Design Project

MAE 526: Design of Aerospace Structures

Dr. Aditi Chattopadhyay

Group 2:

Aishwarya Ledalla

Alexander Scalco

Lakshya Tiwari

Rohan Ravishekar

May 1, 2022

1. Executive Summary

In a fixed-wing aircraft, the spar is often the main structural member of the wing, running spanwise, perpendicular to the fuselage. The purpose of the spar is to carry the weight of the wings while on the ground as well as the associated flight loads while in air. The objective of this project was to design a wing that will support the given loading conditions with minimal weight. To accomplish this goal, the wing structure was assumed to be a box beam and trade-off-studies were conducted between a stiffened thin-walled box beam constructed out of metal and a box beam constructed out of composite. The design of the metal beam included trade-off-studies between varying materials, number of stiffeners, stiffener cross section and dimensions, and skin thicknesses. Furthermore, the design of the composite beam included trade-off-studies between varying fiber volume fractions, box heights, stacking sequences and the number of plies. To ensure full structural safety for each wing design, the maximum stress failure criterion and Tsai-Hill criterion were considered respectively.

2. Recommended Design and Estimated Weight

<u>Metal Design</u>

The purpose of this project was to identify a minimum weight design configuration; however, it was not specified if it was per material, per stiffener cross section, or overall. For this reason, five recommended designs are presented below and have been organized per stiffener cross section. Designs one through three are based on the scope of the project and four contains additional studies.

1) L – Section Stiffener

While additional trade studies in other materials and stiffener cross sections yield in a lighter wing box. *Figure 1* depicts a minimum weight design given the materials presented in the project and only using L stiffeners. Using 7075-T6 aluminum, six L stiffeners and a skin thickness of 5.09 *mm* a wing box can be designed with a weight of 293.93 kg. Furthermore, the L stiffeners have the following dimensions: Height: 19.45 *mm*, Width: 19.45 *mm*, Thickness: 4.88 *mm*.

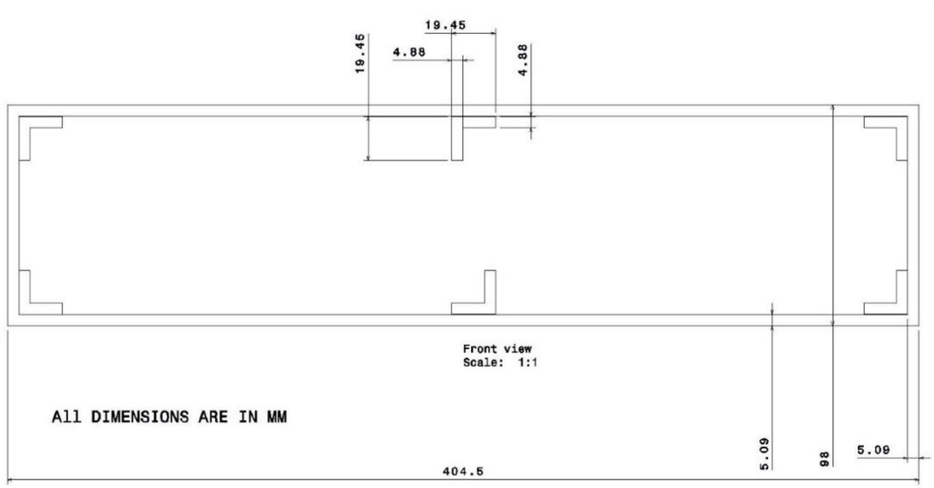


Figure 1: Minimum Weight Design for only L stiffeners using 7075-T6 Aluminum

2) I – Section Stiffener

Similarly, additional trade studies in other materials and stiffener cross sections yield in a lighter wing box. However, using only I section stiffeners and the given materials in the project, a significantly lighter wing design of 181.21 kg was produced. Shown in *figure 2* is the recommended metal design that would generate the minimum weight thin-walled box beam based on the parameters given in the project. This design requires six I stiffeners with a height of 27.37 *mm*, width of 40.66 *mm*, thickness of 8.27 *mm* as well as a skin thickness of 1 *mm* made from 7075-T6 aluminum.

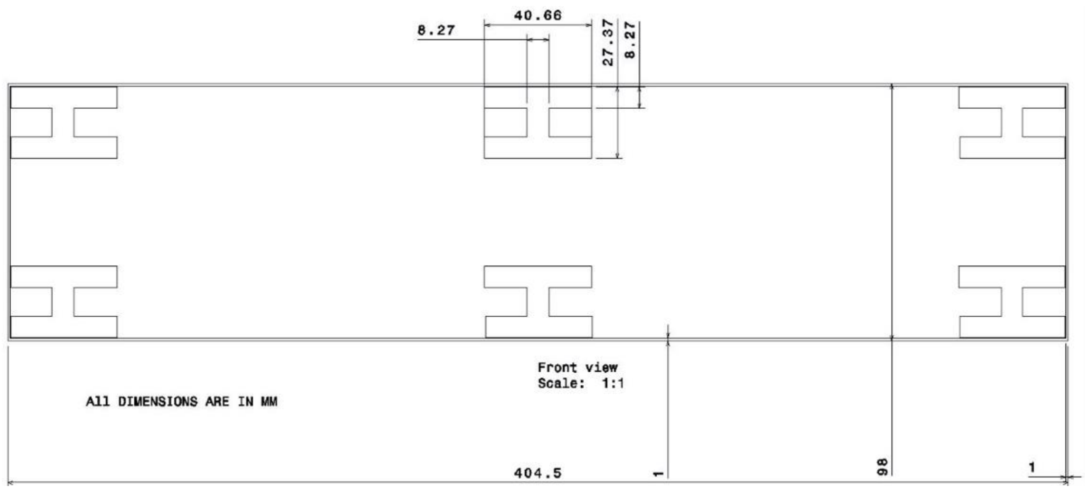


Figure 2: Recommended Minimum Weight Design based on scope of project using 7075-T6 Aluminum

3) L and I stiffener combination

Additionally, an L and I stiffener combination was investigated. Depicted in *figure 3* is the minimum weight design using L and I stiffeners based on the materials given in the project. This minimum weight design resulted in a wing weight of 192.51 *kg*. While lighter than the L stiffener wing, it is not the minimum weight design.

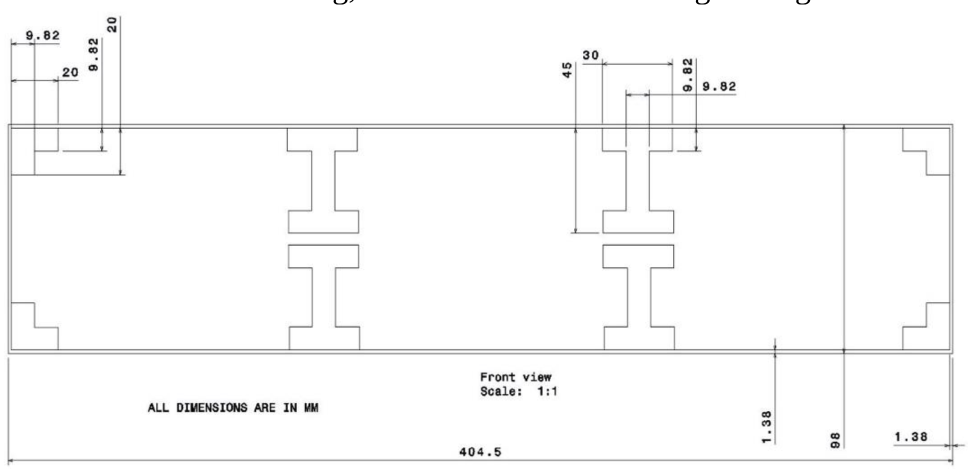


Figure 3: Wing Box design using L and I stiffeners with 7075-T6 Aluminum

4) 7068 Aluminum, Hat Stiffener and L and Hat Stiffener

Further trade studies were conducted in the prospects of finding a minimum weight design. Using 7068 Al instead of 7075-T6 Al increased weight reductions but ultimately six I section stiffeners resulted in a minimum weight design of 142.89 *kg*. Additional stiffener cross sections were investigated and resulted in two additional recommended designs. Shown in *figure 4* is a wing designed with only hat sections resulting in a minimum weight of 143.02 *kg* and had a higher design factor of 1.52. Also, shown in *figure 5* is a wing box designed with a combination of L and Hat stiffeners which ultimately is our recommended minimum weight design for a metal thin-walled box beam wing with a weight of 142.87 kg and a design factor of 1.53.

