));
number_of_nodes_K_L_le=((K*L));
number_of_nodes_M_N_le=((M*N));
number_of_nodes_P_Q_le=((P*Q));
number_of_nodes_R_S_le=((R*S));
coordinates_x=[0 root_of_chord root_of_chord root_of_chord-
tip_chord]';
coordinates_y=[0 0 span_of_the_entire_wing/2
 span_of_the_entire_wing/2]';
x_3=coordinates_x(3);
y_3=coordinates_y(3);
x_4=coordinates_x(4);
y_4=coordinates_y(4);
K_22_alpha_tip_trans=zeros(1,I*J);
for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
        P_non_derive_I_J_x(1,1)=1;
        P_non_derive_I_J_x(1,2)=(gauss_points_x(j));
        P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=(gauss_points_x(j));
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=(gauss_points_y(op));
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=(gauss_points_y(op));
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=(gauss_points_y(op));
P_non_derive_I_J_x_root(1,1)=1;
P_non_derive_I_J_x_root(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y_root(1,1)=1;
P_non_derive_I_J_y_root(1,2)=(gauss_points_y(op));
 P_non_derive_K_L_x_root(1,1)=1;
P_non_derive_K_L_x_root(1,2)=(gauss_points_x(j));
P_non_derive_K_L_y_root(1,1)=1;
P_non_derive_K_L_y_root(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x_root(1,1)=1;
P_non_derive_M_N_x_root(1,2)=gauss_points_x(j);
P_non_derive_M_N_y_root(1,1)=1;
P_non_derive_M_N_y_root(1,2)=gauss_points_y(op);
P_non_derive_P_Q_x_root(1,1)=1;
```

```
P_non_derive_P_Q_x_root(1,2)=(gauss_points_x(j));
P_non_derive_P_Q_y_root(1,1)=1;
P_non_derive_P_Q_y_root(1,2)=gauss_points_y(op);
P_non_derive_R_S_x_root(1,1)=1;
P_non_derive_R_S_x_root(1,2)=(gauss_points_y(op));
P_non_derive_R_S_y_root(1,1)=1;
P_non_derive_R_S_y_root(1,2)=(gauss_points_y(op));
for k=3:number_of_nodes_I_J
    P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2))));
end
for k=3:number_of_nodes_K_L
    P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
end
for k=3:number_of_nodes_M_N
     P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
end
 for k=3:number_of_nodes_P_Q
      P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
 end
  for k=3:number_of_nodes_R_S
       P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
  end
   for k=3:number_of_nodes_I_J
       P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
   end
   for k=3:number_of_nodes_K_L
       P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
   end
    for k=3:number_of_nodes_M_N
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_P_Q_y(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y(1,k-2)));
    end
     for k=3:number_of_nodes_R_S
```

```
         P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
     end
     for k=3:number_of_nodes_I_J_le_x
    P_non_derive_I_J_x_root(1,k)=((((2*k+1)/(k
+1))*(1)*P_non_derive_I_J_x_root(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x_root(1,k-2))));
end
for k=3:number_of_nodes_K_L_le_x
    P_non_derive_K_L_x_root(1,k)=((((2*k+1)/(k
+1))*(1))*P_non_derive_K_L_x_root(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x_root(1,k-2)));
end
for k=3:number_of_nodes_M_N_le_x
     P_non_derive_M_N_x_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_M_N_x_root(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x_root(1,k-2)));
end
 for k=3:number_of_nodes_P_Q_le_x
      P_non_derive_P_Q_x_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_P_Q_x_root(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x_root(1,k-2)));
 end
  for k=3:number_of_nodes_R_S_le_x
       P_non_derive_R_S_x_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_R_S_x_root(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x_root(1,k-2)));
  end
   for k=3:number_of_nodes_I_J_le_y
       P_non_derive_I_J_y_root(1,k)=((((2*k+1)/
(k+1))*(1))*P_non_derive_I_J_y_root(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y_root(1,k-2)));
   end
   for k=3:number_of_nodes_K_L_le_y
       P_non_derive_K_L_y_root(1,k)=((((2*k+1)/
(k+1))*(1))*P_non_derive_K_L_y_root(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y_root(1,k-2)));
   end
    for k=3:number_of_nodes_M_N_le_y
        P_non_derive_M_N_y_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_M_N_y_root(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y_root(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_P_Q_y_root(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y_root(1,k-2)));
    end
     for k=3:number_of_nodes_R_S_le_y
         P_non_derive_R_S_y_root(1,k)=((((2*k+1)/
(k+1))*(1))*P_non_derive_R_S_y_root(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y_root(1,k-2)));
     end
```

```
     final_IJ=P_non_derive_I_J_x'*P_non_derive_I_J_y;
        final_KL=P_non_derive_K_L_x'*P_non_derive_K_L_y;
        final_MN=P_non_derive_M_N_x'*P_non_derive_M_N_y;
        final_PQ=P_non_derive_P_Q_x'*P_non_derive_P_Q_y;
        final_RS=P_non_derive_R_S_x'*P_non_derive_R_S_y;

 final_IJ_root=P_non_derive_I_J_x_root'*P_non_derive_I_J_y_root;

 final_KL_root=P_non_derive_K_L_x_root'*P_non_derive_K_L_y_root;

 final_MN_root=P_non_derive_M_N_x_root'*P_non_derive_M_N_y_root;

 final_PQ_root=P_non_derive_P_Q_x_root'*P_non_derive_P_Q_y_root;
        final_RS_root=
 P_non_derive_R_S_x_root'*P_non_derive_R_S_y_root;
        K_22_alpha_tip_trans=K_22_alpha_tip_trans+((x_3-
x_4)/4)*(final_PQ)'*(final_IJ_root.*final_KL_root.*final_PQ_root.*final_MN_root.*f
    end
end
```

```
Not enough input arguments.

Error in Artificial_stiffness_tip_K_22 (line 14)
 sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 K_22_alpha_te_trans=Artificial_stiffness_trailing_edge_K_22(I,J,K,L,M,N,P,Q,R,S,s
%  I=2;
% J=2;
% K=2;
%  L=2;
%  M=2;
%  N=2;
%  P=2;
%  Q=2;
%  R=2;
%  S=2;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
 sweep_angle_in_radians=sweep_angle*pi/180;
%  root_of_chord=5;
 tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%  gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%  weights_x=[1 1];
% weights_y=[1 1];
 u_o=zeros(I,1);
 P_non_derive_I_J_x=zeros(1,I*J);
P_non_derive_K_L_x=zeros(1,K*L);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_non_derive_K_L_y=zeros(1,K*L);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_non_derive_I_J_x_te=zeros(1,I*J);
P_non_derive_K_L_x_te=zeros(1,K*L);
P_non_derive_M_N_x_te=zeros(1,M*N);
P_non_derive_P_Q_x_te=zeros(1,P*Q);
P_non_derive_R_S_x_te=zeros(1,R*S);
P_non_derive_I_J_y_te=zeros(1,I*J);
P_non_derive_K_L_y_te=zeros(1,K*L);
P_non_derive_M_N_y_te=zeros(1,M*N);
P_non_derive_P_Q_y_te=zeros(1,P*Q);
 P_non_derive_R_S_y_te=zeros(1,R*S);
number_of_nodes_I_J_te_x=I*J;
number_of_nodes_K_L_te_x=K*L;
number_of_nodes_M_N_te_x=M*N;
number_of_nodes_P_Q_te_x=P*Q;
number_of_nodes_R_S_te_x=R*S;
number_of_nodes_I_J_te_y=I*J;
number_of_nodes_K_L_te_y=K*L;
number_of_nodes_M_N_te_y=M*N;
number_of_nodes_R_S_te_y=R*S;
```

```
 number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
 number_of_nodes_I_J_le=((I*J));
number_of_nodes_K_L_le=((K*L));
number_of_nodes_M_N_le=((M*N));
number_of_nodes_P_Q_le=((P*Q));
number_of_nodes_R_S_le=((R*S));
coordinates_x=[0 root_of_chord root_of_chord root_of_chord-
tip_chord]';
coordinates_y=[0 0 span_of_the_entire_wing/2
 span_of_the_entire_wing/2]';
x_3=coordinates_x(3);
y_3=coordinates_y(3);
x_2=coordinates_x(2);
y_2=coordinates_y(2);
K_22_alpha_te_trans=zeros(1,I*J);
for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
        P_non_derive_I_J_x(1,1)=1;
        P_non_derive_I_J_x(1,2)=(gauss_points_x(j));
        P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=(gauss_points_x(j));
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=(gauss_points_y(op));
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=(gauss_points_y(op));
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=(gauss_points_y(op));
P_non_derive_I_J_x_te(1,1)=1;
P_non_derive_I_J_x_te(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y_te(1,1)=1;
P_non_derive_I_J_y_te(1,2)=(gauss_points_y(op));
 P_non_derive_K_L_x_te(1,1)=1;
P_non_derive_K_L_x_te(1,2)=(gauss_points_x(j));
P_non_derive_K_L_y_te(1,1)=1;
P_non_derive_K_L_y_te(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x_te(1,1)=1;
P_non_derive_M_N_x_te(1,2)=gauss_points_x(j);
P_non_derive_M_N_y_te(1,1)=1;
P_non_derive_M_N_y_te(1,2)=gauss_points_y(op);
P_non_derive_P_Q_x_te(1,1)=1;
```

```
P_non_derive_P_Q_x_te(1,2)=(gauss_points_x(j));
P_non_derive_P_Q_y_te(1,1)=1;
P_non_derive_P_Q_y_te(1,2)=gauss_points_y(op);
P_non_derive_R_S_x_te(1,1)=1;
P_non_derive_R_S_x_te(1,2)=(gauss_points_y(op));
P_non_derive_R_S_y_te(1,1)=1;
P_non_derive_R_S_y_te(1,2)=(gauss_points_y(op));
for k=3:number_of_nodes_I_J
    P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2))));
end
for k=3:number_of_nodes_K_L
    P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
end
for k=3:number_of_nodes_M_N
     P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
end
 for k=3:number_of_nodes_P_Q
      P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
 end
  for k=3:number_of_nodes_R_S
       P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
  end
   for k=3:number_of_nodes_I_J
       P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
   end
   for k=3:number_of_nodes_K_L
       P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
   end
    for k=3:number_of_nodes_M_N
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_P_Q_y(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y(1,k-2)));
    end
     for k=3:number_of_nodes_R_S
```

```
         P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
     end
     for k=3:number_of_nodes_I_J_te_x
    P_non_derive_I_J_x_te(1,k)=((((2*k+1)/(k
+1))*(1)*P_non_derive_I_J_x_te(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x_te(1,k-2))));
end
for k=3:number_of_nodes_K_L_te_x
    P_non_derive_K_L_x_te(1,k)=((((2*k+1)/(k
+1))*(1))*P_non_derive_K_L_x_te(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x_te(1,k-2)));
end
for k=3:number_of_nodes_M_N_te_x
     P_non_derive_M_N_x_te(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_M_N_x_te(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x_te(1,k-2)));
end
 for k=3:number_of_nodes_P_Q_te_x
      P_non_derive_P_Q_x_te(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_P_Q_x_te(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x_te(1,k-2)));
 end
  for k=3:number_of_nodes_R_S_te_x
       P_non_derive_R_S_x_te(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_R_S_x_te(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x_te(1,k-2)));
  end
   for k=3:number_of_nodes_I_J_te_y
       P_non_derive_I_J_y_te(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y_te(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y_te(1,k-2)));
   end
   for k=3:number_of_nodes_K_L_te_y
       P_non_derive_K_L_y_te(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y_te(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y_te(1,k-2)));
   end
    for k=3:number_of_nodes_M_N_te_y
        P_non_derive_M_N_y_te(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y_te(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y_te(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y_te(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_P_Q_y_te(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y_te(1,k-2)));
    end
     for k=3:number_of_nodes_R_S_te_y
         P_non_derive_R_S_y_te(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y_te(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y_te(1,k-2)));
     end
```

```
     final_IJ=P_non_derive_I_J_x.*P_non_derive_I_J_y;
        final_KL=P_non_derive_K_L_x.*P_non_derive_K_L_y;
        final_MN=P_non_derive_M_N_x.*P_non_derive_M_N_y;
        final_PQ=P_non_derive_P_Q_x.*P_non_derive_P_Q_y;
        final_RS=P_non_derive_R_S_x.*P_non_derive_R_S_y;
        final_IJ_le=P_non_derive_I_J_x_te.*P_non_derive_I_J_y_te;
        final_KL_le=P_non_derive_K_L_x_te.*P_non_derive_K_L_y_te;
        final_MN_le=P_non_derive_M_N_x_te.*P_non_derive_M_N_y_te;
        final_PQ_le=P_non_derive_P_Q_x_te.*P_non_derive_P_Q_y_te;
        final_RS_le= P_non_derive_R_S_x_te.*P_non_derive_R_S_y_te;
        K_22_alpha_te_trans=K_22_alpha_te_trans+(sin(0)*(x_3-
x_2)+cos(0)*((y_3-
y_2))/4)*(final_KL)'*(final_IJ_le.*final_KL_le.*final_PQ_le.*final_MN_le.*final_RS
    end
end
```

```
Not enough input arguments.

Error in Artificial_stiffness_trailing_edge_K_22 (line 14)
 sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 K_33=Artificial_stiffness_K_11(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span_of_the_entire
%  I=4;
%  J=4;
%  K=4;
%   L=4;
%  M=4;
%   N=4;
%  P=4;
%  Q=4;
%   R=4;
%   S=4;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
%  root_of_chord=5;
 %tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%   gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
%  gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%   weights_x=[1 1];
%  weights_y=[1 1];
%  gauss_points_trans_x=0;
%  gauss_points_trans_y=0;
%  weights_trans_x=2;
%  weights_trans_y=2;
K_11_1=Artificial_stiffness_leading_edge_K_33(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span
K_11_2=Artificial_stiffness_root_K_33(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span_of_the_
K_11_3=Artificial_stiffness_tip_K_33(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span_of_the_e
K_11_4=Artificial_stiffness_trailing_edge_K_33(I,J,K,L,M,N,P,Q,R,S,sweep_angle,spa
K_33=K_11_1+K_11_2+K_11_3+K_11_4;
end
```

```
Not enough input arguments.