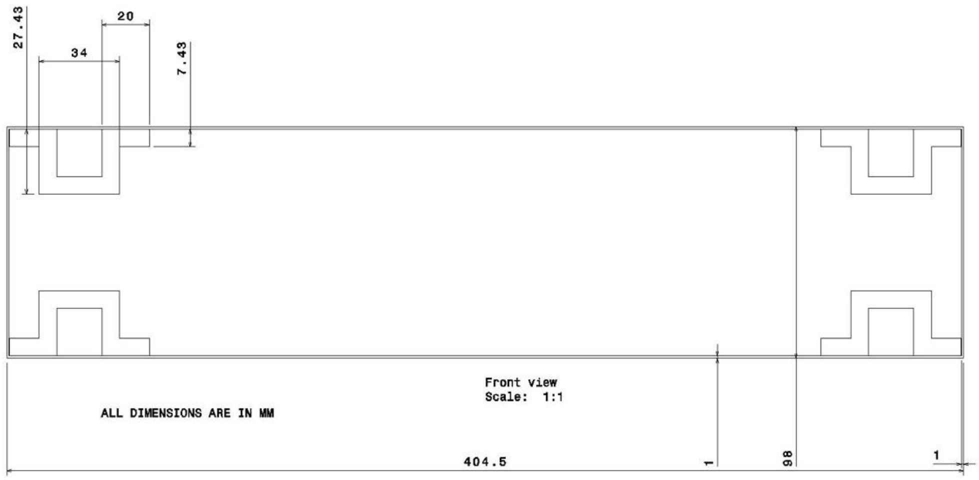


Figure 4: Additional Trade Study of using Hat Stiffeners and 7068 Aluminum

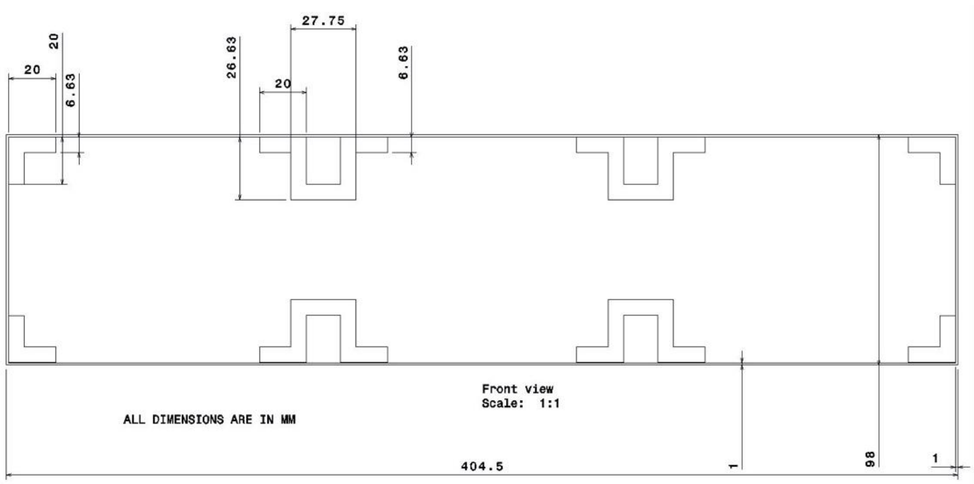


Figure 5: Recommended Minimum Weight Design for a metal thin-walled wing box using 7068

Composite Design

In addition to a minimum weight design using metal, a minimum weight design was also determined using composite material. As shown in *figure 6,* this design consists of a box beam made from T300-Expoxy composite with dimensions of 404 x 15 *mm* and six plies arranged in a stacking sequence of [45/-45/45]s with a volume fraction of 0.9. This wing can support the same lift distribution as well as meet the necessary factor of safety of 1.5 while only weighing 104.72 *kg*. Composites have a significant advantage especially when it comes to weight.

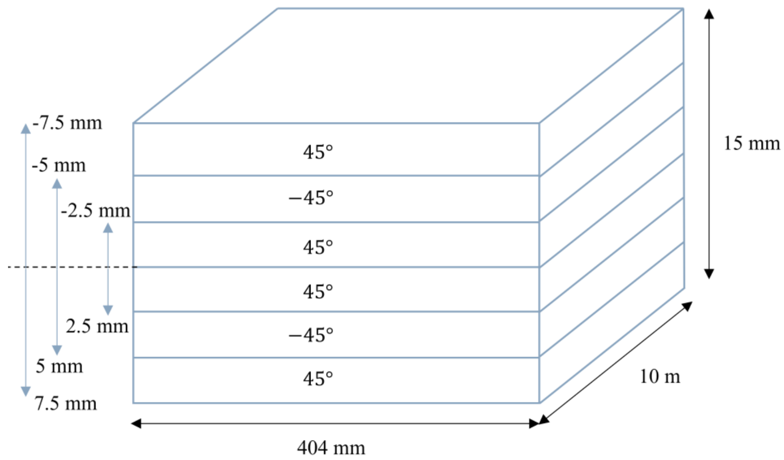


Figure 6: Stacking Sequence for a Composite Wing Box

3. Design Methodology and Trade-off Study

Metal Design

To understand the scope of the project, a free body diagram of the wing was drawn showing the lift force distribution along the length of the wing. Next, the shear force due to the lift distribution was calculated using the following formula $V = Lp' \cdot b$ from the wing root to the wing tip. Using the shear force, the bending moment about the wing root was calculated using $M = \int V\,dx$. Understanding that, bending stress of the wing could be calculated by $\sigma_{yy} = \frac{M_x}{I_x} \cdot z$ where M is the bending moment of the wing root, I is the moment of inertia of wing box, and z is the distance from the centroid, the next step was to solve for I and z. In order to minimize the number of calculations the wing box was assumed to be a symmetric beam which meant the z - distance from the centroid for each stiffener was the same. When calculating the moment of inertia, $I_x = \sum A_f \cdot x^2$, skin-stiffener idealization was used to calculate the idealized area of each stiffener by using the one-half and one-sixth rules, $A_f = A_S + \frac{1}{2}A_p + \frac{1}{6}A_w$. Once this calculation process was understood there was a hand calculation attempt to find the bending moment stress in wing using an L – section stiffener on the corners of the wing box with the following dimensions 2 mm W x 2mm H x 1mm thick. Once the bending stress was determined the structural safety was taken into consideration. As given in the project statement a factor of safety of 1.5 was required. The Factor of Safety is obtained by dividing the ultimate strength by the allowable stress. Depending on the specific material its ultimate yield stress was divided by the calculated bending stress and if it did not equal or exceed 1.5 the design was considered as a failure.

With this calculation process the next step was to design an optimization code to determine the minimum weight design. Using MATLAB, a code was created that varied the following variables: number of stiffeners, stiffener dimensions, and skin thickness and identified the minimum weight design per stiffener cross section. This code was then modified depending on the stiffener cross section used in the analysis of the wing box. The following stiffener cross sections that were investigated are as follows: L-section, I-section, L and I section, Hat section, L and Hat section. In the code, there are multiple for and if statements as well as a failure criterion that insures all the values that were considered as possible designs have a factor of safety equal to or greater than 1.5. The code then displays the number of stiffeners, stiffener dimensions, and skin thickness that would yield the minimum weight design as well as the weight of the design.

Analyzing the L cross-sectional shape. One of the assumptions made was that the cross section was proportional in that the height and width are the same. In its specific optimalization code it varies the thickness per a given height. Once the thickness is greater than or equal to the height it changes the height of the L section. It then returns the thickness back to a minimum thickness of 1 mm and repeats the process until the user specificized maximum height for the L cross-section. A total of 2.39 million variations per material of a L cross-section were tested and seen in table 1 are the minimum weight designs per material.

Table 1: Minimum Weight Design for an L cross section stiffener per material

L Stringer			
Material	2024 T6	7075 T6	7068 T6
No. of Stringers	4	6	8
Weight (kg)	447.97	293.93	184.2
Area (mm^2)	168.9	166.03	296.33
Height (mm)	20	19.45	20
Thickness (mm)	4.8	4.88	9.82
Skin Thickness (mm)	8.36	5.09	2.45

Similarly, an I cross sectional stiffener was investigated. For its analysis an assumption was made that the cross-section was symmetric in that the width at the top and bottom were equal. In its optimization code it varies the thickness until the thickness is equal to or greater than the width. Once a maximum thickness has been reached it will increase the height and width as to ensure that the height is at least 2x the width and would repeat this process until the users specified the maximum height. A total of 2.74 million variations per material of an I cross-section were tested and presented below in table 2 are the minimum weight designs per material.

Table 2: Minimum Weight Design for an I cross section stiffener per material

I Stringer			
Material	2024 T6	7075 T6	7068 T6
No. of Stringers	8	6	6
Weight (kg)	250.08	181.35	142.89
Area (mm^2)	915.34	597.57	523.04
Height (mm)	29.74	28.69	22.38
Width (mm)	44.57	42.83	32.40
Thickness (mm)	9.09	5.82	7.18
Skin Thickness (mm)	1	1	1

A proposed combination of L and I cross sectional stiffeners were investigated. For this analysis the L stiffeners were positioned on the corners of the wing box and the I stiffeners were positioned internally. For this specific optimization code, the assumptions made for the L and I as well as the individual stiffener calculations remained the same. But the code did recognize the difference between corner and internal stiffeners and idealized them accordingly. Additionally, this code was created with the intention that the wing box would always have more than four stiffeners. This combined L and I stiffener wing box had a total of 1.74 million variations per material and present below in table 3 are the minimum weight designs per material.