Error in Artificial_stiffness_K_33 (line 24)
K_11_1=Artificial_stiffness_leading_edge_K_33(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span
```

Published with MATLAB® R2020a

```
function
 K_33_alpha_le_trans=Artificial_stiffness_leading_edge(I,J,K,L,M,N,P,Q,R,S,sweep_a
%  I=2;
% J=2;
% K=2;
%  L=2;
%  M=2;
%  N=2;
%  P=2;
%  Q=2;
%  R=2;
%  S=2;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
 sweep_angle_in_radians=sweep_angle*pi/180;
%  root_of_chord=5;
tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%  gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%  weights_x=[1 1];
% weights_y=[1 1];
 u_o=zeros(I,1);
 P_non_derive_I_J_x=zeros(1,I*J);
P_non_derive_K_L_x=zeros(1,K*L);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_non_derive_K_L_y=zeros(1,K*L);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_non_derive_I_J_x_le=zeros(1,I*J);
P_non_derive_K_L_x_le=zeros(1,K*L);
P_non_derive_M_N_x_le=zeros(1,M*N);
P_non_derive_P_Q_x_le=zeros(1,P*Q);
P_non_derive_R_S_x_le=zeros(1,R*S);
P_non_derive_I_J_y_le=zeros(1,I*J);
P_non_derive_K_L_y_le=zeros(1,K*L);
P_non_derive_M_N_y_le=zeros(1,M*N);
P_non_derive_P_Q_y_le=zeros(1,P*Q);
 P_non_derive_R_S_y_le=zeros(1,R*S);
number_of_nodes_I_J_le_x=I*J;
number_of_nodes_K_L_le_x=K*L;
number_of_nodes_M_N_le_x=M*N;
number_of_nodes_P_Q_le_x=P*Q;
number_of_nodes_R_S_le_x=R*S;
number_of_nodes_I_J_le_y=I*J;
number_of_nodes_K_L_le_y=K*L;
number_of_nodes_M_N_le_y=M*N;
number_of_nodes_R_S_le_y=R*S;
```

```
 number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
 number_of_nodes_I_J_le=((I*J));
number_of_nodes_K_L_le=((K*L));
number_of_nodes_M_N_le=((M*N));
number_of_nodes_P_Q_le=((P*Q));
number_of_nodes_R_S_le=((R*S));
coordinates_x=[0 root_of_chord root_of_chord root_of_chord-
tip_chord]';
coordinates_y=[0 0 span_of_the_entire_wing/2
 span_of_the_entire_wing/2]';
x_2=coordinates_x(2);
y_2=coordinates_y(2);
x_3=coordinates_x(3);
y_3=coordinates_y(3);
K_33_alpha_le_trans=zeros(1,I*J);
for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
        P_non_derive_I_J_x(1,1)=1;
        P_non_derive_I_J_x(1,2)=(gauss_points_x(j));
        P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=(gauss_points_x(j));
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=(gauss_points_y(op));
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=(gauss_points_y(op));
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=(gauss_points_y(op));
P_non_derive_I_J_x_le(1,1)=1;
P_non_derive_I_J_x_le(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y_le(1,1)=1;
P_non_derive_I_J_y_le(1,2)=(gauss_points_y(op));
 P_non_derive_K_L_x_le(1,1)=1;
P_non_derive_K_L_x_le(1,2)=(gauss_points_x(j));
P_non_derive_K_L_y_le(1,1)=1;
P_non_derive_K_L_y_le(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x_le(1,1)=1;
P_non_derive_M_N_x_le(1,2)=gauss_points_x(j);
P_non_derive_M_N_y_le(1,1)=1;
P_non_derive_M_N_y_le(1,2)=gauss_points_y(op);
P_non_derive_P_Q_x_le(1,1)=1;
```

```
P_non_derive_P_Q_x_le(1,2)=(gauss_points_x(j));
P_non_derive_P_Q_y_le(1,1)=1;
P_non_derive_P_Q_y_le(1,2)=gauss_points_y(op);
P_non_derive_R_S_x_le(1,1)=1;
P_non_derive_R_S_x_le(1,2)=(gauss_points_y(op));
P_non_derive_R_S_y_le(1,1)=1;
P_non_derive_R_S_y_le(1,2)=(gauss_points_y(op));
for k=3:number_of_nodes_I_J
    P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2))));
end
for k=3:number_of_nodes_K_L
    P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
end
for k=3:number_of_nodes_M_N
     P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
end
for k=3:number_of_nodes_M_N
     P_non_derive_P_Q_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_P_Q_y(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y(1,k-2)));
end
 for k=3:number_of_nodes_P_Q
      P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
 end
  for k=3:number_of_nodes_R_S
       P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
  end
   for k=3:number_of_nodes_I_J
       P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
   end
   for k=3:number_of_nodes_K_L
       P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
   end
    for k=3:number_of_nodes_M_N
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
     for k=3:number_of_nodes_R_S
```

```
         P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
     end
     for k=3:number_of_nodes_I_J_le_x
    P_non_derive_I_J_x_le(1,k)=((((2*k+1)/(k
+1))*(-1)*P_non_derive_I_J_x_le(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x_le(1,k-2))));
end
for k=3:number_of_nodes_K_L_le_x
    P_non_derive_K_L_x_le(1,k)=((((2*k+1)/(k
+1))*(-1))*P_non_derive_K_L_x_le(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x_le(1,k-2)));
end
for k=3:number_of_nodes_M_N_le_x
     P_non_derive_M_N_x_le(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_M_N_x_le(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x_le(1,k-2)));
end
 for k=3:number_of_nodes_P_Q_le_x
      P_non_derive_P_Q_x_le(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_P_Q_x_le(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x_le(1,k-2)));
 end
  for k=3:number_of_nodes_R_S_le_x
       P_non_derive_R_S_x_le(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_R_S_x_le(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x_le(1,k-2)));
  end
   for k=3:number_of_nodes_I_J_le_y
       P_non_derive_I_J_y_le(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y_le(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y_le(1,k-2)));
   end
   for k=3:number_of_nodes_K_L_le_y
       P_non_derive_K_L_y_le(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y_le(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y_le(1,k-2)));
   end
    for k=3:number_of_nodes_M_N_le_y
        P_non_derive_M_N_y_le(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y_le(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y_le(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y_le(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_P_Q_y_le(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y_le(1,k-2)));
    end
     for k=3:number_of_nodes_R_S_le_y
         P_non_derive_R_S_y_le(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y_le(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y_le(1,k-2)));
     end
```

```
     final_IJ=P_non_derive_I_J_x'*P_non_derive_I_J_y;
        final_KL=P_non_derive_K_L_x'*P_non_derive_K_L_y;
        final_MN=P_non_derive_M_N_x'*P_non_derive_M_N_y;
        final_PQ=P_non_derive_P_Q_x'*P_non_derive_P_Q_y;
        final_RS=P_non_derive_R_S_x'*P_non_derive_R_S_y;
        final_IJ_le=P_non_derive_I_J_x_le'*P_non_derive_I_J_y_le;
        final_KL_le=P_non_derive_K_L_x_le'*P_non_derive_K_L_y_le;
        final_MN_le=P_non_derive_M_N_x_le'*P_non_derive_M_N_y_le;
        final_PQ_le=P_non_derive_P_Q_x_le'*P_non_derive_P_Q_y_le;
        final_RS_le= P_non_derive_R_S_x_le'*P_non_derive_R_S_y_le;
        K_33_alpha_le_trans=K_33_alpha_le_trans
+(sin(sweep_angle_in_radians)*(x_2-
x_3)+cos(sweep_angle_in_radians)*((y_2-
y_3)/4))*(final_MN)'*(final_IJ_le.*final_KL_le.*final_PQ_le.*final_MN_le.*final_RS
    end
end
```

```
Not enough input arguments.

Error in Artificial_stiffness_leading_edge_K_33 (line 14)
 sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 K_33_alpha_tip_trans=Artificial_stiffness_tip_K_11(I,J,K,L,M,N,P,Q,R,S,sweep_angl
%  I=2;
% J=2;
% K=2;
%  L=2;
%  M=2;
%  N=2;
%  P=2;
%  Q=2;
%  R=2;
%  S=2;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
 sweep_angle_in_radians=sweep_angle*pi/180;
%  root_of_chord=5;
 tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%  gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%  weights_x=[1 1];
% weights_y=[1 1];
 u_o=zeros(I,1);
 P_non_derive_I_J_x=zeros(1,I*J);
P_non_derive_K_L_x=zeros(1,K*L);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_non_derive_K_L_y=zeros(1,K*L);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_non_derive_I_J_x_root=zeros(1,I*J);
P_non_derive_K_L_x_root=zeros(1,K*L);
P_non_derive_M_N_x_root=zeros(1,M*N);
P_non_derive_P_Q_x_root=zeros(1,P*Q);
P_non_derive_R_S_x_root=zeros(1,R*S);
P_non_derive_I_J_y_root=zeros(1,I*J);
P_non_derive_K_L_y_root=zeros(1,K*L);
P_non_derive_M_N_y_root=zeros(1,M*N);
P_non_derive_P_Q_y_root=zeros(1,P*Q);
 P_non_derive_R_S_y_root=zeros(1,R*S);
number_of_nodes_I_J_le_x=I*J;
number_of_nodes_K_L_le_x=K*L;
number_of_nodes_M_N_le_x=M*N;
number_of_nodes_P_Q_le_x=P*Q;
number_of_nodes_R_S_le_x=R*S;
number_of_nodes_I_J_le_y=I*J;
number_of_nodes_K_L_le_y=K*L;
number_of_nodes_M_N_le_y=M*N;
number_of_nodes_R_S_le_y=R*S;
```

```
 number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
 number_of_nodes_I_J_le=((I*J));
number_of_nodes_K_L_le=((K*L));
number_of_nodes_M_N_le=((M*N));
number_of_nodes_P_Q_le=((P*Q));
number_of_nodes_R_S_le=((R*S));
coordinates_x=[0 root_of_chord root_of_chord root_of_chord-
tip_chord]';
coordinates_y=[0 0 span_of_the_entire_wing/2
 span_of_the_entire_wing/2]';
x_3=coordinates_x(3);
y_3=coordinates_y(3);
x_4=coordinates_x(4);
y_4=coordinates_y(4);
K_33_alpha_tip_trans=zeros(1,I*J);
for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
        P_non_derive_I_J_x(1,1)=1;
        P_non_derive_I_J_x(1,2)=(gauss_points_x(j));
        P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=(gauss_points_x(j));
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=(gauss_points_y(op));
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=(gauss_points_y(op));
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=(gauss_points_y(op));
P_non_derive_I_J_x_root(1,1)=1;
P_non_derive_I_J_x_root(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y_root(1,1)=1;
P_non_derive_I_J_y_root(1,2)=(gauss_points_y(op));
 P_non_derive_K_L_x_root(1,1)=1;
P_non_derive_K_L_x_root(1,2)=(gauss_points_x(j));
P_non_derive_K_L_y_root(1,1)=1;
P_non_derive_K_L_y_root(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x_root(1,1)=1;
P_non_derive_M_N_x_root(1,2)=gauss_points_x(j);
P_non_derive_M_N_y_root(1,1)=1;
P_non_derive_M_N_y_root(1,2)=gauss_points_y(op);
P_non_derive_P_Q_x_root(1,1)=1;
```

```
P_non_derive_P_Q_x_root(1,2)=(gauss_points_x(j));
P_non_derive_P_Q_y_root(1,1)=1;
P_non_derive_P_Q_y_root(1,2)=gauss_points_y(op);
P_non_derive_R_S_x_root(1,1)=1;
P_non_derive_R_S_x_root(1,2)=(gauss_points_y(op));
P_non_derive_R_S_y_root(1,1)=1;
P_non_derive_R_S_y_root(1,2)=(gauss_points_y(op));
for k=3:number_of_nodes_I_J
    P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2))));
end
for k=3:number_of_nodes_K_L
    P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
end
for k=3:number_of_nodes_M_N
     P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
end
 for k=3:number_of_nodes_P_Q
      P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
 end
  for k=3:number_of_nodes_R_S
       P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
  end
   for k=3:number_of_nodes_I_J
       P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
   end
   for k=3:number_of_nodes_K_L
       P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
   end
    for k=3:number_of_nodes_M_N
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
    for k=3:number_of_nodes_M_N
     P_non_derive_P_Q_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_P_Q_y(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y(1,k-2)));
end
     for k=3:number_of_nodes_R_S
```

```
         P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
     end
     for k=3:number_of_nodes_I_J_le_x
    P_non_derive_I_J_x_root(1,k)=((((2*k+1)/(k
+1))*(1)*P_non_derive_I_J_x_root(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x_root(1,k-2))));
end
for k=3:number_of_nodes_K_L_le_x
    P_non_derive_K_L_x_root(1,k)=((((2*k+1)/(k
+1))*(1))*P_non_derive_K_L_x_root(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x_root(1,k-2)));
end
for k=3:number_of_nodes_M_N_le_x
     P_non_derive_M_N_x_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_M_N_x_root(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x_root(1,k-2)));
end
 for k=3:number_of_nodes_P_Q_le_x
      P_non_derive_P_Q_x_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_P_Q_x_root(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x_root(1,k-2)));
 end
  for k=3:number_of_nodes_R_S_le_x
       P_non_derive_R_S_x_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_R_S_x_root(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x_root(1,k-2)));
  end
   for k=3:number_of_nodes_I_J_le_y
       P_non_derive_I_J_y_root(1,k)=((((2*k+1)/
(k+1))*(1))*P_non_derive_I_J_y_root(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y_root(1,k-2)));
   end
   for k=3:number_of_nodes_K_L_le_y
       P_non_derive_K_L_y_root(1,k)=((((2*k+1)/
(k+1))*(1))*P_non_derive_K_L_y_root(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y_root(1,k-2)));
   end
    for k=3:number_of_nodes_M_N_le_y
        P_non_derive_M_N_y_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_M_N_y_root(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y_root(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_P_Q_y_root(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y_root(1,k-2)));
    end
     for k=3:number_of_nodes_R_S_le_y
         P_non_derive_R_S_y_root(1,k)=((((2*k+1)/
(k+1))*(1))*P_non_derive_R_S_y_root(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y_root(1,k-2)));
     end
```

```
     final_IJ=P_non_derive_I_J_x.*P_non_derive_I_J_y;
        final_KL=P_non_derive_K_L_x.*P_non_derive_K_L_y;
        final_MN=P_non_derive_M_N_x.*P_non_derive_M_N_y;
        final_PQ=P_non_derive_P_Q_x.*P_non_derive_P_Q_y;
        final_RS=P_non_derive_R_S_x.*P_non_derive_R_S_y;

 final_IJ_root=P_non_derive_I_J_x_root.*P_non_derive_I_J_y_root;

 final_KL_root=P_non_derive_K_L_x_root.*P_non_derive_K_L_y_root;

 final_MN_root=P_non_derive_M_N_x_root.*P_non_derive_M_N_y_root;

 final_PQ_root=P_non_derive_P_Q_x_root.*P_non_derive_P_Q_y_root;
        final_RS_root=
 P_non_derive_R_S_x_root.*P_non_derive_R_S_y_root;
        K_33_alpha_tip_trans=K_33_alpha_tip_trans+((x_3-
x_4)/4)*(final_IJ_root.*final_KL_root.*final_PQ_root.*final_MN_root.*final_RS_root
    end
end
end
```

```
Not enough input arguments.