Table 3:Minimum Weight Design for an L and I cross section stiffeners per material

L and I Stringer Combination			
Material	2024 T6	7075 T6	7068 T6
L Stringer			
No. of Stringers	8	8	8
Weight (kg)	365.13	192.51	142.93
Area (mm^2)	292.76	296.33	219.93
Height (mm)	19.82	20	17.64
Thickness (mm)	9.82	9.82	8.09
Skin Thickness (mm)	5.22	1.38	1
I Stringer			
Area (mm^2)	828.5	838.12	615.56
Height (mm)	44.59	45	39.62
Width (mm)	29.72	30	26.32
Thickness (mm)	9.82	9.82	8.09

Further Trade Studies were conducted outside of the project scope. Aside from investigating an additional material, two additional cross-sections were investigated in the prospect of a minimum weight design. The additional cross-sections were a wing box with only hat stiffeners and a combination of L and hat stiffeners. Analyzing the impact of the hat cross-sections with regards to the minimum weight design required a complete overhaul of the initial code. In this specific code it varies the width of the hat section, thickness, and height. In this optimization code, it analyzes a hat section at a constant height and thickness and varies the total width of the hat section until the specified user's maximum width. It then resets and increases the thickness while keeping the height constant. If the thickness is equal to or greater than the height of the leg it increases the total height and the whole process is repeated. This further trade study included an additional 1.3 million variations per material and resulted in our ultimate minimum weight design for a metal wing box. Below, table 4 and table 5 are the minimum weight designs for their respective cross-sections per material.

Table 4:Minimum Weight Design for a Hat cross section stiffeners per material

Hat Stringer			
Material	2024 T6	7075 T6	7068 T6
No. of Stringers	8	8	6
Weight (kg)	249.9	181.78	142.91
Area (mm^2)	914.5	559.50	523.18
Height (mm)	25.50	19.88	19.07
Width Hat (mm)	58	55.75	51.57
Width of L (mm)	15.8	11	9.71
Thickness (mm)	10	8.88	9.36

Skin Thickness (mm)	1	1	1

Table 5:Minimum Weight Design for an L and Hat cross section stiffeners per material

L and Hat Stringer Combination			
Material	2024 T6	7075 T6	7068 T6
L Stringer			
No. of Stringers	8	8	8
Weight (kg)	315.10	181.2	142.87
Area (mm^2)	276.23	242.52	165
Height (mm)	17.75	18.80	20
Thickness (mm)	10	10	6.63
Skin Thickness (mm)	3.38	1	1
Hat Stringer			
Area (mm^2)	1153.75	951.28	670
Height (mm)	27.75	28.80	26.63
Width Hat (mm)	70	41.20	27.75
Width of L (mm)	17.75	18.80	20
Thickness (mm)	10	10	6.63

Composite Design

For the composite design, the box beam was simplified to a rectangular cross section made from a graphite epoxy laminate. Due to the lift acting along the wing, the N vector was zero, the M vector consisted of a moment load, $M_x = \left(\int_0^{10} y * L_p \, dy\right) - W$, where $y\ (m)$ was the half span of the wing, $L_p\ (N/m)$, was the lift distribution, and $W\ (Nm)$ was the weight of the wing. For the material, a T300-epoxy polymer matrix composite was chosen since T300 carbon fibers were the baseline reinforcement material used in the aerospace industry [8], and a generalized epoxy was chosen for the matrix based on averaged properties [9]. An optimization program was written in MATLAB to help find the best set of laminate parameters for the box beam.

First, the composite material properties were calculated from the fiber and matrix properties using the rule of mixtures. For example, the application of the rule to find the Young's modulus of the composite is as follows: $E_1 = E_f V_f + E_m V_m$. The same rule was applied to the density ρ, shear modulus E_2, and poison's ratio ν_{12}. The fiber volume fraction was varied from 0.1 to 0.9 to understand how every tenth of the fiber percentage affected the design, with increments of 0.1 for simplicity. Ply orientations of -45°, 90°, 45°, and 0° were chosen to analyze since these angles would result in optimal laminate properties. This is because the stiffness properties of the 90° and 0° plies are the extreme cases while the properties of the 45° and 45° plies lie in between the extreme cases. To account for the interactions between plies, the laminate was designed according to the Tsai-Hill criterion with a safety factor of 1.5 as given by the requirements. Furthermore, it is best to use a conservative failure criterion for aerospace structures. The carbon fiber/epoxy ultimate tensile, transverse, and shear strengths that were in the class notes were chosen as the allowable properties. The criteria for the best stacking sequence were a) produce the minimum weight design, b) distribute loads to each ply so every ply carries and counteracts loads. Six stacking sequences were analyzed to find the minimum weight design: [90/0], [90/0/90], [90/0]s, [45/-45]s, [90/45/-45/90], and [45/-45/45]s. The first three stacking sequences compared laminates with 0° and 90° plies, while the last three analyzed laminates with 45° plies. Upon, investigation as shown in *table 6,* the 0° plies were not carrying any loads in the [90/0] and [90/0/90] laminates, while the [90/0]s laminate had every ply carrying loads. Overall, the [45/-45]s, and [45/-45/45]s were the optimal laminates with each ply carrying counteracting the loads.

For simplicity, the laminate was designed assuming it is orthotropic. The mass of the wing was estimated using the cross-sectional area of the box beam and laminate density. The height of the beam was halved, from the given 98 mm to 49 mm since an optimal stacking sequence could not be determined as none of the laminates were failing.

For each stacking sequence and fiber volume fraction, the wing weight was calculated. First, the off-axis stiffness matrices Q for each ply given its orientation were calculated using the transformation matrix:

$$T = \begin{bmatrix} \cos^2(\theta) & \sin^2(\theta) & 2\cos(\theta)\sin(\theta) \\ \sin^2(\theta) & \cos^2(\theta) & -2\cos(\theta)\sin(\theta) \\ -\cos(\theta)\sin(\theta) & \cos(\theta)\sin(\theta) & \cos^2(\theta) - \sin^2(\theta) \end{bmatrix}$$

Second, the ABD (laminate stiffness) matrix was calculated using extensional, coupling, and flexural matrices respectfully: $A = \sum_{p=1}^{P} \left(\overline{Q_{IJ}}\right)_p (z_p - z_{p-1})$, $B = \sum_{p=1}^{P} \frac{1}{2} \left(\overline{Q_{IJ}}\right)_p (z_p^2 - z_{p-1}^2)$, and $D = \sum_{p=1}^{P} \frac{1}{3} \left(\overline{Q_{IJ}}\right)_p (z_p^3 - z_{p-1}^3)$. Where P is the number of plies in the laminate, z is the distance of the ply from the midplane of the laminate, and $\overline{Q_{IJ}}$ is the off-axis stiffness matrix of each ply.

Third, the midplane strains ε^0 and curvatures k were calculated:

$$\begin{Bmatrix} \varepsilon^0 \\ k \end{Bmatrix} = \begin{bmatrix} A & B \\ B & D \end{bmatrix}^{-1} \begin{Bmatrix} N \\ M \end{Bmatrix}.$$

Fourth, the local laminate strains $\{\varepsilon\} = \{\varepsilon^0\} + z\{k\}$ were calculated to find the stresses in each ply $\{\sigma_{xy}\}_p = \overline{[Q_P\,]}\,\{\varepsilon\}_P$, which were then transformed back to yield the global ply stresses. The ply stresses for the laminate were multiplied by the safety factor: $\{\sigma_{12}\}_P = 1.5[T]\{\sigma_{xy}\}_P$

Based on wing weight, [45/-45]s and [45/-45/45]s were the optimal stacking sequences as shown in *table 6,* but [45/-45/45]s was chosen since the Tsai-Hill values were 0.87 and 0.43 respectfully (Appendix: Optimization Code - Composite). This was because the lowest Tsai-Hill would yield the lowest weight as the beam height is varied. Thus, the beam height was further decreased until the minimum weight was achieved at a height of 15 *mm* as shown in *table 7.*

The minimum weight design of the box beam was designed to be made of a T300 Carbon fiber-Epoxy laminate composed of 90% fibers and 10% matrix, with a width of 404 *mm*, height of 15 *mm*, span of 10 *m*, and stacking sequence of [45/-45/45]s with plies of equal thickness of 2.5 mm. The minimized weight of the wing is 104.72 kg and the box cross-sectional area is 6100 mm^2.