Error in Artificial_stiffness_tip_K_33 (line 14)
 sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 K_33_alpha_root_trans=Artificial_stiffness_root(I,J,K,L,M,N,P,Q,R,S,sweep_angle,s
%  I=2;
% J=2;
% K=2;
%  L=2;
%  M=2;
%  N=2;
%  P=2;
%  Q=2;
%  R=2;
%  S=2;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
 sweep_angle_in_radians=sweep_angle*pi/180;
%  root_of_chord=5;
tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%  gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%  weights_x=[1 1];
% weights_y=[1 1];
 u_o=zeros(I,1);
 P_non_derive_I_J_x=zeros(1,I*J);
P_non_derive_K_L_x=zeros(1,K*L);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_non_derive_K_L_y=zeros(1,K*L);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_non_derive_I_J_x_root=zeros(1,I*J);
P_non_derive_K_L_x_root=zeros(1,K*L);
P_non_derive_M_N_x_root=zeros(1,M*N);
P_non_derive_P_Q_x_root=zeros(1,P*Q);
P_non_derive_R_S_x_root=zeros(1,R*S);
P_non_derive_I_J_y_root=zeros(1,I*J);
P_non_derive_K_L_y_root=zeros(1,K*L);
P_non_derive_M_N_y_root=zeros(1,M*N);
P_non_derive_P_Q_y_root=zeros(1,P*Q);
 P_non_derive_R_S_y_root=zeros(1,R*S);
number_of_nodes_I_J_le_x=I*J;
number_of_nodes_K_L_le_x=K*L;
number_of_nodes_M_N_le_x=M*N;
number_of_nodes_P_Q_le_x=P*Q;
number_of_nodes_R_S_le_x=R*S;
number_of_nodes_I_J_le_y=I*J;
number_of_nodes_K_L_le_y=K*L;
number_of_nodes_M_N_le_y=M*N;
number_of_nodes_R_S_le_y=R*S;
```

```
 number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
 number_of_nodes_I_J_le=((I*J));
number_of_nodes_K_L_le=((K*L));
number_of_nodes_M_N_le=((M*N));
number_of_nodes_P_Q_le=((P*Q));
number_of_nodes_R_S_le=((R*S));
coordinates_x=[0 root_of_chord root_of_chord root_of_chord-
tip_chord]';
coordinates_y=[0 0 span_of_the_entire_wing/2
 span_of_the_entire_wing/2]';
x_2=coordinates_x(2);
y_2=coordinates_y(2);
x_3=coordinates_x(3);
y_3=coordinates_y(3);
K_33_alpha_root_trans=zeros(1,I*J);
for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
        P_non_derive_I_J_x(1,1)=1;
        P_non_derive_I_J_x(1,2)=(gauss_points_x(j));
        P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=(gauss_points_x(j));
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=(gauss_points_y(op));
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=(gauss_points_y(op));
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=(gauss_points_y(op));
P_non_derive_I_J_x_root(1,1)=1;
P_non_derive_I_J_x_root(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y_root(1,1)=1;
P_non_derive_I_J_y_root(1,2)=(gauss_points_y(op));
 P_non_derive_K_L_x_root(1,1)=1;
P_non_derive_K_L_x_root(1,2)=(gauss_points_x(j));
P_non_derive_K_L_y_root(1,1)=1;
P_non_derive_K_L_y_root(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x_root(1,1)=1;
P_non_derive_M_N_x_root(1,2)=gauss_points_x(j);
P_non_derive_M_N_y_root(1,1)=1;
P_non_derive_M_N_y_root(1,2)=gauss_points_y(op);
P_non_derive_P_Q_x_root(1,1)=1;
```

```
P_non_derive_P_Q_x_root(1,2)=(gauss_points_x(j));
P_non_derive_P_Q_y_root(1,1)=1;
P_non_derive_P_Q_y_root(1,2)=gauss_points_y(op);
P_non_derive_R_S_x_root(1,1)=1;
P_non_derive_R_S_x_root(1,2)=(gauss_points_y(op));
P_non_derive_R_S_y_root(1,1)=1;
P_non_derive_R_S_y_root(1,2)=(gauss_points_y(op));
for k=3:number_of_nodes_I_J
    P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2))));
end
for k=3:number_of_nodes_K_L
    P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
end
for k=3:number_of_nodes_M_N
     P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
end
 for k=3:number_of_nodes_P_Q
      P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
 end
  for k=3:number_of_nodes_R_S
       P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
  end
   for k=3:number_of_nodes_I_J
       P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
   end
   for k=3:number_of_nodes_K_L
       P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
   end
    for k=3:number_of_nodes_M_N
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
    for k=3:number_of_nodes_M_N
     P_non_derive_P_Q_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_P_Q_y(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y(1,k-2)));
end
     for k=3:number_of_nodes_R_S
```

```
        P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
     end
     for k=3:number_of_nodes_I_J_le_x
    P_non_derive_I_J_x_root(1,k)=((((2*k+1)/(k
+1))*(-1)*P_non_derive_I_J_x_root(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x_root(1,k-2))));
end
for k=3:number_of_nodes_K_L_le_x
    P_non_derive_K_L_x_root(1,k)=((((2*k+1)/(k
+1))*(-1))*P_non_derive_K_L_x_root(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x_root(1,k-2)));
end
for k=3:number_of_nodes_M_N_le_x
     P_non_derive_M_N_x_root(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_M_N_x_root(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x_root(1,k-2)));
end
 for k=3:number_of_nodes_P_Q_le_x
      P_non_derive_P_Q_x_root(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_P_Q_x_root(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x_root(1,k-2)));
 end
  for k=3:number_of_nodes_R_S_le_x
       P_non_derive_R_S_x_root(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_R_S_x_root(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x_root(1,k-2)));
  end
   for k=3:number_of_nodes_I_J_le_y
       P_non_derive_I_J_y_root(1,k)=((((2*k+1)/
(k+1))*(-1))*P_non_derive_I_J_y_root(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y_root(1,k-2)));
   end
   for k=3:number_of_nodes_K_L_le_y
       P_non_derive_K_L_y_root(1,k)=((((2*k+1)/
(k+1))*(-1))*P_non_derive_K_L_y_root(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y_root(1,k-2)));
   end
    for k=3:number_of_nodes_M_N_le_y
        P_non_derive_M_N_y_root(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_M_N_y_root(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y_root(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y_root(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_P_Q_y_root(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y_root(1,k-2)));
    end
     for k=3:number_of_nodes_R_S_le_y
         P_non_derive_R_S_y_root(1,k)=((((2*k+1)/
(k+1))*(-1))*P_non_derive_R_S_y_root(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y_root(1,k-2)));
     end
```

```
     final_IJ=P_non_derive_I_J_x'*P_non_derive_I_J_y;
        final_KL=P_non_derive_K_L_x'*P_non_derive_K_L_y;
        final_MN=P_non_derive_M_N_x'*P_non_derive_M_N_y;
        final_PQ=P_non_derive_P_Q_x'*P_non_derive_P_Q_y;
        final_RS=P_non_derive_R_S_x'*P_non_derive_R_S_y;

 final_IJ_root=P_non_derive_I_J_x_root'*P_non_derive_I_J_y_root;

 final_KL_root=P_non_derive_K_L_x_root'*P_non_derive_K_L_y_root;

 final_MN_root=P_non_derive_M_N_x_root'*P_non_derive_M_N_y_root;

 final_PQ_root=P_non_derive_P_Q_x_root'*P_non_derive_P_Q_y_root;
        final_RS_root=
 P_non_derive_R_S_x_root'*P_non_derive_R_S_y_root;
        K_33_alpha_root_trans=K_33_alpha_root_trans+((x_2-
x_3)/4)*(final_MN)'*(final_IJ_root.*final_KL_root.*final_PQ_root.*final_MN_root.*f
    end
end
```

```
Not enough input arguments.

Error in Artificial_stiffness_root_K_33 (line 14)
 sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 K_33_alpha_te_trans=Artificial_stiffness_trailing_edge_K_11(I,J,K,L,M,N,P,Q,R,S,s
%  I=2;
% J=2;
% K=2;
%  L=2;
%  M=2;
%  N=2;
%  P=2;
%  Q=2;
%  R=2;
%  S=2;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
 sweep_angle_in_radians=sweep_angle*pi/180;
%  root_of_chord=5;
tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%  gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%  weights_x=[1 1];
% weights_y=[1 1];
 u_o=zeros(I,1);
 P_non_derive_I_J_x=zeros(1,I*J);
P_non_derive_K_L_x=zeros(1,K*L);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_non_derive_K_L_y=zeros(1,K*L);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_non_derive_I_J_x_te=zeros(1,I*J);
P_non_derive_K_L_x_te=zeros(1,K*L);
P_non_derive_M_N_x_te=zeros(1,M*N);
P_non_derive_P_Q_x_te=zeros(1,P*Q);
P_non_derive_R_S_x_te=zeros(1,R*S);
P_non_derive_I_J_y_te=zeros(1,I*J);
P_non_derive_K_L_y_te=zeros(1,K*L);
P_non_derive_M_N_y_te=zeros(1,M*N);
P_non_derive_P_Q_y_te=zeros(1,P*Q);
 P_non_derive_R_S_y_te=zeros(1,R*S);
number_of_nodes_I_J_te_x=I*J;
number_of_nodes_K_L_te_x=K*L;
number_of_nodes_M_N_te_x=M*N;
number_of_nodes_P_Q_te_x=P*Q;
number_of_nodes_R_S_te_x=R*S;
number_of_nodes_I_J_te_y=I*J;
number_of_nodes_K_L_te_y=K*L;
number_of_nodes_M_N_te_y=M*N;
number_of_nodes_R_S_te_y=R*S;
```

```
 number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
 number_of_nodes_I_J_le=((I*J));
number_of_nodes_K_L_le=((K*L));
number_of_nodes_M_N_le=((M*N));
number_of_nodes_P_Q_le=((P*Q));
number_of_nodes_R_S_le=((R*S));
coordinates_x=[0 root_of_chord root_of_chord root_of_chord-
tip_chord]';
coordinates_y=[0 0 span_of_the_entire_wing/2
 span_of_the_entire_wing/2]';
x_3=coordinates_x(3);
y_3=coordinates_y(3);
x_2=coordinates_x(2);
y_2=coordinates_y(2);
K_33_alpha_te_trans=zeros(1,I*J);
for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
        P_non_derive_I_J_x(1,1)=1;
        P_non_derive_I_J_x(1,2)=(gauss_points_x(j));
        P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=(gauss_points_x(j));
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=(gauss_points_y(op));
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=(gauss_points_y(op));
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=(gauss_points_y(op));
P_non_derive_I_J_x_te(1,1)=1;
P_non_derive_I_J_x_te(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y_te(1,1)=1;
P_non_derive_I_J_y_te(1,2)=(gauss_points_y(op));
 P_non_derive_K_L_x_te(1,1)=1;
P_non_derive_K_L_x_te(1,2)=(gauss_points_x(j));
P_non_derive_K_L_y_te(1,1)=1;
P_non_derive_K_L_y_te(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x_te(1,1)=1;
P_non_derive_M_N_x_te(1,2)=gauss_points_x(j);
P_non_derive_M_N_y_te(1,1)=1;
P_non_derive_M_N_y_te(1,2)=gauss_points_y(op);
P_non_derive_P_Q_x_te(1,1)=1;
```

```
P_non_derive_P_Q_x_te(1,2)=(gauss_points_x(j));
P_non_derive_P_Q_y_te(1,1)=1;
P_non_derive_P_Q_y_te(1,2)=gauss_points_y(op);
P_non_derive_R_S_x_te(1,1)=1;
P_non_derive_R_S_x_te(1,2)=(gauss_points_y(op));
P_non_derive_R_S_y_te(1,1)=1;
P_non_derive_R_S_y_te(1,2)=(gauss_points_y(op));
for k=3:number_of_nodes_I_J
    P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2))));
end
for k=3:number_of_nodes_K_L
    P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
end
for k=3:number_of_nodes_M_N
     P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
end
 for k=3:number_of_nodes_P_Q
      P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
 end
  for k=3:number_of_nodes_R_S
       P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
  end
   for k=3:number_of_nodes_I_J
       P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
   end
   for k=3:number_of_nodes_K_L
       P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
   end
    for k=3:number_of_nodes_M_N
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
    for k=3:number_of_nodes_M_N
     P_non_derive_P_Q_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_P_Q_y(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y(1,k-2)));
end
     for k=3:number_of_nodes_R_S
```

```
         P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
     end
     for k=3:number_of_nodes_I_J_te_x
    P_non_derive_I_J_x_te(1,k)=((((2*k+1)/(k
+1))*(1)*P_non_derive_I_J_x_te(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x_te(1,k-2))));
end
for k=3:number_of_nodes_K_L_te_x
    P_non_derive_K_L_x_te(1,k)=((((2*k+1)/(k
+1))*(1))*P_non_derive_K_L_x_te(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x_te(1,k-2)));
end
for k=3:number_of_nodes_M_N_te_x
     P_non_derive_M_N_x_te(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_M_N_x_te(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x_te(1,k-2)));
end
 for k=3:number_of_nodes_P_Q_te_x
      P_non_derive_P_Q_x_te(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_P_Q_x_te(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x_te(1,k-2)));
 end
  for k=3:number_of_nodes_R_S_te_x
       P_non_derive_R_S_x_te(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_R_S_x_te(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x_te(1,k-2)));
  end
   for k=3:number_of_nodes_I_J_te_y
       P_non_derive_I_J_y_te(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y_te(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y_te(1,k-2)));
   end
   for k=3:number_of_nodes_K_L_te_y
       P_non_derive_K_L_y_te(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y_te(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y_te(1,k-2)));
   end
    for k=3:number_of_nodes_M_N_te_y
        P_non_derive_M_N_y_te(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y_te(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y_te(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y_te(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_P_Q_y_te(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y_te(1,k-2)));
    end
     for k=3:number_of_nodes_R_S_te_y
         P_non_derive_R_S_y_te(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y_te(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y_te(1,k-2)));
     end
```

```
     final_IJ=P_non_derive_I_J_x.*P_non_derive_I_J_y;
        final_KL=P_non_derive_K_L_x.*P_non_derive_K_L_y;
        final_MN=P_non_derive_M_N_x.*P_non_derive_M_N_y;
        final_PQ=P_non_derive_P_Q_x.*P_non_derive_P_Q_y;
        final_RS=P_non_derive_R_S_x.*P_non_derive_R_S_y;
        final_IJ_le=P_non_derive_I_J_x_te.*P_non_derive_I_J_y_te;
        final_KL_le=P_non_derive_K_L_x_te.*P_non_derive_K_L_y_te;
        final_MN_le=P_non_derive_M_N_x_te.*P_non_derive_M_N_y_te;
        final_PQ_le=P_non_derive_P_Q_x_te.*P_non_derive_P_Q_y_te;
        final_RS_le= P_non_derive_R_S_x_te.*P_non_derive_R_S_y_te;
        K_33_alpha_te_trans=K_33_alpha_te_trans+(sin(0)*(x_3-
x_2)+cos(0)*((y_3-
y_2))/4)*(final_IJ_le.*final_KL_le.*final_PQ_le.*final_MN_le.*final_RS_le)'*(final
    end
end
```

```
Not enough input arguments.

Error in Artificial_stiffness_trailing_edge_K_33 (line 14)
 sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 K_44=Artificial_stiffness_K_11(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span_of_the_entire
%  I=4;
%  J=4;
%  K=4;
%   L=4;
%  M=4;
%   N=4;
%  P=4;
%  Q=4;
%   R=4;
%   S=4;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
%  root_of_chord=5;
%  sweep_angle_in_radians=sweep_angle*pi/180;
%  tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%   gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
%  gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%  gauss_points_trans_x=0;
%  gauss_points_trans_y=0;
%  weights_trans_x=2;
%  weights_trans_y=2;
%   weights_x=[1 1];
%  weights_y=[1 1];
K_11_1=Artificial_stiffness_leading_edge_K_44(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span
K_11_2=Artificial_stiffness_root_K_44(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span_of_the_
K_11_3=Artificial_stiffness_tip_K_44(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span_of_the_e
K_11_4=Artificial_stiffness_trailing_edge_K_44(I,J,K,L,M,N,P,Q,R,S,sweep_angle,spa
K_44=K_11_1+K_11_2+K_11_3+K_11_4;
end
```

```
Not enough input arguments.

Error in Artificial_stiffness_K_44 (line 25)
K_11_1=Artificial_stiffness_leading_edge_K_44(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span
```

Published with MATLAB® R2020a

```
function
 K_44_alpha_le_trans=Artificial_stiffness_leading_edge(I,J,K,L,M,N,P,Q,R,S,sweep_a
%  I=2;
% J=2;
% K=2;
%  L=2;
%  M=2;
%  N=2;
%  P=2;
%  Q=2;
%  R=2;
%  S=2;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
 sweep_angle_in_radians=sweep_angle*pi/180;
%   root_of_chord=5;
 tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%   gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%   weights_x=[1 1];
%  weights_y=[1 1];
 u_o=zeros(I,1);
 P_non_derive_I_J_x=zeros(1,I*J);
P_non_derive_K_L_x=zeros(1,K*L);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_non_derive_K_L_y=zeros(1,K*L);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_non_derive_I_J_x_le=zeros(1,I*J);
P_non_derive_K_L_x_le=zeros(1,K*L);
P_non_derive_M_N_x_le=zeros(1,M*N);
P_non_derive_P_Q_x_le=zeros(1,P*Q);
P_non_derive_R_S_x_le=zeros(1,R*S);
P_non_derive_I_J_y_le=zeros(1,I*J);
P_non_derive_K_L_y_le=zeros(1,K*L);
P_non_derive_M_N_y_le=zeros(1,M*N);
P_non_derive_P_Q_y_le=zeros(1,P*Q);
 P_non_derive_R_S_y_le=zeros(1,R*S);
number_of_nodes_I_J_le_x=I*J;
number_of_nodes_K_L_le_x=K*L;
number_of_nodes_M_N_le_x=M*N;
number_of_nodes_P_Q_le_x=P*Q;
number_of_nodes_R_S_le_x=R*S;
number_of_nodes_I_J_le_y=I*J;
number_of_nodes_K_L_le_y=K*L;
number_of_nodes_M_N_le_y=M*N;
number_of_nodes_R_S_le_y=R*S;
```

```
 number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
 number_of_nodes_I_J_le=((I*J));
number_of_nodes_K_L_le=((K*L));
number_of_nodes_M_N_le=((M*N));
number_of_nodes_P_Q_le=((P*Q));
number_of_nodes_R_S_le=((R*S));
coordinates_x=[0 root_of_chord root_of_chord root_of_chord-
tip_chord]';
coordinates_y=[0 0 span_of_the_entire_wing/2
 span_of_the_entire_wing/2]';
x_2=coordinates_x(2);
y_2=coordinates_y(2);
x_3=coordinates_x(3);
y_3=coordinates_y(3);
K_44_alpha_le_trans=zeros(1,I*J);
for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
        P_non_derive_I_J_x(1,1)=1;
        P_non_derive_I_J_x(1,2)=(gauss_points_x(j));
        P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=(gauss_points_x(j));
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=(gauss_points_y(op));
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=(gauss_points_y(op));
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=(gauss_points_y(op));
P_non_derive_I_J_x_le(1,1)=1;
P_non_derive_I_J_x_le(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y_le(1,1)=1;
P_non_derive_I_J_y_le(1,2)=(gauss_points_y(op));
 P_non_derive_K_L_x_le(1,1)=1;
P_non_derive_K_L_x_le(1,2)=(gauss_points_x(j));
P_non_derive_K_L_y_le(1,1)=1;
P_non_derive_K_L_y_le(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x_le(1,1)=1;
P_non_derive_M_N_x_le(1,2)=gauss_points_x(j);
P_non_derive_M_N_y_le(1,1)=1;
P_non_derive_M_N_y_le(1,2)=gauss_points_y(op);
P_non_derive_P_Q_x_le(1,1)=1;
```

```
P_non_derive_P_Q_x_le(1,2)=(gauss_points_x(j));
P_non_derive_P_Q_y_le(1,1)=1;
P_non_derive_P_Q_y_le(1,2)=gauss_points_y(op);
P_non_derive_R_S_x_le(1,1)=1;
P_non_derive_R_S_x_le(1,2)=(gauss_points_y(op));
P_non_derive_R_S_y_le(1,1)=1;
P_non_derive_R_S_y_le(1,2)=(gauss_points_y(op));
for k=3:number_of_nodes_I_J
    P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2))));
end
for k=3:number_of_nodes_K_L
    P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
end
for k=3:number_of_nodes_M_N
     P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
end
 for k=3:number_of_nodes_P_Q
      P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
 end
  for k=3:number_of_nodes_R_S
       P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
  end
   for k=3:number_of_nodes_I_J
       P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
   end
   for k=3:number_of_nodes_K_L
       P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
   end
    for k=3:number_of_nodes_M_N
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
     for k=3:number_of_nodes_M_N
        P_non_derive_P_Q_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_P_Q_y(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y(1,k-2)));
    end
     for k=3:number_of_nodes_R_S
```

```
         P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
     end
     for k=3:number_of_nodes_I_J_le_x
    P_non_derive_I_J_x_le(1,k)=((((2*k+1)/(k
+1))*(-1)*P_non_derive_I_J_x_le(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x_le(1,k-2))));
end
for k=3:number_of_nodes_K_L_le_x
    P_non_derive_K_L_x_le(1,k)=((((2*k+1)/(k
+1))*(-1))*P_non_derive_K_L_x_le(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x_le(1,k-2)));
end
for k=3:number_of_nodes_M_N_le_x
     P_non_derive_M_N_x_le(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_M_N_x_le(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x_le(1,k-2)));
end
 for k=3:number_of_nodes_P_Q_le_x
      P_non_derive_P_Q_x_le(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_P_Q_x_le(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x_le(1,k-2)));
 end
  for k=3:number_of_nodes_R_S_le_x
       P_non_derive_R_S_x_le(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_R_S_x_le(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x_le(1,k-2)));
  end
   for k=3:number_of_nodes_I_J_le_y
       P_non_derive_I_J_y_le(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y_le(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y_le(1,k-2)));
   end
   for k=3:number_of_nodes_K_L_le_y
       P_non_derive_K_L_y_le(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y_le(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y_le(1,k-2)));
   end
    for k=3:number_of_nodes_M_N_le_y
        P_non_derive_M_N_y_le(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y_le(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y_le(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y_le(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_P_Q_y_le(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y_le(1,k-2)));
    end
     for k=3:number_of_nodes_R_S_le_y
         P_non_derive_R_S_y_le(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y_le(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y_le(1,k-2)));
     end
```

```
     final_IJ=P_non_derive_I_J_x.*P_non_derive_I_J_y;
        final_KL=P_non_derive_K_L_x.*P_non_derive_K_L_y;
        final_MN=P_non_derive_M_N_x.*P_non_derive_M_N_y;
        final_PQ=P_non_derive_P_Q_x.*P_non_derive_P_Q_y;
        final_RS=P_non_derive_R_S_x.*P_non_derive_R_S_y;
        final_IJ_le=P_non_derive_I_J_x_le.*P_non_derive_I_J_y_le;
        final_KL_le=P_non_derive_K_L_x_le.*P_non_derive_K_L_y_le;
        final_MN_le=P_non_derive_M_N_x_le.*P_non_derive_M_N_y_le;
        final_PQ_le=P_non_derive_P_Q_x_le.*P_non_derive_P_Q_y_le;
        final_RS_le= P_non_derive_R_S_x_le.*P_non_derive_R_S_y_le;
        K_44_alpha_le_trans=K_44_alpha_le_trans
+(sin(sweep_angle_in_radians)*abs(x_2-
x_3)+cos(sweep_angle_in_radians)*(abs(y_2-
y_3))/4)*(final_IJ_le.*final_KL_le.*final_PQ_le.*final_MN_le.*final_RS_le)'*(final
    end
end
```

```
Not enough input arguments.

Error in Artificial_stiffness_leading_edge_K_44 (line 14)
 sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 K_33_alpha_root_trans=Artificial_stiffness_root(I,J,K,L,M,N,P,Q,R,S,sweep_angle,s
%  I=2;
% J=2;
% K=2;
%  L=2;
%  M=2;
%  N=2;
%  P=2;
%  Q=2;
%  R=2;
%  S=2;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
 sweep_angle_in_radians=sweep_angle*pi/180;
%  root_of_chord=5;
 tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%  gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%  weights_x=[1 1];
% weights_y=[1 1];
 u_o=zeros(I,1);
 P_non_derive_I_J_x=zeros(1,I*J);
P_non_derive_K_L_x=zeros(1,K*L);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_non_derive_K_L_y=zeros(1,K*L);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_non_derive_I_J_x_root=zeros(1,I*J);
P_non_derive_K_L_x_root=zeros(1,K*L);
P_non_derive_M_N_x_root=zeros(1,M*N);
P_non_derive_P_Q_x_root=zeros(1,P*Q);
P_non_derive_R_S_x_root=zeros(1,R*S);
P_non_derive_I_J_y_root=zeros(1,I*J);
P_non_derive_K_L_y_root=zeros(1,K*L);
P_non_derive_M_N_y_root=zeros(1,M*N);
P_non_derive_P_Q_y_root=zeros(1,P*Q);
 P_non_derive_R_S_y_root=zeros(1,R*S);
number_of_nodes_I_J_le_x=I*J;
number_of_nodes_K_L_le_x=K*L;
number_of_nodes_M_N_le_x=M*N;
number_of_nodes_P_Q_le_x=P*Q;
number_of_nodes_R_S_le_x=R*S;
number_of_nodes_I_J_le_y=I*J;
number_of_nodes_K_L_le_y=K*L;
number_of_nodes_M_N_le_y=M*N;
number_of_nodes_R_S_le_y=R*S;
```

```
 number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
 number_of_nodes_I_J_le=((I*J));
number_of_nodes_K_L_le=((K*L));
number_of_nodes_M_N_le=((M*N));
number_of_nodes_P_Q_le=((P*Q));
number_of_nodes_R_S_le=((R*S));
coordinates_x=[0 root_of_chord root_of_chord root_of_chord-
tip_chord]';
coordinates_y=[0 0 span_of_the_entire_wing/2
 span_of_the_entire_wing/2]';
x_2=coordinates_x(2);
y_2=coordinates_y(2);
x_3=coordinates_x(3);
y_3=coordinates_y(3);
K_33_alpha_root_trans=zeros(1,I*J);
for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
        P_non_derive_I_J_x(1,1)=1;
        P_non_derive_I_J_x(1,2)=(gauss_points_x(j));
        P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=(gauss_points_x(j));
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=(gauss_points_y(op));
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=(gauss_points_y(op));
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=(gauss_points_y(op));
P_non_derive_I_J_x_root(1,1)=1;
P_non_derive_I_J_x_root(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y_root(1,1)=1;
P_non_derive_I_J_y_root(1,2)=(gauss_points_y(op));
 P_non_derive_K_L_x_root(1,1)=1;
P_non_derive_K_L_x_root(1,2)=(gauss_points_x(j));
P_non_derive_K_L_y_root(1,1)=1;
P_non_derive_K_L_y_root(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x_root(1,1)=1;
P_non_derive_M_N_x_root(1,2)=gauss_points_x(j);
P_non_derive_M_N_y_root(1,1)=1;
P_non_derive_M_N_y_root(1,2)=gauss_points_y(op);
P_non_derive_P_Q_x_root(1,1)=1;
```

```
P_non_derive_P_Q_x_root(1,2)=(gauss_points_x(j));
P_non_derive_P_Q_y_root(1,1)=1;
P_non_derive_P_Q_y_root(1,2)=gauss_points_y(op);
P_non_derive_R_S_x_root(1,1)=1;
P_non_derive_R_S_x_root(1,2)=(gauss_points_y(op));
P_non_derive_R_S_y_root(1,1)=1;
P_non_derive_R_S_y_root(1,2)=(gauss_points_y(op));
for k=3:number_of_nodes_I_J
    P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2))));
end
for k=3:number_of_nodes_K_L
    P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
end
for k=3:number_of_nodes_M_N
     P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
end
 for k=3:number_of_nodes_P_Q
      P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
 end
  for k=3:number_of_nodes_R_S
       P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
  end
   for k=3:number_of_nodes_I_J
       P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
   end
   for k=3:number_of_nodes_K_L
       P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
   end
    for k=3:number_of_nodes_M_N
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
     for k=3:number_of_nodes_R_S
         P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
     end
     for k=3:number_of_nodes_I_J_le_x
```