Table 6:Stacking Sequence vs Wing Weight

Stacking Sequence	Fiber Volume Fraction V_f	Ply Thickness (mm)	Wing Weight (kg)
[90/0]	0.5	24.5	316.74
[90/0/90]	0.6	16.3	323.07
[90/0]s	0.3	12.3	304.07
[45/-45]s	0.1	12.3	291.4
[90/45/-45/90]	0.7	12.3	329.41
[45/-45/45]s	0.1	8.2	291.4

Table 7: Wing Height Vs. Wing Weight

Wing Height (mm)	Fiber Volume Fraction Vf	Wing Weight (kg)	Ply Thickness (mm)	Wing Cross-Section Area (mm^2)
49	0.1	291.4	12.3	19800
35	0.3	217.2	8.8	14100
25	0.4	158.37	4.2	10100
20	0.6	131.87	3.3	8100
17	0.8	116.48	2.8	6900
15	0.9	104.72	2.5	6100

4. Results and Discussion

Metal Design Results

The primary objective of the design project is to compare the minimum weight design of a thin-walled wing box constructed from two materials, Al-2024 and Al-7075. Based on the parameters and conditions of the project, a wing box made from 7075 T6 aluminum alloy and designed with six I – section stiffeners would produce the minimum weight design of 181.35 *kg* and meet the factor of safety criteria of 1.50. However, further trade-off studies were conducted with a different material (Al-7068) and different stiffener cross-sectional areas (hat-section). To give a brief background of Al-7068, it is one of the strongest commercially available aluminum alloys ever made. It was initially used in building the bodies of military ordinances but has now slowly found its way into commercial aerospace and automotive industries, as well as the field of medicine (prosthetic limbs for instance) [7]. This new minimum weight design calls for a wing box designed with a total of eight stiffeners - four L stiffeners on the corner and four hat stiffeners internally - all made from 7068 T6 aluminum alloy. This new alloy and combination of stiffeners produce an even more optimal design for the wing box, with a final weight of 142.87 *kg* and a higher factor of safety of 1.53. Since we are looking at the minimum weight (equivalent to the minimum material used), with a maximum number of stiffener and a higher factor of safety, the most optimal case would be to use the Al-7068 alloy, with a combination of four L and Hat stiffeners each.

Composite Design Results

In addition to finding a minimum weight design using metal, an additional minimum weight design was found using only composite material. Given the same lift distribution on the wing, a wing box made from T300-Epoxy composite would also support the wing loads as well as satisfy the structural factor of safety requirement of 1.5. This wing box would weigh 104.72 kg and be 404 x 15 *mm* in size. The composite material would have six plies arranged in the following stacking sequence of [45/-45/45]s and have a volume fraction of 0.9. It is clear from the above studies that a box beam made of T300-Epoxy can be significantly lighter than even the best of minimum weight designs found using only metal. It is important to note that the composite contained much more material than the hollow metal beam with stiffeners and still managed to be lighter, speculating the idea for further weight reduction.

5. Conclusions

Based on the problem statement for a wing designed with only metal components between the two primary metal alloys (Al-2024 T6 and Al-7075 T6), the recommended design would contain six I – section stiffener made of 7075 T6 Aluminum with the dimensions as 42.83 *mm* x 28.69 *mm* x 5.82 *mm*, resulting in a wing box weight of 181.21 *kg*. However, after conducting additional trade off studies on a new type of Aluminum alloy (Al-7068 T6) and different cross sections for the stiffeners, the minimum weight design is one with four internal hat section stiffeners, and four L section stiffeners on the corners of the wing box. The dimensions of the L stiffeners in this case were 20 *mm* x 20 *mm* x 6.63 *mm*, while that of the Hat stiffener was 27.75 *mm* x 26.63 *mm* x 6.63 *mm*, with a higher factor of safety F.S of 1.53. The weight of the wing box with skin-stiffener idealization came out as 142.87 *kg*. Comparing the values obtained, this is the lightest metal design achieved for the wing box. Currently, these aluminum alloys are majorly used in the aerospace industry and for military equipment such as missile parts and aircraft fittings, especially Al-7075 T6, due to its impressive ultimate tensile strength, and low density [10]. Al-2024 T6 is used in the aerospace and automotive industries (for instance in truck wheels, aircraft structural components and pistons) due to its resistance to corrosion [11]. Future applications of these alloys may be extended to the field of medicine, possibly in hip, knee or back replacements, or a new, lighter alloy containing the appropriate mix of each of these alloys could be created for use in aircraft structural components that are under high stress conditions.

Moving on to the composite designs, the lightest weight obtained was 104.72 *kg* with the dimensions of the box beam laminates as 404 *mm* x 15 *mm* made of a T300-Epoxy composite with a fiber volume fraction of 0.9 and stacking sequence of [45/-45/45]s. This case results in a minimum weight that is approximately 27% lower than the minimum weight design obtained in the case of aluminum alloys. Composites are now being increasingly utilized in the aerospace industries due to their extremely light weight and high strength (close to aluminum alloys). For instance, the Boeing 787 Dreamliner has a 50% weight percent of carbon fiber reinforced and other advanced composites. This results in a 20% reduction in the weight of the aircraft, as compared to other aluminum designs [12]. Future applications of composites in the aerospace industry can include carbon nanotubes as a substituent to the more conventional CFRPs, due to its high tensile strength (almost 100 times that of steel), high Young’s Modulus (about 5 times that of steel), and a density that is almost a quarter than that of steel [13].

References

Book Material

[1] D. J. Peery, *Aircraft Structures*. Mineola, New York: Dover Publication Inc, 2011.

[2] E. F. Bruhn, *Flight Vehicle Structures*. Carmel, Indiana: Jacobs Publishing Inc, 1973.

[3] R. M. Jones, *Mechanics of Composite Materials*. New York, NY: Taylor & Francis, 2006.

Material Properties

[4] "Aluminum 2024 T6 Material Property Data," *MatWeb*. [Online]. Available: https://www.matweb.com/search/DataSheet.aspx?MatGUID=ecf8530875cb4ded9675b827f77bfac5&ckck=1. [Accessed: 28-Apr-2022].

[5] "Aluminum 7075 T6 Material Property Data," *MatWeb*. [Online]. Available: https://www.matweb.com/search/DataSheet.aspx?MatGUID=4f19a42be94546b686bbf43f79c51b7d. [Accessed: 28-Apr-2022].

[6] "Aluminum 7068 T6 Material Property Data," *MatWeb*. [Online]. Available: https://www.matweb.com/search/datasheet_print.aspx?matguid=1ee9c9628f9d455b97fb4606daa84bd9. [Accessed: 28-Apr-2022].

[7] "7068 Aircraft Aluminum Alloy," Aircraft Aluminum. [Online]. Available https://www.aircraftaluminium.com/product/7068-aircraft-aluminum-alloy.html [Accessed, 30-Apr-22]

[8] "Torayca® Carbon Fiber," *Toray Composite Materials America, Inc.*, 03-Dec-2021. [Online]. Available: https://www.toraycma.com/products/carbon-fiber/. [Accessed: 01-May-2022].

[9] "Overview of materials for Epoxy Adhesive Material Property Data," *MatWeb*. [Online]. Available: https://www.matweb.com/search/datasheettext.aspx?matguid=c1ec1ad603c74f628578663aaf44f261#:~:text=Shear%20Modulus%2C%20Average%20value%3A%202.10%20GPa%20Grade%20Count%3A3&text=Adhesive%20Bond%20Strength%2C%20Average%20value%3A%2016.8%20MPa%20Grade%20Count%3A3 [Accessed: 01-May-2022]

[10] "All About 7075 Aluminum (Properties, Strength, and Uses)," *Thomasnet.* [Online] Available: https://www.thomasnet.com/articles/metals-metal-products/all-about-7075-aluminum-properties-strength-and-uses/ [Accessed 01-May 2022]

[11] "All About 2024 Aluminum (Properties, Strength, and Uses)," *Thomasnet.* [Online] Available: https://www.thomasnet.com/articles/metals-metal-products/2024-aluminum/ [Accessed 01-May-2022]

[12] “Boeing 787 From the Ground Up,” *Boeing.* [Online] Available: https://www.boeing.com/commercial/aeromagazine/articles/qtr_4_06/article_04_2.html [Accessed 01-May-2022]

[13] “Scanning the Properties of Nanotubes,” *UnderstandingNano.* [Online] Available: https://www.understandingnano.com/nanotubes-carbon-properties.html [Accessed 01-May-2022]

Appendix A

Metal Design Tables

- L and I Stiffeners with 2024 T6 Aluminum

L Stringer			
No. of Stringers	4	6	8
Weight (kg)	447.97	438.52	485.68
Area (mm^2)	168.9	165.91	173.79
Height (mm)	20	19.09	20
Thickness (mm)	4.8	5	4.96
Skin Thickness (mm)	8.36	8	9.82

I Stringer			
No. of Stringers	4	6	8
Weight (kg)	349.5	287.41	250.08
Area (mm^2)	981.5	992.71	915.34
Height (mm)	29.74	30	29.74
Width (mm)	44.57	45	44.57
Thickness (mm)	9.91	9.91	9.09
Skin Thickness (mm)	4.65	2.34	1

L and I Stringer Combination		
L Stringer		
No. of Stringers	6	8
Weight (kg)	385.59	365.13
Area (mm^2)	285.59	292.76
Height (mm)	19.64	19.82
Thickness (mm)	9.64	9.82
Skin Thickness (mm)	5.99	5.22
I Stringer		
Area (mm^2)	807.22	828.5
Height (mm)	44.17	44.59
Width (mm)	29.43	29.72
Thickness (mm)	9.64	9.82

Minimum Weight Design			
Cross Section	L	I	L & I
No. of Stringers	6	8	8
Weight (kg)	438.52	250.08	405.8
Area (mm^2)	165.91	915.34	L: 292.76 || I:828.5
Height (mm)	19.09	29.74	L: 19.82 || I: 44.59
Width (mm)		44.57	|| I: 29.72
Thickness (mm)	5	9.09	9.82