```
    P_non_derive_I_J_x_root(1,k)=((((2*k+1)/(k
+1))*(-1)*P_non_derive_I_J_x_root(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x_root(1,k-2))));
end
for k=3:number_of_nodes_K_L_le_x
    P_non_derive_K_L_x_root(1,k)=((((2*k+1)/(k
+1))*(-1))*P_non_derive_K_L_x_root(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x_root(1,k-2)));
end
for k=3:number_of_nodes_M_N_le_x
     P_non_derive_M_N_x_root(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_M_N_x_root(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x_root(1,k-2)));
end
 for k=3:number_of_nodes_P_Q_le_x
      P_non_derive_P_Q_x_root(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_P_Q_x_root(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x_root(1,k-2)));
 end
  for k=3:number_of_nodes_R_S_le_x
       P_non_derive_R_S_x_root(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_R_S_x_root(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x_root(1,k-2)));
  end
   for k=3:number_of_nodes_I_J_le_y
       P_non_derive_I_J_y_root(1,k)=((((2*k+1)/
(k+1))*(-1))*P_non_derive_I_J_y_root(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y_root(1,k-2)));
   end
   for k=3:number_of_nodes_K_L_le_y
       P_non_derive_K_L_y_root(1,k)=((((2*k+1)/
(k+1))*(-1))*P_non_derive_K_L_y_root(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y_root(1,k-2)));
   end
    for k=3:number_of_nodes_M_N_le_y
        P_non_derive_M_N_y_root(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_M_N_y_root(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y_root(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y_root(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_P_Q_y_root(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y_root(1,k-2)));
    end
     for k=3:number_of_nodes_R_S_le_y
         P_non_derive_R_S_y_root(1,k)=((((2*k+1)/
(k+1))*(-1))*P_non_derive_R_S_y_root(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y_root(1,k-2)));
     end
     final_IJ=P_non_derive_I_J_x'*P_non_derive_I_J_y;
        final_KL=P_non_derive_K_L_x'*P_non_derive_K_L_y;
        final_MN=P_non_derive_M_N_x'*P_non_derive_M_N_y;
        final_PQ=P_non_derive_P_Q_x'*P_non_derive_P_Q_y;
        final_RS=P_non_derive_R_S_x'*P_non_derive_R_S_y;
```

```
 final_IJ_root=P_non_derive_I_J_x_root'*P_non_derive_I_J_y_root;

 final_KL_root=P_non_derive_K_L_x_root'*P_non_derive_K_L_y_root;

 final_MN_root=P_non_derive_M_N_x_root'*P_non_derive_M_N_y_root;

 final_PQ_root=P_non_derive_P_Q_x_root'*P_non_derive_P_Q_y_root;
        final_RS_root=
 P_non_derive_R_S_x_root'*P_non_derive_R_S_y_root;
        K_33_alpha_root_trans=K_33_alpha_root_trans+(abs(x_2-
x_3)/4)*(final_PQ)'*(final_IJ_root.*final_KL_root.*final_PQ_root.*final_MN_root.*f
    end
end
```

```
Not enough input arguments.

Error in Artificial_stiffness_root_K_44 (line 14)
 sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 K_33_alpha_tip_trans=Artificial_stiffness_tip_K_11(I,J,K,L,M,N,P,Q,R,S,sweep_angl
%  I=4;
% J=4;
% K=4;
%  L=4;
%  M=4;
%  N=4;
%  P=4;
%  Q=4;
%  R=4;
%  S=4;
%  sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
 sweep_angle_in_radians=sweep_angle*pi/180;
%   root_of_chord=5;
 tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%  gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%  weights_x=[1 1];
% weights_y=[1 1];
%  u_o=zeros(I,1);
 P_non_derive_I_J_x=zeros(1,I*J);
P_non_derive_K_L_x=zeros(1,K*L);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_non_derive_K_L_y=zeros(1,K*L);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_non_derive_I_J_x_root=zeros(1,I*J);
P_non_derive_K_L_x_root=zeros(1,K*L);
P_non_derive_M_N_x_root=zeros(1,M*N);
P_non_derive_P_Q_x_root=zeros(1,P*Q);
P_non_derive_R_S_x_root=zeros(1,R*S);
P_non_derive_I_J_y_root=zeros(1,I*J);
P_non_derive_K_L_y_root=zeros(1,K*L);
P_non_derive_M_N_y_root=zeros(1,M*N);
P_non_derive_P_Q_y_root=zeros(1,P*Q);
 P_non_derive_R_S_y_root=zeros(1,R*S);
number_of_nodes_I_J_le_x=I*J;
number_of_nodes_K_L_le_x=K*L;
number_of_nodes_M_N_le_x=M*N;
number_of_nodes_P_Q_le_x=P*Q;
number_of_nodes_R_S_le_x=R*S;
number_of_nodes_I_J_le_y=I*J;
number_of_nodes_K_L_le_y=K*L;
number_of_nodes_M_N_le_y=M*N;
number_of_nodes_R_S_le_y=R*S;
```

```
 number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
 number_of_nodes_I_J_le=((I*J));
number_of_nodes_K_L_le=((K*L));
number_of_nodes_M_N_le=((M*N));
number_of_nodes_P_Q_le=((P*Q));
number_of_nodes_R_S_le=((R*S));
coordinates_x=[0 root_of_chord root_of_chord root_of_chord-
tip_chord]';
coordinates_y=[0 0 span_of_the_entire_wing/2
 span_of_the_entire_wing/2]';
x_3=coordinates_x(3);
y_3=coordinates_y(3);
x_4=coordinates_x(4);
y_4=coordinates_y(4);
K_33_alpha_tip_trans=zeros(1,I*J);
for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
        P_non_derive_I_J_x(1,1)=1;
        P_non_derive_I_J_x(1,2)=(gauss_points_x(j));
        P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=(gauss_points_x(j));
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=(gauss_points_y(op));
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=(gauss_points_y(op));
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=(gauss_points_y(op));
P_non_derive_I_J_x_root(1,1)=1;
P_non_derive_I_J_x_root(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y_root(1,1)=1;
P_non_derive_I_J_y_root(1,2)=(gauss_points_y(op));
 P_non_derive_K_L_x_root(1,1)=1;
P_non_derive_K_L_x_root(1,2)=(gauss_points_x(j));
P_non_derive_K_L_y_root(1,1)=1;
P_non_derive_K_L_y_root(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x_root(1,1)=1;
P_non_derive_M_N_x_root(1,2)=gauss_points_x(j);
P_non_derive_M_N_y_root(1,1)=1;
P_non_derive_M_N_y_root(1,2)=gauss_points_y(op);
P_non_derive_P_Q_x_root(1,1)=1;
```

```
P_non_derive_P_Q_x_root(1,2)=(gauss_points_x(j));
P_non_derive_P_Q_y_root(1,1)=1;
P_non_derive_P_Q_y_root(1,2)=gauss_points_y(op);
P_non_derive_R_S_x_root(1,1)=1;
P_non_derive_R_S_x_root(1,2)=(gauss_points_y(op));
P_non_derive_R_S_y_root(1,1)=1;
P_non_derive_R_S_y_root(1,2)=(gauss_points_y(op));
for k=3:number_of_nodes_I_J
    P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2))));
end
for k=3:number_of_nodes_K_L
    P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
end
for k=3:number_of_nodes_M_N
     P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
end
 for k=3:number_of_nodes_P_Q
      P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
 end
  for k=3:number_of_nodes_R_S
       P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
  end
   for k=3:number_of_nodes_I_J
       P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
   end
   for k=3:number_of_nodes_K_L
       P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
   end
    for k=3:number_of_nodes_M_N
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
    for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_P_Q_y(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y(1,k-2)));
    end
     for k=3:number_of_nodes_R_S
```

```
         P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
     end
     for k=3:number_of_nodes_I_J_le_x
    P_non_derive_I_J_x_root(1,k)=((((2*k+1)/(k
+1))*(1)*P_non_derive_I_J_x_root(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x_root(1,k-2))));
end
for k=3:number_of_nodes_K_L_le_x
    P_non_derive_K_L_x_root(1,k)=((((2*k+1)/(k
+1))*(1))*P_non_derive_K_L_x_root(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x_root(1,k-2)));
end
for k=3:number_of_nodes_M_N_le_x
     P_non_derive_M_N_x_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_M_N_x_root(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x_root(1,k-2)));
end
 for k=3:number_of_nodes_P_Q_le_x
      P_non_derive_P_Q_x_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_P_Q_x_root(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x_root(1,k-2)));
 end
  for k=3:number_of_nodes_R_S_le_x
       P_non_derive_R_S_x_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_R_S_x_root(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x_root(1,k-2)));
  end
   for k=3:number_of_nodes_I_J_le_y
       P_non_derive_I_J_y_root(1,k)=((((2*k+1)/
(k+1))*(1))*P_non_derive_I_J_y_root(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y_root(1,k-2)));
   end
   for k=3:number_of_nodes_K_L_le_y
       P_non_derive_K_L_y_root(1,k)=((((2*k+1)/
(k+1))*(1))*P_non_derive_K_L_y_root(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y_root(1,k-2)));
   end
    for k=3:number_of_nodes_M_N_le_y
        P_non_derive_M_N_y_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_M_N_y_root(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y_root(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_P_Q_y_root(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y_root(1,k-2)));
    end
     for k=3:number_of_nodes_R_S_le_y
         P_non_derive_R_S_y_root(1,k)=((((2*k+1)/
(k+1))*(1))*P_non_derive_R_S_y_root(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y_root(1,k-2)));
     end
```

```
     final_IJ=P_non_derive_I_J_x.*P_non_derive_I_J_y;
        final_KL=P_non_derive_K_L_x.*P_non_derive_K_L_y;
        final_MN=P_non_derive_M_N_x.*P_non_derive_M_N_y;
        final_PQ=P_non_derive_P_Q_x.*P_non_derive_P_Q_y;
        final_RS=P_non_derive_R_S_x.*P_non_derive_R_S_y;

 final_IJ_root=P_non_derive_I_J_x_root.*P_non_derive_I_J_y_root;

 final_KL_root=P_non_derive_K_L_x_root.*P_non_derive_K_L_y_root;

 final_MN_root=P_non_derive_M_N_x_root.*P_non_derive_M_N_y_root;

 final_PQ_root=P_non_derive_P_Q_x_root.*P_non_derive_P_Q_y_root;
        final_RS_root=
 P_non_derive_R_S_x_root.*P_non_derive_R_S_y_root;
        K_33_alpha_tip_trans=K_33_alpha_tip_trans+((x_3-
x_4)/4)*(final_IJ_root.*final_KL_root.*final_PQ_root.*final_MN_root.*final_RS_root
    end
end
end
```

```
Not enough input arguments.

Error in Artificial_stiffness_tip_K_44 (line 14)
 sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 K_33_alpha_te_trans=Artificial_stiffness_trailing_edge_K_11(I,J,K,L,M,N,P,Q,R,S,s
%  I=4;
% J=4;
% K=4;
%  L=4;
%  M=4;
%  N=4;
%  P=4;
%  Q=4;
%  R=4;
%  S=4;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
 sweep_angle_in_radians=sweep_angle*pi/180;
%  root_of_chord=5;
 tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%  gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%  weights_x=[1 1];
% weights_y=[1 1];
 u_o=zeros(I,1);
 P_non_derive_I_J_x=zeros(1,I*J);
P_non_derive_K_L_x=zeros(1,K*L);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_non_derive_K_L_y=zeros(1,K*L);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_non_derive_I_J_x_te=zeros(1,I*J);
P_non_derive_K_L_x_te=zeros(1,K*L);
P_non_derive_M_N_x_te=zeros(1,M*N);
P_non_derive_P_Q_x_te=zeros(1,P*Q);
P_non_derive_R_S_x_te=zeros(1,R*S);
P_non_derive_I_J_y_te=zeros(1,I*J);
P_non_derive_K_L_y_te=zeros(1,K*L);
P_non_derive_M_N_y_te=zeros(1,M*N);
P_non_derive_P_Q_y_te=zeros(1,P*Q);
 P_non_derive_R_S_y_te=zeros(1,R*S);
number_of_nodes_I_J_te_x=I*J;
number_of_nodes_K_L_te_x=K*L;
number_of_nodes_M_N_te_x=M*N;
number_of_nodes_P_Q_te_x=P*Q;
number_of_nodes_R_S_te_x=R*S;
number_of_nodes_I_J_te_y=I*J;
number_of_nodes_K_L_te_y=K*L;
number_of_nodes_M_N_te_y=M*N;
number_of_nodes_R_S_te_y=R*S;
```

```
 number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
 number_of_nodes_I_J_le=((I*J));
number_of_nodes_K_L_le=((K*L));
number_of_nodes_M_N_le=((M*N));
number_of_nodes_P_Q_le=((P*Q));
number_of_nodes_R_S_le=((R*S));
coordinates_x=[0 root_of_chord root_of_chord root_of_chord-
tip_chord]';
coordinates_y=[0 0 span_of_the_entire_wing/2
 span_of_the_entire_wing/2]';
x_3=coordinates_x(3);
y_3=coordinates_y(3);
x_2=coordinates_x(2);
y_2=coordinates_y(2);
K_33_alpha_te_trans=zeros(1,I*J);
for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
        P_non_derive_I_J_x(1,1)=1;
        P_non_derive_I_J_x(1,2)=(gauss_points_x(j));
        P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=(gauss_points_x(j));
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=(gauss_points_y(op));
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=(gauss_points_y(op));
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=(gauss_points_y(op));
P_non_derive_I_J_x_te(1,1)=1;
P_non_derive_I_J_x_te(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y_te(1,1)=1;
P_non_derive_I_J_y_te(1,2)=(gauss_points_y(op));
 P_non_derive_K_L_x_te(1,1)=1;
P_non_derive_K_L_x_te(1,2)=(gauss_points_x(j));
P_non_derive_K_L_y_te(1,1)=1;
P_non_derive_K_L_y_te(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x_te(1,1)=1;
P_non_derive_M_N_x_te(1,2)=gauss_points_x(j);
P_non_derive_M_N_y_te(1,1)=1;
P_non_derive_M_N_y_te(1,2)=gauss_points_y(op);
P_non_derive_P_Q_x_te(1,1)=1;
```

```
P_non_derive_P_Q_x_te(1,2)=(gauss_points_x(j));
P_non_derive_P_Q_y_te(1,1)=1;
P_non_derive_P_Q_y_te(1,2)=gauss_points_y(op);
P_non_derive_R_S_x_te(1,1)=1;
P_non_derive_R_S_x_te(1,2)=(gauss_points_y(op));
P_non_derive_R_S_y_te(1,1)=1;
P_non_derive_R_S_y_te(1,2)=(gauss_points_y(op));
for k=3:number_of_nodes_I_J
    P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2))));
end
for k=3:number_of_nodes_K_L
    P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
end
for k=3:number_of_nodes_M_N
     P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
end
 for k=3:number_of_nodes_P_Q
      P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
 end
  for k=3:number_of_nodes_R_S
       P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
  end
   for k=3:number_of_nodes_I_J
       P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
   end
   for k=3:number_of_nodes_K_L
       P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
   end
    for k=3:number_of_nodes_M_N
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
    for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_P_Q_y(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y(1,k-2)));
    end
     for k=3:number_of_nodes_R_S
```

```
         P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
     end
     for k=3:number_of_nodes_I_J_te_x
    P_non_derive_I_J_x_te(1,k)=((((2*k+1)/(k
+1))*(1)*P_non_derive_I_J_x_te(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x_te(1,k-2))));
end
for k=3:number_of_nodes_K_L_te_x
    P_non_derive_K_L_x_te(1,k)=((((2*k+1)/(k
+1))*(1))*P_non_derive_K_L_x_te(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x_te(1,k-2)));
end
for k=3:number_of_nodes_M_N_te_x
     P_non_derive_M_N_x_te(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_M_N_x_te(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x_te(1,k-2)));
end
 for k=3:number_of_nodes_P_Q_te_x
      P_non_derive_P_Q_x_te(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_P_Q_x_te(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x_te(1,k-2)));
 end
  for k=3:number_of_nodes_R_S_te_x
       P_non_derive_R_S_x_te(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_R_S_x_te(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x_te(1,k-2)));
  end
   for k=3:number_of_nodes_I_J_te_y
       P_non_derive_I_J_y_te(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y_te(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y_te(1,k-2)));
   end
   for k=3:number_of_nodes_K_L_te_y
       P_non_derive_K_L_y_te(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y_te(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y_te(1,k-2)));
   end
    for k=3:number_of_nodes_M_N_te_y
        P_non_derive_M_N_y_te(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y_te(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y_te(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y_te(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_P_Q_y_te(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y_te(1,k-2)));
    end
     for k=3:number_of_nodes_R_S_te_y
         P_non_derive_R_S_y_te(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y_te(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y_te(1,k-2)));
     end
```

```
     final_IJ=P_non_derive_I_J_x.*P_non_derive_I_J_y;
        final_KL=P_non_derive_K_L_x.*P_non_derive_K_L_y;
        final_MN=P_non_derive_M_N_x.*P_non_derive_M_N_y;
        final_PQ=P_non_derive_P_Q_x.*P_non_derive_P_Q_y;
        final_RS=P_non_derive_R_S_x.*P_non_derive_R_S_y;
        final_IJ_le=P_non_derive_I_J_x_te.*P_non_derive_I_J_y_te;
        final_KL_le=P_non_derive_K_L_x_te.*P_non_derive_K_L_y_te;
        final_MN_le=P_non_derive_M_N_x_te.*P_non_derive_M_N_y_te;
        final_PQ_le=P_non_derive_P_Q_x_te.*P_non_derive_P_Q_y_te;
        final_RS_le= P_non_derive_R_S_x_te.*P_non_derive_R_S_y_te;
        K_33_alpha_te_trans=K_33_alpha_te_trans+(sin(0)*(x_3-
x_2)+cos(0)*((y_3-
y_2))/4)*(final_IJ_le.*final_KL_le.*final_PQ_le.*final_MN_le.*final_RS_le)'*(final
    end
end
end
```

```
Not enough input arguments.

Error in Artificial_stiffness_trailing_edge_K_44 (line 14)
 sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 K_55=Artificial_stiffness_K_11(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span_of_the_entire
%  I=4;
%  J=4;
%  K=4;
%   L=4;
%  M=4;
%   N=4;
%  P=4;
%  Q=4;
%   R=4;
%   S=4;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
%  root_of_chord=5;
%  %tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%   gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
%  gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%   weights_x=[1 1];
%  weights_y=[1 1];
%  gauss_points_trans_x=0;
%  gauss_points_trans_y=0;
%  weights_trans_x=2;
%  weights_trans_y=2;
K_11_1=Artificial_stiffness_leading_edge_K_55(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span
K_11_2=Artificial_stiffness_root_K_55(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span_of_the_
K_11_3=Artificial_stiffness_tip_K_55(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span_of_the_e
K_11_4=Artificial_stiffness_trailing_edge_K_55(I,J,K,L,M,N,P,Q,R,S,sweep_angle,spa
K_55=K_11_1+K_11_2+K_11_3+K_11_4;
end
```

```
Not enough input arguments.

Error in Artificial_stiffness_K_55 (line 24)
K_11_1=Artificial_stiffness_leading_edge_K_55(I,J,K,L,M,N,P,Q,R,S,sweep_angle,span
```

Published with MATLAB® R2020a

```
function
 K_55_alpha_le_trans=Artificial_stiffness_leading_edge(I,J,K,L,M,N,P,Q,R,S,sweep_a
%  I=4;
% J=4;
% K=4;
%  L=4;
%  M=4;
%  N=4;
%  P=4;
%  Q=4;
%  R=4;
%  S=4;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
 sweep_angle_in_radians=sweep_angle*pi/180;
%  root_of_chord=5;
 tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%  gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%  weights_x=[1 1];
% weights_y=[1 1];
 u_o=zeros(I,1);
 P_non_derive_I_J_x=zeros(1,I*J);
P_non_derive_K_L_x=zeros(1,K*L);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_non_derive_K_L_y=zeros(1,K*L);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_non_derive_I_J_x_le=zeros(1,I*J);
P_non_derive_K_L_x_le=zeros(1,K*L);
P_non_derive_M_N_x_le=zeros(1,M*N);
P_non_derive_P_Q_x_le=zeros(1,P*Q);
P_non_derive_R_S_x_le=zeros(1,R*S);
P_non_derive_I_J_y_le=zeros(1,I*J);
P_non_derive_K_L_y_le=zeros(1,K*L);
P_non_derive_M_N_y_le=zeros(1,M*N);
P_non_derive_P_Q_y_le=zeros(1,P*Q);
 P_non_derive_R_S_y_le=zeros(1,R*S);
number_of_nodes_I_J_le_x=I*J;
number_of_nodes_K_L_le_x=K*L;
number_of_nodes_M_N_le_x=M*N;
number_of_nodes_P_Q_le_x=P*Q;
number_of_nodes_R_S_le_x=R*S;
number_of_nodes_I_J_le_y=I*J;
number_of_nodes_K_L_le_y=K*L;
number_of_nodes_M_N_le_y=M*N;
number_of_nodes_R_S_le_y=R*S;
```

```
 number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
 number_of_nodes_I_J_le=((I*J));
number_of_nodes_K_L_le=((K*L));
number_of_nodes_M_N_le=((M*N));
number_of_nodes_P_Q_le=((P*Q));
number_of_nodes_R_S_le=((R*S));
coordinates_x=[0 root_of_chord root_of_chord root_of_chord-
tip_chord]';
coordinates_y=[0 0 span_of_the_entire_wing/2
 span_of_the_entire_wing/2]';
x_3=coordinates_x(3);
y_3=coordinates_y(3);
x_4=coordinates_x(4);
y_4=coordinates_y(4);
K_55_alpha_le_trans=zeros(1,I*J);
for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
        P_non_derive_I_J_x(1,1)=1;
        P_non_derive_I_J_x(1,2)=(gauss_points_x(j));
        P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=(gauss_points_x(j));
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=(gauss_points_y(op));
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=(gauss_points_y(op));
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=(gauss_points_y(op));
P_non_derive_I_J_x_le(1,1)=1;
P_non_derive_I_J_x_le(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y_le(1,1)=1;
P_non_derive_I_J_y_le(1,2)=(gauss_points_y(op));
 P_non_derive_K_L_x_le(1,1)=1;
P_non_derive_K_L_x_le(1,2)=(gauss_points_x(j));
P_non_derive_K_L_y_le(1,1)=1;
P_non_derive_K_L_y_le(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x_le(1,1)=1;
P_non_derive_M_N_x_le(1,2)=gauss_points_x(j);
P_non_derive_M_N_y_le(1,1)=1;
P_non_derive_M_N_y_le(1,2)=gauss_points_y(op);
P_non_derive_P_Q_x_le(1,1)=1;
```

```
P_non_derive_P_Q_x_le(1,2)=(gauss_points_x(j));
P_non_derive_P_Q_y_le(1,1)=1;
P_non_derive_P_Q_y_le(1,2)=gauss_points_y(op);
P_non_derive_R_S_x_le(1,1)=1;
P_non_derive_R_S_x_le(1,2)=(gauss_points_y(op));
P_non_derive_R_S_y_le(1,1)=1;
P_non_derive_R_S_y_le(1,2)=(gauss_points_y(op));
for k=3:number_of_nodes_I_J
    P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2))));
end
for k=3:number_of_nodes_K_L
    P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
end
for k=3:number_of_nodes_M_N
     P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
end
 for k=3:number_of_nodes_P_Q
      P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
 end
  for k=3:number_of_nodes_R_S
       P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
  end
   for k=3:number_of_nodes_I_J
       P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
   end
   for k=3:number_of_nodes_K_L
       P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
   end
    for k=3:number_of_nodes_M_N
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
    for k=3:number_of_nodes_P_Q
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
     for k=3:number_of_nodes_R_S
```

```
         P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
     end
     for k=3:number_of_nodes_I_J_le_x
    P_non_derive_I_J_x_le(1,k)=((((2*k+1)/(k
+1))*(-1)*P_non_derive_I_J_x_le(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x_le(1,k-2))));
end
for k=3:number_of_nodes_K_L_le_x
    P_non_derive_K_L_x_le(1,k)=((((2*k+1)/(k
+1))*(-1))*P_non_derive_K_L_x_le(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x_le(1,k-2)));
end
for k=3:number_of_nodes_M_N_le_x
     P_non_derive_M_N_x_le(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_M_N_x_le(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x_le(1,k-2)));
end
 for k=3:number_of_nodes_P_Q_le_x
      P_non_derive_P_Q_x_le(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_P_Q_x_le(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x_le(1,k-2)));
 end
  for k=3:number_of_nodes_R_S_le_x
       P_non_derive_R_S_x_le(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_R_S_x_le(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x_le(1,k-2)));
  end
   for k=3:number_of_nodes_I_J_le_y
       P_non_derive_I_J_y_le(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y_le(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y_le(1,k-2)));
   end
   for k=3:number_of_nodes_K_L_le_y
       P_non_derive_K_L_y_le(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y_le(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y_le(1,k-2)));
   end
    for k=3:number_of_nodes_M_N_le_y
        P_non_derive_M_N_y_le(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y_le(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y_le(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y_le(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_P_Q_y_le(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y_le(1,k-2)));
    end
     for k=3:number_of_nodes_R_S_le_y
         P_non_derive_R_S_y_le(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y_le(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y_le(1,k-2)));
     end
```

```
     final_IJ=P_non_derive_I_J_x.*P_non_derive_I_J_y;
        final_KL=P_non_derive_K_L_x.*P_non_derive_K_L_y;
        final_MN=P_non_derive_M_N_x.*P_non_derive_M_N_y;
        final_PQ=P_non_derive_P_Q_x.*P_non_derive_P_Q_y;
        final_RS=P_non_derive_R_S_x.*P_non_derive_R_S_y;
        final_IJ_le=P_non_derive_I_J_x_le.*P_non_derive_I_J_y_le;
        final_KL_le=P_non_derive_K_L_x_le.*P_non_derive_K_L_y_le;
        final_MN_le=P_non_derive_M_N_x_le.*P_non_derive_M_N_y_le;
        final_PQ_le=P_non_derive_P_Q_x_le.*P_non_derive_P_Q_y_le;
        final_RS_le= P_non_derive_R_S_x_le.*P_non_derive_R_S_y_le;
        K_55_alpha_le_trans=K_55_alpha_le_trans
+(sin(sweep_angle_in_radians)*(x_3-
x_4)+cos(sweep_angle_in_radians)*((y_3-
y_4))/4)*(final_IJ_le.*final_KL_le.*final_PQ_le.*final_MN_le.*final_RS_le)'*(final
    end
end
end
```

```
Not enough input arguments.