Skin Thickness (mm)	8	1	5.22

- L and I Stiffeners with 7075 T6 Aluminum

L Stringer			
No. of Stringers	4	6	8
Weight (kg)	303.76	293.93	320.98
Area (mm^2)	169.55	166.03	169.55
Height (mm)	19.45	19.45	19.45
Thickness (mm)	5	4.88	5
Skin Thickness (mm)	5.45	5.09	6.09

I Stringer			
No. of Stringers	4	6	8
Weight (kg)	202.86	181.21	181.35
Area (mm^2)	997.37	762.26	597.57
Height (mm)	29.47	27.37	28.69
Width (mm)	44.13	40.66	42.83
Thickness (mm)	10	8.27	5.82
Skin Thickness (mm)	1.77	1	1

L and I Stringer Combination		
L Stringer		
No. of Stringers	6	8
Weight (kg)	234.69	192.51
Area (mm^2)	300	296.33
Height (mm)	20	20
Thickness (mm)	10	9.82
Skin Thickness (mm)	2.92	1.38
I Stringer		
Area (mm^2)	850	838.12
Height (mm)	45	45
Width (mm)	30	30
Thickness (mm)	10	9.82

Minimum Weight Design

<table>
<tr><td>Cross Section</td><td>L</td><td>I</td><td>L & I</td></tr>
<tr><td>No. of Stringers</td><td>6</td><td>6</td><td>8</td></tr>
<tr><td>Weight (kg)</td><td>293.93</td><td>181.21</td><td>192.51</td></tr>
<tr><td>Area (mm^2)</td><td>166.03</td><td>762.26</td><td>L: 296.33 || I: 838.12</td></tr>
<tr><td>Height (mm)</td><td>19.45</td><td>27.37</td><td>L: 20 || I: 45</td></tr>
<tr><td>Width (mm)</td><td></td><td>40.66</td><td>|| I: 30</td></tr>
<tr><td>Thickness (mm)</td><td>4.88</td><td>8.27</td><td>9.82</td></tr>
<tr><td>Skin Thickness (mm)</td><td>5.09</td><td>1</td><td>1.38</td></tr>
</table>

Further Trade Studies

- L and I Stiffeners with 7068 T6 Aluminum

L Stringer			
No. of Stringers	4	6	8
Weight (kg)	206.99	186.59	184.2
Area (mm^2)	288.79	298.17	296.33
Height (mm)	20	20	20
Thickness (mm)	9.45	9.91	9.82
Skin Thickness (mm)	3.27	2.55	2.45

I Stringer			
No. of Stringers	4	6	8
Weight (kg)	142.9	142.89	142.97
Area (mm^2)	784.63	523.04	417.92
Height (mm)	25.80	22.38	18.18
Width (mm)	38.05	32.40	25.45
Thickness (mm)	9.45	7.18	7.82
Skin Thickness (mm)	1	1	1

L and I Stringer Combination		
L Stringer		
No. of Stringers	6	8
Weight (kg)	153.78	142.93
Area (mm^2)	290.70	219.93
Height (mm)	20	17.64
Thickness (mm)	9.55	8.09
Skin Thickness (mm)	1.38	1
I Stringer		
Area (mm^2)	820.04	615.56
Height (mm)	45	39.62
Width (mm)	30	26.32
Thickness (mm)	9.55	8.09

Minimum Weight Design			
Cross Section	L	I	L & I
No. of Stringers	8	6	8
Weight (kg)	184.2	142.89	142.93
Area (mm^2)	296.33	523.04	L: 219.93 || I: 615.56
Height (mm)	20	22.38	L: 17.64 || I: 39.62
Width (mm)		32.40	L: || I: 26.32
Thickness (mm)	9.82	7.18	8.09
Skin Thickness (mm)	2.45	1	1

- Hat Stiffener with 2024 T6 Aluminum

Hat Stringer			
No. of Stringers	4	6	8
Weight (kg)	319	253.96	249.9
Area (mm^2)	1300	1210	914.5
Height (mm)	30	28.88	25.50
Width Hat (mm)	70	67.75	58
Width of L (mm)	20	20	15.8
Thickness (mm)	10	8.88	10
Skin Thickness (mm)	3.38	1	1

L and Hat Stringer Combination		
L Stringer		
No. of Stringers	6	8
Weight (kg)	379.76	315.10
Area (mm^2)	276.23	276.23
Height (mm)	17.75	17.75
Thickness (mm)	10	10
Skin Thickness (mm)	5.75	3.38
Hat Stringer		
Area (mm^2)	940.75	1153.75
Height (mm)	27.75	27.75
Width Hat (mm)	46	70
Width of L (mm)	17.75	17.75
Thickness (mm)	10	10

- Hat Stiffener with 7075 T6 Aluminum

Hat Stinger			
No. of Stringers	4	6	8
Weight (kg)	181.96	183.36	181.78
Area (mm^2)	1150	775	559.50

Height (mm)	28.88	2.25	19.88
Width Hat (mm)	61.75	53.5	55.75
Width of L (mm)	20	15.50	11
Thickness (mm)	8.88	7.75	8.88
Skin Thickness (mm)	1	1	1

L and Hat Stringer Combination		
L Stringer		
No. of Stringers	6	8
Weight (kg)	216.75	181.2
Area (mm^2)	287.64	242.52
Height (mm)	18.80	18.80
Thickness (mm)	10	10
Skin Thickness (mm)	2.27	1
Hat Stringer		
Area (mm^2)	1161.84	951.28
Height (mm)	28.80	28.80
Width Hat (mm)	63.60	41.20
Width of L (mm)	18.80	18.80
Thickness (mm)	10	10

- Hat Stiffener with 7068 T6 Aluminum

Hat Stinger			
No. of Stringers	4	6	8
Weight (kg)	143.02	142.91	142.94
Area (mm^2)	785.71	523.18	417.82
Height (mm)	27.43	19.07	17.79
Width Hat (mm)	34	51.57	59.29
Width of L (mm)	20	9.71	9.71
Thickness (mm)	7.43	9.36	8.07
Skin Thickness (mm)	1	1	1

L and Hat Stringer Combination		
L Stringer		
No. of Stringers	6	8
Weight (kg)	144.63	142.87
Area (mm^2)	276.23	165
Height (mm)	17.75	20
Thickness (mm)	10.0	6.63
Skin Thickness (mm)	1	1
Hat Stringer		
Area (mm^2)	1047.25	670
Height (mm)	27.75	26.63

Width Hat (mm)	58	27.75
Width of L (mm)	17.75	20
Thickness (mm)	10	6.63

Composite Design Tables:

- Graphite Epoxy

Material Properties		
	T300 Carbon Fibers	Epoxy
Young's Modulus E (GPa)	233.0	3.340
Shear Modulus G (GPa)	8.963	2.100
Poison's Ratio v	0.200	0.321
Density ρ (kg/m^3)	1760	1440

- Wing Height: 49 mm, Wing Cross Sectional Area: 19800 mm^2

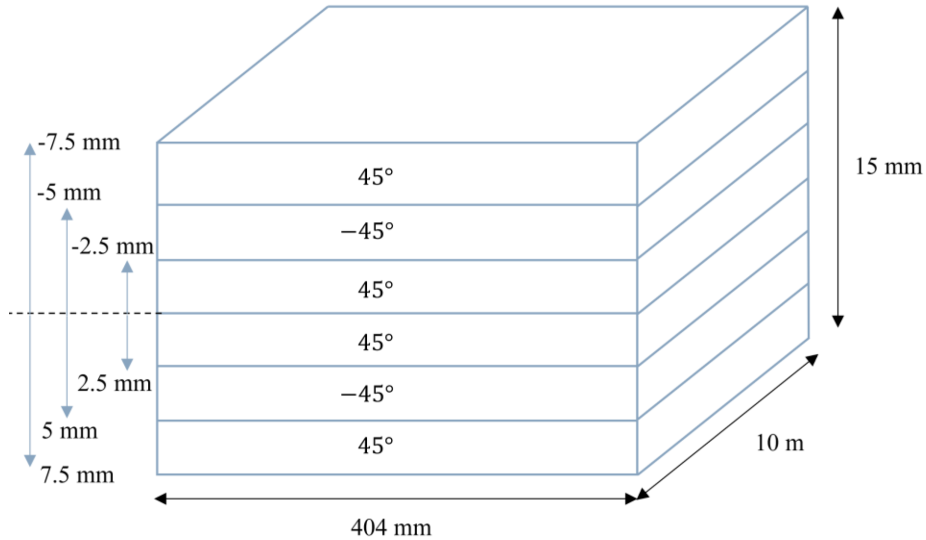
90° and 0° Laminates			
Stacking Sequence	Fiber Volume Fraction V_f	Ply Thickness t (mm)	Wing Weight M (kg)
[90/0]	0.5	24.5	316.74
[90/0/90]	0.6	16.3	323.07
[90/0]s	0.3	12.3	304.07

90°, 45°, and Laminates			
Stacking Sequence	Fiber Volume Fraction V_f	Ply Thickness t (mm)	Wing Weight M (kg)
[45/-45]s	0.1	12.3	291.4
[90/45/-45/90]	0.7	12.3	329.41
[45/-45/45]s	0.1	8.2	291.4