Error in Artificial_stiffness_leading_edge_K_55 (line 14)
 sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 K_55_alpha_root_trans=Artificial_stiffness_root(I,J,K,L,M,N,P,Q,R,S,sweep_angle,s
%  I=2;
% J=2;
% K=2;
%  L=2;
%  M=2;
%  N=2;
%  P=2;
%  Q=2;
%  R=2;
%  S=2;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
 sweep_angle_in_radians=sweep_angle*pi/180;
%  root_of_chord=5;
 tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%  gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%  weights_x=[1 1];
% weights_y=[1 1];
 u_o=zeros(I,1);
 P_non_derive_I_J_x=zeros(1,I*J);
P_non_derive_K_L_x=zeros(1,K*L);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_non_derive_K_L_y=zeros(1,K*L);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_non_derive_I_J_x_root=zeros(1,I*J);
P_non_derive_K_L_x_root=zeros(1,K*L);
P_non_derive_M_N_x_root=zeros(1,M*N);
P_non_derive_P_Q_x_root=zeros(1,P*Q);
P_non_derive_R_S_x_root=zeros(1,R*S);
P_non_derive_I_J_y_root=zeros(1,I*J);
P_non_derive_K_L_y_root=zeros(1,K*L);
P_non_derive_M_N_y_root=zeros(1,M*N);
P_non_derive_P_Q_y_root=zeros(1,P*Q);
 P_non_derive_R_S_y_root=zeros(1,R*S);
number_of_nodes_I_J_le_x=I*J;
number_of_nodes_K_L_le_x=K*L;
number_of_nodes_M_N_le_x=M*N;
number_of_nodes_P_Q_le_x=P*Q;
number_of_nodes_R_S_le_x=R*S;
number_of_nodes_I_J_le_y=I*J;
number_of_nodes_K_L_le_y=K*L;
number_of_nodes_M_N_le_y=M*N;
number_of_nodes_R_S_le_y=R*S;
```

```
 number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
 number_of_nodes_I_J_le=((I*J));
number_of_nodes_K_L_le=((K*L));
number_of_nodes_M_N_le=((M*N));
number_of_nodes_P_Q_le=((P*Q));
number_of_nodes_R_S_le=((R*S));
coordinates_x=[0 root_of_chord root_of_chord root_of_chord-
tip_chord]';
coordinates_y=[0 0 span_of_the_entire_wing/2
 span_of_the_entire_wing/2]';
x_1=coordinates_x(1);
y_1=coordinates_y(1);
x_4=coordinates_x(1);
y_1=coordinates_y(1);
K_55_alpha_root_trans=zeros(1,I*J);
for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
        P_non_derive_I_J_x(1,1)=1;
        P_non_derive_I_J_x(1,2)=(gauss_points_x(j));
        P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=(gauss_points_x(j));
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=(gauss_points_y(op));
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=(gauss_points_y(op));
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=(gauss_points_y(op));
P_non_derive_I_J_x_root(1,1)=1;
P_non_derive_I_J_x_root(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y_root(1,1)=1;
P_non_derive_I_J_y_root(1,2)=(gauss_points_y(op));
 P_non_derive_K_L_x_root(1,1)=1;
P_non_derive_K_L_x_root(1,2)=(gauss_points_x(j));
P_non_derive_K_L_y_root(1,1)=1;
P_non_derive_K_L_y_root(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x_root(1,1)=1;
P_non_derive_M_N_x_root(1,2)=gauss_points_x(j);
P_non_derive_M_N_y_root(1,1)=1;
P_non_derive_M_N_y_root(1,2)=gauss_points_y(op);
P_non_derive_P_Q_x_root(1,1)=1;
```

```
P_non_derive_P_Q_x_root(1,2)=(gauss_points_x(j));
P_non_derive_P_Q_y_root(1,1)=1;
P_non_derive_P_Q_y_root(1,2)=gauss_points_y(op);
P_non_derive_R_S_x_root(1,1)=1;
P_non_derive_R_S_x_root(1,2)=(gauss_points_y(op));
P_non_derive_R_S_y_root(1,1)=1;
P_non_derive_R_S_y_root(1,2)=(gauss_points_y(op));
for k=3:number_of_nodes_I_J
    P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2))));
end
for k=3:number_of_nodes_K_L
    P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
end
for k=3:number_of_nodes_M_N
     P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
end
 for k=3:number_of_nodes_P_Q
      P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
 end
  for k=3:number_of_nodes_R_S
       P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
  end
   for k=3:number_of_nodes_I_J
       P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
   end
   for k=3:number_of_nodes_K_L
       P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
   end
    for k=3:number_of_nodes_M_N
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
    for k=3:number_of_nodes_P_Q
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
     for k=3:number_of_nodes_R_S
```

```
        P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
     end
     for k=3:number_of_nodes_I_J_le_x
    P_non_derive_I_J_x_root(1,k)=((((2*k+1)/(k
+1))*(-1)*P_non_derive_I_J_x_root(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x_root(1,k-2))));
end
for k=3:number_of_nodes_K_L_le_x
    P_non_derive_K_L_x_root(1,k)=((((2*k+1)/(k
+1))*(-1))*P_non_derive_K_L_x_root(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x_root(1,k-2)));
end
for k=3:number_of_nodes_M_N_le_x
     P_non_derive_M_N_x_root(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_M_N_x_root(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x_root(1,k-2)));
end
 for k=3:number_of_nodes_P_Q_le_x
      P_non_derive_P_Q_x_root(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_P_Q_x_root(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x_root(1,k-2)));
 end
  for k=3:number_of_nodes_R_S_le_x
       P_non_derive_R_S_x_root(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_R_S_x_root(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x_root(1,k-2)));
  end
   for k=3:number_of_nodes_I_J_le_y
       P_non_derive_I_J_y_root(1,k)=((((2*k+1)/
(k+1))*(-1))*P_non_derive_I_J_y_root(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y_root(1,k-2)));
   end
   for k=3:number_of_nodes_K_L_le_y
       P_non_derive_K_L_y_root(1,k)=((((2*k+1)/
(k+1))*(-1))*P_non_derive_K_L_y_root(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y_root(1,k-2)));
   end
    for k=3:number_of_nodes_M_N_le_y
        P_non_derive_M_N_y_root(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_M_N_y_root(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y_root(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y_root(1,k)=((((2*k+1)/
(k+1))*-1)*P_non_derive_P_Q_y_root(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y_root(1,k-2)));
    end
     for k=3:number_of_nodes_R_S_le_y
         P_non_derive_R_S_y_root(1,k)=((((2*k+1)/
(k+1))*(-1))*P_non_derive_R_S_y_root(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y_root(1,k-2)));
     end
```

```
     final_IJ=P_non_derive_I_J_x'*P_non_derive_I_J_y;
        final_KL=P_non_derive_K_L_x'*P_non_derive_K_L_y;
        final_MN=P_non_derive_M_N_x'*P_non_derive_M_N_y;
        final_PQ=P_non_derive_P_Q_x'*P_non_derive_P_Q_y;
        final_RS=P_non_derive_R_S_x'*P_non_derive_R_S_y;

 final_IJ_root=P_non_derive_I_J_x_root'*P_non_derive_I_J_y_root;

 final_KL_root=P_non_derive_K_L_x_root'*P_non_derive_K_L_y_root;

 final_MN_root=P_non_derive_M_N_x_root'*P_non_derive_M_N_y_root;
        final_PQ_root=P_non_derive_P_Q_x_root'*P_non_derive_P_Q_y;
        final_RS_root=
 P_non_derive_R_S_x_root'*P_non_derive_R_S_y_root;
        K_55_alpha_root_trans=K_55_alpha_root_trans+((x_1-
x_4)/4)*(final_IJ_root.*final_KL_root.*final_PQ_root.*final_MN_root.*final_RS_root
    end
end
```

```
Not enough input arguments.

Error in Artificial_stiffness_root_K_55 (line 14)
 sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 K_55_alpha_tip_trans=Artificial_stiffness_tip_K_11(I,J,K,L,M,N,P,Q,R,S,sweep_angl
%  I=2;
% J=2;
% K=2;
%  L=2;
%  M=2;
%  N=2;
%  P=2;
%  Q=2;
%  R=2;
%  S=2;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
 sweep_angle_in_radians=sweep_angle*pi/180;
%  root_of_chord=5;
 tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%  gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%  weights_x=[1 1];
% weights_y=[1 1];
 u_o=zeros(I,1);
 P_non_derive_I_J_x=zeros(1,I*J);
P_non_derive_K_L_x=zeros(1,K*L);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_non_derive_K_L_y=zeros(1,K*L);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_non_derive_I_J_x_root=zeros(1,I*J);
P_non_derive_K_L_x_root=zeros(1,K*L);
P_non_derive_M_N_x_root=zeros(1,M*N);
P_non_derive_P_Q_x_root=zeros(1,P*Q);
P_non_derive_R_S_x_root=zeros(1,R*S);
P_non_derive_I_J_y_root=zeros(1,I*J);
P_non_derive_K_L_y_root=zeros(1,K*L);
P_non_derive_M_N_y_root=zeros(1,M*N);
P_non_derive_P_Q_y_root=zeros(1,P*Q);
 P_non_derive_R_S_y_root=zeros(1,R*S);
number_of_nodes_I_J_le_x=I*J;
number_of_nodes_K_L_le_x=K*L;
number_of_nodes_M_N_le_x=M*N;
number_of_nodes_P_Q_le_x=P*Q;
number_of_nodes_R_S_le_x=R*S;
number_of_nodes_I_J_le_y=I*J;
number_of_nodes_K_L_le_y=K*L;
number_of_nodes_M_N_le_y=M*N;
number_of_nodes_R_S_le_y=R*S;
```

```
 number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
 number_of_nodes_I_J_le=((I*J));
number_of_nodes_K_L_le=((K*L));
number_of_nodes_M_N_le=((M*N));
number_of_nodes_P_Q_le=((P*Q));
number_of_nodes_R_S_le=((R*S));
coordinates_x=[0 root_of_chord root_of_chord root_of_chord-
tip_chord]';
coordinates_y=[0 0 span_of_the_entire_wing/2
 span_of_the_entire_wing/2]';
x_3=coordinates_x(3);
y_3=coordinates_y(3);
x_4=coordinates_x(4);
y_4=coordinates_y(4);
K_55_alpha_tip_trans=zeros(1,I*J);
for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
        P_non_derive_I_J_x(1,1)=1;
        P_non_derive_I_J_x(1,2)=(gauss_points_x(j));
        P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=(gauss_points_x(j));
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=(gauss_points_y(op));
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=(gauss_points_y(op));
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=(gauss_points_y(op));
P_non_derive_I_J_x_root(1,1)=1;
P_non_derive_I_J_x_root(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y_root(1,1)=1;
P_non_derive_I_J_y_root(1,2)=(gauss_points_y(op));
 P_non_derive_K_L_x_root(1,1)=1;
P_non_derive_K_L_x_root(1,2)=(gauss_points_x(j));
P_non_derive_K_L_y_root(1,1)=1;
P_non_derive_K_L_y_root(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x_root(1,1)=1;
P_non_derive_M_N_x_root(1,2)=gauss_points_x(j);
P_non_derive_M_N_y_root(1,1)=1;
P_non_derive_M_N_y_root(1,2)=gauss_points_y(op);
P_non_derive_P_Q_x_root(1,1)=1;
```

```
P_non_derive_P_Q_x_root(1,2)=(gauss_points_x(j));
P_non_derive_P_Q_y_root(1,1)=1;
P_non_derive_P_Q_y_root(1,2)=gauss_points_y(op);
P_non_derive_R_S_x_root(1,1)=1;
P_non_derive_R_S_x_root(1,2)=(gauss_points_y(op));
P_non_derive_R_S_y_root(1,1)=1;
P_non_derive_R_S_y_root(1,2)=(gauss_points_y(op));
for k=3:number_of_nodes_I_J
    P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2))));
end
for k=3:number_of_nodes_K_L
    P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
end
for k=3:number_of_nodes_M_N
     P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
end
 for k=3:number_of_nodes_P_Q
      P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
 end
  for k=3:number_of_nodes_R_S
       P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
  end
   for k=3:number_of_nodes_I_J
       P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
   end
   for k=3:number_of_nodes_K_L
       P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
   end
    for k=3:number_of_nodes_M_N
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
    for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_P_Q_y(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y(1,k-2)));
    end
     for k=3:number_of_nodes_R_S
```

```
         P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
     end
     for k=3:number_of_nodes_I_J_le_x
    P_non_derive_I_J_x_root(1,k)=((((2*k+1)/(k
+1))*(1)*P_non_derive_I_J_x_root(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x_root(1,k-2))));
end
for k=3:number_of_nodes_K_L_le_x
    P_non_derive_K_L_x_root(1,k)=((((2*k+1)/(k
+1))*(1))*P_non_derive_K_L_x_root(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x_root(1,k-2)));
end
for k=3:number_of_nodes_M_N_le_x
     P_non_derive_M_N_x_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_M_N_x_root(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x_root(1,k-2)));
end
 for k=3:number_of_nodes_P_Q_le_x
      P_non_derive_P_Q_x_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_P_Q_x_root(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x_root(1,k-2)));
 end
  for k=3:number_of_nodes_R_S_le_x
       P_non_derive_R_S_x_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_R_S_x_root(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x_root(1,k-2)));
  end
   for k=3:number_of_nodes_I_J_le_y
       P_non_derive_I_J_y_root(1,k)=((((2*k+1)/
(k+1))*(1))*P_non_derive_I_J_y_root(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y_root(1,k-2)));
   end
   for k=3:number_of_nodes_K_L_le_y
       P_non_derive_K_L_y_root(1,k)=((((2*k+1)/
(k+1))*(1))*P_non_derive_K_L_y_root(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y_root(1,k-2)));
   end
    for k=3:number_of_nodes_M_N_le_y
        P_non_derive_M_N_y_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_M_N_y_root(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y_root(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y_root(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_P_Q_y_root(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y_root(1,k-2)));
    end
     for k=3:number_of_nodes_R_S_le_y
         P_non_derive_R_S_y_root(1,k)=((((2*k+1)/
(k+1))*(1))*P_non_derive_R_S_y_root(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y_root(1,k-2)));
     end
```

```
     final_IJ=P_non_derive_I_J_x'*P_non_derive_I_J_y;
        final_KL=P_non_derive_K_L_x'*P_non_derive_K_L_y;
        final_MN=P_non_derive_M_N_x'*P_non_derive_M_N_y;
        final_PQ=P_non_derive_P_Q_x'*P_non_derive_P_Q_y;
        final_RS=P_non_derive_R_S_x'*P_non_derive_R_S_y;

 final_IJ_root=P_non_derive_I_J_x_root'*P_non_derive_I_J_y_root;

 final_KL_root=P_non_derive_K_L_x_root'*P_non_derive_K_L_y_root;

 final_MN_root=P_non_derive_M_N_x_root'*P_non_derive_M_N_y_root;

 final_PQ_root=P_non_derive_P_Q_x_root'*P_non_derive_P_Q_y_root;
        final_RS_root=
 P_non_derive_R_S_x_root'*P_non_derive_R_S_y_root;
        K_55_alpha_tip_trans=K_55_alpha_tip_trans+((x_3-
x_4)/4)*(final_IJ_root.*final_KL_root.*final_PQ_root.*final_MN_root.*final_RS_root
    end
end
```

```
Not enough input arguments.

Error in Artificial_stiffness_tip_K_55 (line 14)
 sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a

```
function
 K_55_alpha_te_trans=Artificial_stiffness_trailing_edge_K_22(I,J,K,L,M,N,P,Q,R,S,s
%  I=2;
% J=2;
% K=2;
%  L=2;
%  M=2;
%  N=2;
%  P=2;
%  Q=2;
%  R=2;
%  S=2;
%   sweep_angle=30;%degree
%  span_of_the_entire_wing=4;
 sweep_angle_in_radians=sweep_angle*pi/180;
%  root_of_chord=5;
 tip_chord=root_of_chord-
((span_of_the_entire_wing/2)*tan(sweep_angle_in_radians));
%  gauss_points_x=[-1/sqrt(3) 1/sqrt(3)];
% gauss_points_y=[-1/sqrt(3) 1/sqrt(3)];
%  weights_x=[1 1];
% weights_y=[1 1];
 u_o=zeros(I,1);
 P_non_derive_I_J_x=zeros(1,I*J);
P_non_derive_K_L_x=zeros(1,K*L);
P_non_derive_M_N_x=zeros(1,M*N);
P_non_derive_P_Q_x=zeros(1,P*Q);
P_non_derive_R_S_x=zeros(1,R*S);
P_non_derive_I_J_y=zeros(1,I*J);
P_non_derive_K_L_y=zeros(1,K*L);
P_non_derive_M_N_y=zeros(1,M*N);
P_non_derive_P_Q_y=zeros(1,P*Q);
P_non_derive_R_S_y=zeros(1,R*S);
P_non_derive_I_J_x_te=zeros(1,I*J);
P_non_derive_K_L_x_te=zeros(1,K*L);
P_non_derive_M_N_x_te=zeros(1,M*N);
P_non_derive_P_Q_x_te=zeros(1,P*Q);
P_non_derive_R_S_x_te=zeros(1,R*S);
P_non_derive_I_J_y_te=zeros(1,I*J);
P_non_derive_K_L_y_te=zeros(1,K*L);
P_non_derive_M_N_y_te=zeros(1,M*N);
P_non_derive_P_Q_y_te=zeros(1,P*Q);
 P_non_derive_R_S_y_te=zeros(1,R*S);
number_of_nodes_I_J_te_x=I*J;
number_of_nodes_K_L_te_x=K*L;
number_of_nodes_M_N_te_x=M*N;
number_of_nodes_P_Q_te_x=P*Q;
number_of_nodes_R_S_te_x=R*S;
number_of_nodes_I_J_te_y=I*J;
number_of_nodes_K_L_te_y=K*L;
number_of_nodes_M_N_te_y=M*N;
number_of_nodes_R_S_te_y=R*S;
```

```
 number_of_nodes_I_J=((I*J));
number_of_nodes_K_L=((K*L));
number_of_nodes_M_N=((M*N));
number_of_nodes_P_Q=((P*Q));
number_of_nodes_R_S=((R*S));
 number_of_nodes_I_J_le=((I*J));
number_of_nodes_K_L_le=((K*L));
number_of_nodes_M_N_le=((M*N));
number_of_nodes_P_Q_le=((P*Q));
number_of_nodes_R_S_le=((R*S));
coordinates_x=[0 root_of_chord root_of_chord root_of_chord-
tip_chord]';
coordinates_y=[0 0 span_of_the_entire_wing/2
 span_of_the_entire_wing/2]';
x_3=coordinates_x(3);
y_3=coordinates_y(3);
x_2=coordinates_x(2);
y_2=coordinates_y(2);
K_55_alpha_te_trans=zeros(1,I*J);
for op=1:length(gauss_points_y)
    for j=1:length(gauss_points_x)
        P_non_derive_I_J_x(1,1)=1;
        P_non_derive_I_J_x(1,2)=(gauss_points_x(j));
        P_non_derive_K_L_x(1,1)=1;
P_non_derive_K_L_x(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x(1,1)=1;
P_non_derive_M_N_x(1,2)=gauss_points_x(j);
P_non_derive_P_Q_x(1,1)=1;
P_non_derive_P_Q_x(1,2)=(gauss_points_x(j));
P_non_derive_R_S_x(1,1)=1;
P_non_derive_R_S_x(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y(1,1)=1;
P_non_derive_I_J_y(1,2)=(gauss_points_y(op));
P_non_derive_K_L_y(1,1)=1;
P_non_derive_K_L_y(1,2)=(gauss_points_y(op));
P_non_derive_M_N_y(1,1)=1;
P_non_derive_M_N_y(1,2)=gauss_points_y(op);
P_non_derive_P_Q_y(1,1)=1;
P_non_derive_P_Q_y(1,2)=gauss_points_y(op);
P_non_derive_R_S_y(1,1)=1;
P_non_derive_R_S_y(1,2)=(gauss_points_y(op));
P_non_derive_I_J_x_te(1,1)=1;
P_non_derive_I_J_x_te(1,2)=(gauss_points_x(j));
P_non_derive_I_J_y_te(1,1)=1;
P_non_derive_I_J_y_te(1,2)=(gauss_points_y(op));
 P_non_derive_K_L_x_te(1,1)=1;
P_non_derive_K_L_x_te(1,2)=(gauss_points_x(j));
P_non_derive_K_L_y_te(1,1)=1;
P_non_derive_K_L_y_te(1,2)=(gauss_points_x(j));
P_non_derive_M_N_x_te(1,1)=1;
P_non_derive_M_N_x_te(1,2)=gauss_points_x(j);
P_non_derive_M_N_y_te(1,1)=1;
P_non_derive_M_N_y_te(1,2)=gauss_points_y(op);
P_non_derive_P_Q_x_te(1,1)=1;
```

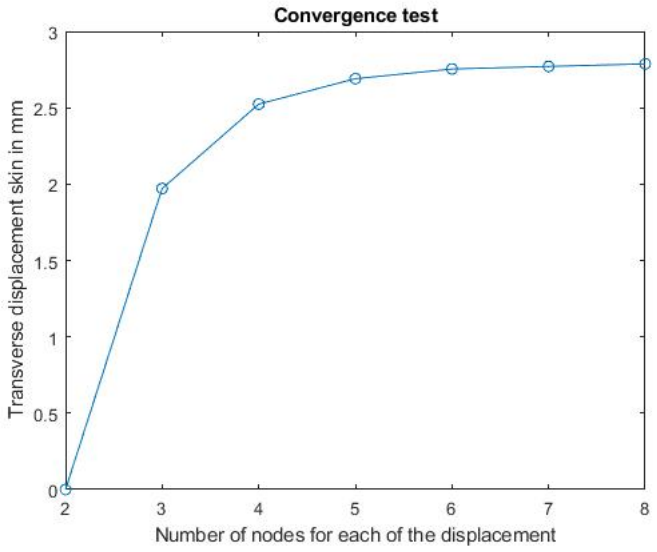
```
P_non_derive_P_Q_x_te(1,2)=(gauss_points_x(j));
P_non_derive_P_Q_y_te(1,1)=1;
P_non_derive_P_Q_y_te(1,2)=gauss_points_y(op);
P_non_derive_R_S_x_te(1,1)=1;
P_non_derive_R_S_x_te(1,2)=(gauss_points_y(op));
P_non_derive_R_S_y_te(1,1)=1;
P_non_derive_R_S_y_te(1,2)=(gauss_points_y(op));
for k=3:number_of_nodes_I_J
    P_non_derive_I_J_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j))*P_non_derive_I_J_x(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x(1,k-2))));
end
for k=3:number_of_nodes_K_L
    P_non_derive_K_L_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_K_L_x(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x(1,k-2)));
end
for k=3:number_of_nodes_M_N
     P_non_derive_M_N_x(1,k)=((((2*k+1)/(k
+1))*gauss_points_x(op))*P_non_derive_M_N_x(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x(1,k-2)));
end
 for k=3:number_of_nodes_P_Q
      P_non_derive_P_Q_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_P_Q_x(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x(1,k-2)));
 end
  for k=3:number_of_nodes_R_S
       P_non_derive_R_S_x(1,k)=((((2*k+1)/(k
+1))*(gauss_points_x(j)))*P_non_derive_R_S_x(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x(1,k-2)));
  end
   for k=3:number_of_nodes_I_J
       P_non_derive_I_J_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y(1,k-2)));
   end
   for k=3:number_of_nodes_K_L
       P_non_derive_K_L_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y(1,k-2)));
   end
    for k=3:number_of_nodes_M_N
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
    for k=3:number_of_nodes_P_Q
        P_non_derive_M_N_y(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y(1,k-2)));
    end
     for k=3:number_of_nodes_R_S
```

```
         P_non_derive_R_S_y(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y(1,k-2)));
     end
     for k=3:number_of_nodes_I_J_te_x
    P_non_derive_I_J_x_te(1,k)=((((2*k+1)/(k
+1))*(1)*P_non_derive_I_J_x_te(1,k-1)-(k/(k
+1)*P_non_derive_I_J_x_te(1,k-2))));
end
for k=3:number_of_nodes_K_L_te_x
    P_non_derive_K_L_x_te(1,k)=((((2*k+1)/(k
+1))*(1))*P_non_derive_K_L_x_te(1,k-1)-(k/(k
+1)*P_non_derive_K_L_x_te(1,k-2)));
end
for k=3:number_of_nodes_M_N_te_x
     P_non_derive_M_N_x_te(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_M_N_x_te(1,k-1)-(k/(k
+1)*P_non_derive_M_N_x_te(1,k-2)));
end
 for k=3:number_of_nodes_P_Q_te_x
      P_non_derive_P_Q_x_te(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_P_Q_x_te(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_x_te(1,k-2)));
 end
  for k=3:number_of_nodes_R_S_te_x
       P_non_derive_R_S_x_te(1,k)=((((2*k+1)/
(k+1))*1)*P_non_derive_R_S_x_te(1,k-1)-(k/(k
+1)*P_non_derive_R_S_x_te(1,k-2)));
  end
   for k=3:number_of_nodes_I_J_te_y
       P_non_derive_I_J_y_te(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_I_J_y_te(1,k-1)-(k/(k
+1)*P_non_derive_I_J_y_te(1,k-2)));
   end
   for k=3:number_of_nodes_K_L_te_y
       P_non_derive_K_L_y_te(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_K_L_y_te(1,k-1)-(k/(k
+1)*P_non_derive_K_L_y_te(1,k-2)));
   end
    for k=3:number_of_nodes_M_N_te_y
        P_non_derive_M_N_y_te(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_M_N_y_te(1,k-1)-(k/(k
+1)*P_non_derive_M_N_y_te(1,k-2)));
    end
     for k=3:number_of_nodes_P_Q
        P_non_derive_P_Q_y_te(1,k)=((((2*k+1)/(k
+1))*gauss_points_y(op))*P_non_derive_P_Q_y_te(1,k-1)-(k/(k
+1)*P_non_derive_P_Q_y_te(1,k-2)));
    end
     for k=3:number_of_nodes_R_S_te_y
         P_non_derive_R_S_y_te(1,k)=((((2*k+1)/(k
+1))*(gauss_points_y(op)))*P_non_derive_R_S_y_te(1,k-1)-(k/(k
+1)*P_non_derive_R_S_y_te(1,k-2)));
     end
```

```
     final_IJ=P_non_derive_I_J_x.*P_non_derive_I_J_y;
        final_KL=P_non_derive_K_L_x.*P_non_derive_K_L_y;
        final_MN=P_non_derive_M_N_x.*P_non_derive_M_N_y;
        final_PQ=P_non_derive_P_Q_x.*P_non_derive_P_Q_y;
        final_RS=P_non_derive_R_S_x.*P_non_derive_R_S_y;
        final_IJ_le=P_non_derive_I_J_x_te.*P_non_derive_I_J_y_te;
        final_KL_le=P_non_derive_K_L_x_te.*P_non_derive_K_L_y_te;
        final_MN_le=P_non_derive_M_N_x_te.*P_non_derive_M_N_y_te;
        final_PQ_le=P_non_derive_P_Q_x_te.*P_non_derive_P_Q_y_te;
        final_RS_le= P_non_derive_R_S_x_te.*P_non_derive_R_S_y_te;
        K_55_alpha_te_trans=K_55_alpha_te_trans+(sin(0)*(x_3-
x_2)+cos(0)*((y_3-
y_2))/4)*(final_IJ_le.*final_KL_le.*final_PQ_le.*final_MN_le.*final_RS_le)'*(final
    end
end
```

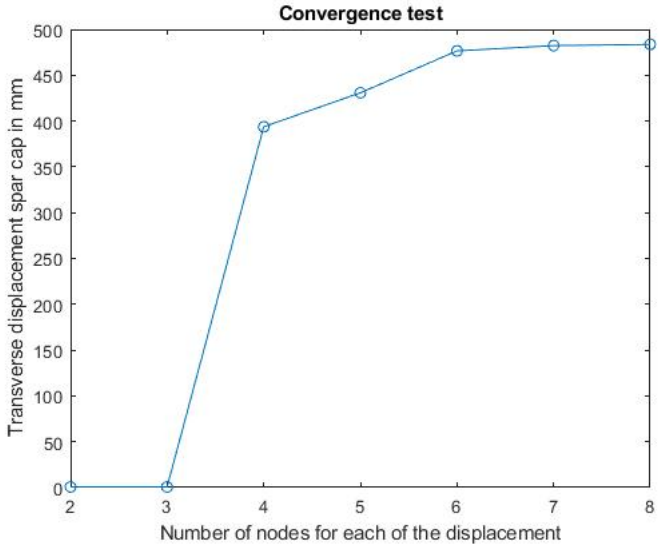
```
Not enough input arguments.

Error in Artificial_stiffness_trailing_edge_K_55 (line 14)
 sweep_angle_in_radians=sweep_angle*pi/180;
```

Published with MATLAB® R2020a



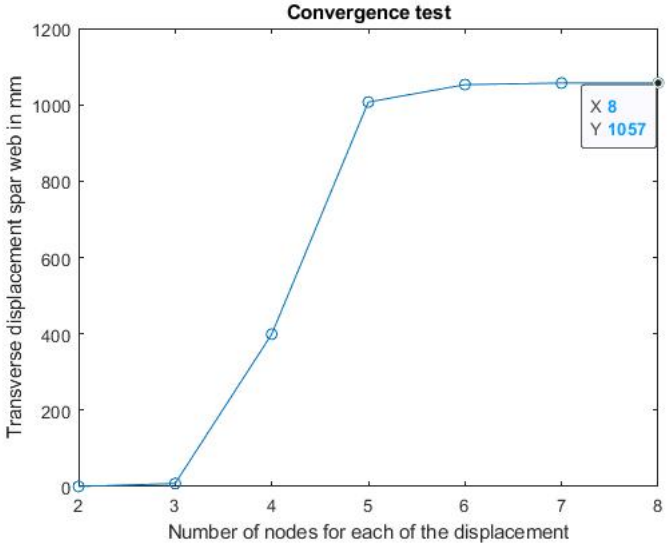

```
>> tolerance=(Final_converge(end)-Final_converge(end-1))/Final_converge(end)

tolerance =

    0.0060
```

Figure 6: Convergence maximum test for skin




```
>> tolerance=(Final_converge(end)-Final_converge(end-1))/Final_converge(end)

tolerance =

    0.0027
```

Figure 7: Convergence test maximum for spar cap




```
>> tolerance=(Final_converge(end)-Final_converge(end-1))/Final_converge(end)

tolerance =

   1.4793e-04
```

Figure 8: Convergence test maximum for spar web

```
Elapsed time is 128.640937 seconds.
```

Figure 9: Elapsed time for the code to run