- [45/-45/45]s vs. Wing Height

[45/-45/45]s				
Wing Height (mm)	Fiber Volume Fraction V_f	Wing Weight M (kg)	Ply Thickness t (mm)	Wing Cross-Section Area A (mm^2)
49	0.1	291.4	12.3	19800
35	0.3	217.2	8.8	14100
25	0.4	158.37	4.2	10100
20	0.6	131.87	3.3	8100
17	0.8	116.48	2.8	6900
15	0.9	104.72	2.5	6100

- Minimum Weight Design:



Optimization Code - Metal:

MAE 526 Design Project Group 2 - Metal Design
Aishwarya Ledalla, Alexander Scalco, Lakshya Tiwari, Rohan Ravishekar

```
clear;clc;
```

Part 1: L Stringer
Length Across Wingspan

```
b = 10; % m
c = 1; % m
y = linspace(0,b,1000); % m

% Lift Across Wingspan
for i = 1:length(y)
    if y(i) <= 0.65*b
        Lp(i) = y(i)*5000;% (N)

    else
         Lp(i) = y(i)*(b-y(i))/b*5000; % (N)
    end
end

% Wing Box Dimentions
Lwing = 404.5E-3;          % (m) total length of box fitted inside airfoil
Hwing = 98E-3;             % (m) total height of box fitted inside airfoil
W = 808*9.81;              % (kg) weight of airfoil
xc = Lwing/2;              % centroid of wing in the horizontal direction

% Stringers
n = [2:4];   % number of stringers in top half (total = n*2)
nc = 2;      % corner stringers in top half
ni = n-nc;   % internal stringers in top half

% Moments
Mx = trapz(y,Lp)-W*5; % Nm
Mz = 0; % Nm

% Material Properties
Syield = 343E6;             % Nm
rho = 2.78*((100)^3/1000); % kg/m^3

q=100;                                 % Number of points

%Stiffner Dimensions
h = linspace(2,20,q)*10^(-3);          % Height of L stiffner
ts = linspace(1,5,q)*10^(-3);          % Stiffner Thickness
tw = linspace(1,10,q)*10^(-3);         % Skin Thickness
```

```
for t = 1:length(h)
    for j = 1:length(ts)
        if h(t)-ts(j) <=0
            break
        else
          % Area of L Section Calculation
             A_L(t,j) = (h(t)*ts(j))+(h(t)-ts(j)).*ts(j);
        end
    end
end
Astring=nonzeros(A_L);      %Removal of zero valued areas of the stiffner
```

Idealization of Area of Each Stringer

```
for s = 1:length(n)

    if nc == n(s)
        dx(s) = Lwing;  % (m) no internal stringers
    else
        dx(s) = Lwing/(2*ni(s)); % (m) internal stringers
    end
        dz = Hwing;

    for k=1:length(tw)

        for i=1:length(Astring)
            %Idealization of the Stringer on the Corner
            Ac(s,i,k)=Astring(i)+(1/2)*dx(s)*tw(k)+(1/6)*dz*tw(k);

            if ni(s) < 0
                Ai(s,i,k)=0
            else
            %Idealization of the Internal Stringer
             Ai(s,i,k) = Astring(i) + dx(s)*tw(k);
             xi(s) = xc - dx(s);
            end

         %Moments of Inertia Calculation
         Ixx(s,i,k)=2*nc*Ac(s,i,k)*(dz/2)^2 + 2*ni(s)*Ai(s,i,k)*(dz/2)^2;
         Izz(s,i,k)=2*nc*Ac(s,i,k)*(xc)^2 + 2*ni(s)*Ai(s,i,k)*(xi(s))^2;
         Izx = 0;

         % % Design Stress
         Z = Hwing/2;
         Syy(s,i,k) = -Z*Mx/Ixx(s,i,k);
         sf(s,i,k) = abs(Syield/Syy(s,i,k));
        end
    end
end
```

Minimum Weight Design

```
%Stringers on the Corner (4 Total)
for i=1:length(Astring)
    for k=1:length(tw)
    if sf(1,i,k) >=1.5
        safety_factor(i,k) = sf(1,i,k);
        Acorner(i,k) = Ac(1,i,k);
        Atot(i,k)=(404.5E-3*98E-3)-(404.5E-3-2.*tw(k)).*(98E-3 - 2.*tw(k))+4*Acorner(i,k);
        mass_z(i,k)=rho*Atot(i,k)*10;
        end
    end
   if i == length(Astring) && k == length(tw)
       min_mass_zero=min(nonzeros(mass_z));
       [i,k]=find(mass_z==min_mass_zero);

       fprintf('Stringers on the Corner Only\n')
       fprintf('Wing Box Weight = %0.2f kg \n',min_mass_zero)
       fprintf('Safety Factor is %0.2f \n',safety_factor(i,k))
       fprintf('\nStringer Information\n')
       fprintf('Area of L Stringer is %0.2f mm^2\n',Astring(i)*10^6)

       [t,j]=find(A_L==Astring(i));
       fprintf('Height is %0.2f mm \n',h(t)*10^3)
       fprintf('Thickness is %0.2f for mm\n',ts(j)*10^3)
       fprintf('The skin thickness is %0.2f mm\n',tw(k)*10^3)
       fprintf('-----------------------------------------------')

   end
    end

% One Internal Stringer (6 total)
for i=1:length(Astring)
    for k=1:length(tw)

    if sf(2,i,k) >=1.5
        safety_factor1(i,k) = sf(2,i,k);
        Acorner1(i,k) = Ac(2,i,k);
        Aint1(i,k) = Ai(2,i,k);

        Atot1(i,k)=(404.5E-3*98E-3)-(404.5E-3-2.*tw(k)).*(98E-3 -
2.*tw(k))+4*Acorner1(i,k)+2*ni(2)*Aint1(i,k);
        mass_o(i,k)=rho*Atot1(i,k)*10;
    end

    end
   if i == length(Astring) && k == length(tw)
       min_mass_one=min(nonzeros(mass_o));
       [i,k]=find(mass_o==min_mass_one);

       fprintf('\nStringers on the Corner and Two internally\n')
       fprintf('Wing Box Weight = %0.2f kg \n',min_mass_one)
       fprintf('Safety Factor is %0.2f \n',safety_factor1(i,k))
```

```
        fprintf('\nStringer Information\n')
        fprintf('Area of L Stringer is %0.2f mm^2\n',Astring(i)*10^6)

        [t,j]=find(A_L==Astring(i));
        fprintf('Height is %0.2f mm \n',h(t)*10^3)
        fprintf('Thickness is %0.2f for mm\n',ts(j)*10^3)
        fprintf('The skin thickness is %0.2f mm\n',tw(k)*10^3)
        fprintf('-----------------------------------------------')

    end
end

%Two Internal Stringers (8 total)
 for i=1:length(Astring)
    for k=1:length(tw)

    if sf(3,i,k) >=1.5
        safety_factor3(i,k) = sf(3,i,k);
        Acorner2(i,k) = Ac(3,i,k);
        Aint2(i,k) = Ai(3,i,k);
        Atot2(i,k)=(404.5E-3*98E-3)-(404.5E-3-2.*tw(k)).*(98E-3 -
2.*tw(k))+4*Acorner2(i,k)+2*ni(3)*Aint2(i,k); %math probably needs to be checked
        mass_t(i,k)=rho*Atot2(i,k)*10;
    end

    end
   if i == length(Astring) && k == length(tw)
       min_mass_two=min(nonzeros(mass_t));
       [i,k]=find(mass_t==min_mass_two);

       fprintf('\nStringers on the Corner and Four internally\n')
       fprintf('Wing Box Weight = %0.2f kg \n',min_mass_two)
       fprintf('Safety Factor is %0.2f \n',safety_factor3(i,k))
       fprintf('\nStringer Information\n')
       fprintf('Area of L Stringer is %0.2f mm^2\n',Astring(i)*10^6)

       [t,j]=find(A_L==Astring(i));
       fprintf('Height is %0.2f mm \n',h(t)*10^3)
       fprintf('Thickness is %0.2f for mm\n',ts(j)*10^3)
       fprintf('The skin thickness is %0.2f mm\n',tw(k)*10^3)
       fprintf('-----------------------------------------------\n')
   end

 end
```

```
Stringers on the Corner Only
Wing Box Weight = 447.97 kg
Safety Factor is 1.50

Stringer Information
Area of L Stringer is 168.90 mm^2
Height is 20.00 mm
Thickness is 4.80 for mm
```

```
The skin thickness is 8.36 mm
-----------------------------------------------
Stringers on the Corner and Two internally
Wing Box Weight = 438.52 kg
Safety Factor is 1.50

Stringer Information
Area of L Stringer is 165.91 mm^2
Height is 19.09 mm
Thickness is 5.00 for mm
The skin thickness is 8.00 mm
-----------------------------------------------
Stringers on the Corner and Four internally
Wing Box Weight = 485.68 kg
Safety Factor is 1.50

Stringer Information
Area of L Stringer is 173.79 mm^2
Height is 20.00 mm
Thickness is 4.96 for mm
The skin thickness is 9.82 mm
-----------------------------------------------
```

Part 2 I Stringer

```
clear;clc;

% Length Across Wingspan
b = 10; % m
c = 1; % m
y = linspace(0,b,1000); % m

% Lift Across Wingspan
for i = 1:length(y)
    if y(i) <= 0.65*b
        Lp(i) = y(i)*5000;% (N)

    else
         Lp(i) = y(i)*(b-y(i))/b*5000; % (N)
    end
end

% Wing Box Dimentions
Lwing = 404.5E-3;         % (m) total length of box fitted inside airfoil
Hwing = 98E-3;            % (m) total height of box fitted inside airfoil
W = 808*9.81;             % (kg) weight of airfoil
xc = Lwing/2;             % centroid of wing in the horizontal direction

% Stringers
n = [2:4];   % number of stringers in top half (total = n*2)
nc = 2;      % corner stringers in top half
ni = n-nc;   % internal stringers in top half

% Moments
```

```
Mx = trapz(y,Lp)-W*5; % Nm
Mz = 0; % Nm

% Material Properties
Syield = 343E6; % Nm
rho = 2.78*((100)^3/1000); % kg/m^3

q=100;                              % Number of points

%Stiffner Dimensions
w = linspace(2,30,q)*10^(-3);       % Width of I stiffner
h = linspace(4,20,q)*10^(-3);       % Height of I stiffner
ts = linspace(1,10,q)*10^(-3);      % Stiffner Thickness
tw = linspace(1,20,q)*10^(-3);      % Skin Thickness

for t = 1:length(w)
    for j = 1:length(ts)
        if h(t)-2*ts(j) <=0 && ts(j) >= w(t)
            break
        else
          %Area of I Section Calculation
             A_I(t,j) = 2*(w(t)*ts(j))+(h(t)-2*ts(j))*ts(j);
        end
    end
end

Astring=nonzeros(A_I);      % Removal of Zero Valued Areas of the Stiffner
```

Idealization of Area of Each Stringer

```
for s = 1:length(n)

    if nc == n(s)
        dx(s) = Lwing;            % (m) no internal stringers
    else
        dx(s) = Lwing/(2*ni(s)); % (m) internal stringers
    end
        dz = Hwing;

    for k=1:length(tw)

        for i=1:length(Astring)
            %Idealization of the Stringer on the Corner
            Ac(s,i,k)=Astring(i)+(1/2)*dx(s)*tw(k)+(1/6)*dz*tw(k);

            if ni(s) < 0
                Ai(s,i,k)=0
            else
             %Idealization of the Internal Stringer
             Ai(s,i,k) = Astring(i) + dx(s)*tw(k);
             xi(s) = xc - dx(s);
            end
```

```
            %Moments of Inertia Calculation
         Ixx(s,i,k)=2*nc*Ac(s,i,k)*(dz/2)^2 + 2*ni(s)*Ai(s,i,k)*(dz/2)^2;
         Izz(s,i,k)=2*nc*Ac(s,i,k)*(xc)^2 + 2*ni(s)*Ai(s,i,k)*(xi(s))^2;
         Izx = 0;

         % % Design Stress
         Z = Hwing/2;
         Syy(s,i,k) = -Z*Mx/Ixx(s,i,k);
         sf(s,i,k) = abs(Syield/Syy(s,i,k));
        end
    end
end
```

Minimum Weight Design

```
%Stringers on the Corner (4 Total)
for i=1:length(Astring)
    for k=1:length(tw)
    if sf(1,i,k) >=1.5
        safety_factor(i,k) = sf(1,i,k);
        Acorner(i,k) = Ac(1,i,k);
        Atot(i,k)=(404.5E-3*98E-3)-(404.5E-3-2.*tw(k)).*(98E-3 - 2.*tw(k))+4*Acorner(i,k);
        mass_z(i,k)=rho*Atot(i,k)*10;
        end
    end
   if i == length(Astring) && k == length(tw)
       min_mass_zero=min(nonzeros(mass_z));
       [i,k]=find(mass_z==min_mass_zero);

       fprintf('Stringers on the Corner Only\n')
       fprintf('Wing Box Weight = %0.2f kg \n',min_mass_zero)
       fprintf('Safety Factor is %0.2f \n',safety_factor(i,k))
       fprintf('\nStringer Information\n')
       fprintf('Area of I Stringer is %0.2f mm^2\n',Astring(i)*10^6)

       [t,j]=find(A_I==Astring(i));
       fprintf('Height is %0.2f mm \n',h(t)*10^3)
       fprintf('Width is %0.2f mm\n',w(t)*10^3)
       fprintf('Thickness is %0.2f for mm\n',ts(j)*10^3)
       fprintf('The skin thickness is %0.2f mm\n',tw(k)*10^3)
       fprintf('-----------------------------------------------')
   end
end

% One Internal Stringer (6 Total Stringers)
for i=1:length(Astring)
    for k=1:length(tw)

    if sf(2,i,k) >=1.5
        safety_factor1(i,k) = sf(2,i,k);
        Acorner1(i,k) = Ac(2,i,k);
        Aint1(i,k) = Ai(2,i,k);
```

```
        Atot1(i,k)=(404.5E-3*98E-3)-(404.5E-3-2.*tw(k)).*(98E-3 -
2.*tw(k))+4*Acorner1(i,k)+2*ni(2)*Aint1(i,k);
        mass_o(i,k)=rho*Atot1(i,k)*10;
    end

    end
   if i == length(Astring) && k == length(tw)
       min_mass_one=min(nonzeros(mass_o));
       [i,k]=find(mass_o==min_mass_one);

       fprintf('\nStringers on the Corner and two internally\n')
       fprintf('Wing Box Weight = %0.2f kg \n',min_mass_one)
       fprintf('Safety Factor is %0.2f \n',safety_factor1(i,k))
       fprintf('\nStringer Information\n')
       fprintf('Area of I Stringer is %0.2f mm^2\n',Astring(i)*10^6)

       [t,j]=find(A_I==Astring(i));
       fprintf('Height is %0.2f mm \n',h(t)*10^3)
       fprintf('Width is %0.2f mm\n',w(t)*10^3)
       fprintf('Thickness is %0.2f for mm\n',ts(j)*10^3)
       fprintf('The skin thickness is %0.2f mm\n',tw(k)*10^3)
       fprintf('-----------------------------------------------')

   end
end

% %Two Internal Stringers (8 Total Stringers)
 for i=1:length(Astring)
    for k=1:length(tw)

    if sf(3,i,k) >=1.5
        safety_factor3(i,k) = sf(3,i,k);
        Acorner2(i,k) = Ac(3,i,k);
        Aint2(i,k) = Ai(3,i,k);
        Atot2(i,k)=(404.5E-3*98E-3)-(404.5E-3-2.*tw(k)).*(98E-3 -
2.*tw(k))+4*Acorner2(i,k)+2*ni(3)*Aint2(i,k); %math probably needs to be checked
        mass_t(i,k)=rho*Atot2(i,k)*10;
    end
    end
   if i == length(Astring) && k == length(tw)
       min_mass_two=min(nonzeros(mass_t));
       [i,k]=find(mass_t==min_mass_two);

       fprintf('\nStringers on the Corner and four internally\n')
       fprintf('Wing Box Weight = %0.2f kg \n',min_mass_two)
       fprintf('Safety Factor is %0.2f \n',safety_factor3(i,k))
       fprintf('\nStringer Information\n')
       fprintf('Area of I Stringer is %0.2f mm^2\n',Astring(i)*10^6)

       [t,j]=find(A_I==Astring(i));
       fprintf('Height is %0.2f mm \n',h(t)*10^3)
       fprintf('Width is %0.2f mm\n',w(t)*10^3)
       fprintf('Thickness is %0.2f for mm\n',ts(j)*10^3)
       fprintf('The skin thickness is %0.2f mm\n',tw(k)*10^3)
```

```
        fprintf('-----------------------------------------------\n')
    end

  end
```

```
Stringers on the Corner Only
Wing Box Weight = 400.72 kg
Safety Factor is 1.50

Stringer Information
Area of I Stringer is 561.98 mm^2
Height is 20.00 mm
Width is 30.00 mm
Thickness is 9.09 for mm
The skin thickness is 6.57 mm
-----------------------------------------------
Stringers on the Corner and two internally
Wing Box Weight = 360.08 kg
Safety Factor is 1.50

Stringer Information
Area of I Stringer is 600.00 mm^2
Height is 20.00 mm
Width is 30.00 mm
Thickness is 10.00 for mm
The skin thickness is 5.03 mm
-----------------------------------------------
Stringers on the Corner and four internally
Wing Box Weight = 354.78 kg
Safety Factor is 1.50

Stringer Information
Area of I Stringer is 592.66 mm^2
Height is 20.00 mm
Width is 30.00 mm
Thickness is 9.82 for mm
The skin thickness is 4.84 mm
-----------------------------------------------
```

Part 3: Combination L & I Stringers

```
 clear;clc;

% Length Across Wingspan
b = 10; % m
c = 1; % m
y = linspace(0,b,1000); % m

% Lift Across Wingspan
for i = 1:length(y)
    if y(i) <= 0.65*b
        Lp(i) = y(i)*5000;% (N)
```

```
    else
         Lp(i) = y(i)*(b-y(i))/b*5000; % (N)
    end
end

% Wing Box Dimentions
Lwing = 404.5E-3;          % (m) total length of box fitted inside airfoil
Hwing = 98E-3;             % (m) total height of box fitted inside airfoil
W = 808*9.81;              % (kg) weight of airfoil
xc = Lwing/2;              % centroid of wing in the horizontal direction

% Stringers
n = [3,4];   % number of stringers in top half (total = n*2) (I)
nc = 2;      % corner stringers in top half
ni = n-nc;   % internal stringers in top half

% Moments
Mx = trapz(y,Lp)-W*5; % Nm
Mz = 0; % Nm

% Material
Syield = 343E6; % Nm
rho = 2.78*((100)^3/1000); % kg/m^3

q=100;                              % Number of points

%Stiffner Dimensions
w = linspace(2,20,q)*10^(-3);       % Width of I stiffner
h_I = linspace(4,30,q)*10^(-3);     % Height of I stiffner
h_L=linspace(2,20,q)*10^(-3);       % Height of L stiffner
ts = linspace(1,10,q)*10^(-3);      % Stiffner Thickness
tw = linspace(1,20,q)*10^(-3);      % Skin Thickness

for t = 1:length(w)
    for j = 1:length(ts)
        if h_I(t)-2*ts(j) <=0 && ts(j) >= w(t)
            break
        else
          %Area of L Section Calculation
          A_L(t,j) = (h_L(t)*ts(j))+(h_L(t)-ts(j)).*ts(j);
          %Area of I Section Calculation
           A_I(t,j) = 2*(w(t)*ts(j))+(h_I(t)-2*ts(j))*ts(j);
%               fprintf('t = %f\nj=%f\n \n',t,j)
        end
    end
end
Astring_corner=nonzeros(A_L);       % Removal of Zero Valued Areas of the L-Stiffner
Astring_internal=nonzeros(A_I);     % Removal of Zero Valued Areas of the I-Stiffner
```

Idealization of Area of Each Stringer

```
for s = 1:length(n)
          for k=1:length(tw)
```

```
            for i=1:length(Astring_corner)
           dx(s) = Lwing/(2*ni(s)); % (m) internal stringers
           dz = Hwing;

           %Idealization of the Stringer on the Corner
            Ac(s,i,k)=Astring_corner(i)+(1/2)*dx(s)*tw(k)+(1/6)*dz*tw(k);

           %Idealization of the Internal Stringer
            Ai(s,i,k) = Astring_internal(i) + dx(s)*tw(k);
            xi(s) = xc - dx(s);

            %Moments of Inertia Calculation
            Ixx(s,i,k)=2*nc*Ac(s,i,k)*(dz/2)^2 + 2*ni(s)*Ai(s,i,k)*(dz/2)^2;
            Izz(s,i,k)=2*nc*Ac(s,i,k)*(xc)^2 + 2*ni(s)*Ai(s,i,k)*(xi(s))^2;
            Izx = 0;

            % % Design Stress
            Z = Hwing/2;
            Syy(s,i,k) = -Z*Mx/Ixx(s,i,k);
            sf(s,i,k) = abs(Syield/Syy(s,i,k));
        end
    end
end
```

Minimum Weight Design

```
% One Internal Stringer (6 Total)
for i=1:length(Astring_corner)
    for k=1:length(tw)

    if sf(1,i,k) >=1.5
        safety_factor1(i,k) = sf(1,i,k);
        Acorner1(i,k) = Ac(1,i,k);
        Aint1(i,k) = Ai(1,i,k);
        Atot1(i,k)=(404.5E-3*98E-3)-(404.5E-3-2.*tw(k)).*(98E-3 -
2.*tw(k))+4*Acorner1(i,k)+2*ni(1)*Aint1(i,k);
        mass_o(i,k)=rho*Atot1(i,k)*10;
    end

    end
   if i == length(Astring_corner) && k == length(tw)
       min_mass_one=min(nonzeros(mass_o));
       [i,k]=find(mass_o==min_mass_one);
       fprintf('\nL Stringers on the Corner Two I Stiffner Internal\n')
       fprintf('Wing Box Weight = %0.2f kg \n',min_mass_one)
       fprintf('Safety Factor is %0.2f \n',safety_factor1(i,k))

       fprintf('\nL Stringer Information\n')
       fprintf('Area of L Stringer is %0.2f mm^2\n',Astring_corner(i)*10^6)

       [t,j]=find(A_L==Astring_corner(i));
       fprintf('Height is %0.2f mm \n',h_L(t)*10^3)
       fprintf('Thickness is %0.2f for mm\n',ts(j)*10^3)
```

```
        fprintf('The skin thickness is %0.2f mm\n',tw(k)*10^3)

        fprintf('\nI Stringer Information\n')
        fprintf('Area of I Stringer is %0.2f mm^2\n',Astring_internal(i)*10^6)

        [t,j]=find(A_I==Astring_internal(i));
        fprintf('Height is %0.2f mm \n',h_I(t)*10^3)
        fprintf('Width is %0.2f mm\n',w(t)*10^3)
        fprintf('Thickness is %0.2f for mm\n',ts(j)*10^3)
        fprintf('The skin thickness is %0.2f mm\n',tw(k)*10^3)
        fprintf('-----------------------------------------------')
    end
end
%
% %Two Internal Stringers (8 Total)
 for i=1:length(Astring_corner)
    for k=1:length(tw)

    if sf(2,i,k) >=1.5
        safety_factor3(i,k) = sf(2,i,k);
        Acorner2(i,k) = Ac(2,i,k);
        Aint2(i,k) = Ai(2,i,k);

        Atot2(i,k)=(404.5E-3*98E-3)-(404.5E-3-2.*tw(k)).*(98E-3 -
2.*tw(k))+4*Acorner2(i,k)+2*ni(2)*Aint2(i,k); %math probably needs to be checked
        mass_t(i,k)=rho*Atot2(i,k)*10;
    end

    end
   if i == length(Astring_corner) && k == length(tw)
       min_mass_two=min(nonzeros(mass_t));
       [i,k]=find(mass_t==min_mass_two);
       fprintf('\nL Stringers on the Corner Four I Stiffner Internal\n')
       fprintf('Wing Box Weight = %0.2f kg \n',min_mass_two)
       fprintf('Safety Factor is %0.2f \n',safety_factor3(i,k))

       fprintf('\nL Stringer Information\n')
       fprintf('Area of L Stringer is %0.2f mm^2\n',Astring_corner(i)*10^6)

       [t,j]=find(A_L==Astring_corner(i));
       fprintf('Height is %0.2f mm \n',h_L(t)*10^3)
       fprintf('Thickness is %0.2f for mm\n',ts(j)*10^3)
       fprintf('The skin thickness is %0.2f mm\n',tw(k)*10^3)

       fprintf('\nI Stringer Information\n')
       fprintf('Area of I Stringer is %0.2f mm^2\n',Astring_internal(i)*10^6)

       [t,j]=find(A_I==Astring_internal(i));
       fprintf('Height is %0.2f mm \n',h_I(t)*10^3)
       fprintf('Width is %0.2f mm\n',w(t)*10^3)
       fprintf('Thickness is %0.2f for mm\n',ts(j)*10^3)
       fprintf('The skin thickness is %0.2f mm\n',tw(k)*10^3)
       fprintf('-----------------------------------------------\n')
```

```
    end
  end
```

```
L Stringers on the Corner Two I Stiffner Internal
Wing Box Weight = 405.80 kg
Safety Factor is 1.50

L Stringer Information
Area of L Stringer is 283.76 mm^2
Height is 19.27 mm
Thickness is 9.91 for mm
The skin thickness is 6.76 mm

I Stringer Information
Area of I Stringer is 472.43 mm^2
Height is 28.95 mm
Width is 19.27 mm
Thickness is 9.91 for mm
The skin thickness is 6.76 mm
-----------------------------------------------
L Stringers on the Corner Four I Stiffner Internal
Wing Box Weight = 416.24 kg
Safety Factor is 1.50

L Stringer Information
Area of L Stringer is 300.00 mm^2
Height is 20.00 mm
Thickness is 10.00 for mm
The skin thickness is 7.14 mm

I Stringer Information
Area of I Stringer is 500.00 mm^2
Height is 30.00 mm
Width is 20.00 mm
Thickness is 10.00 for mm
The skin thickness is 7.14 mm
-----------------------------------------------
```

Part 4 Hat Stringer

```
clear;clc;

% Length Across Wingspan
b = 10; % m
c = 1; % m
y = linspace(0,b,1000); % m

% Lift Across Wingspan
for i = 1:length(y)
    if y(i) <= 0.65*b
        Lp(i) = y(i)*5000;% (N)
```

```
    else
         Lp(i) = y(i)*(b-y(i))/b*5000; % (N)
    end
end

% Wing Box Dimentions
Lwing = 404.5E-3;          % (m) total length of box fitted inside airfoil
Hwing = 98E-3;             % (m) total height of box fitted inside airfoil
W = 808*9.81;              % (kg) weight of airfoil
xc = Lwing/2;              % centroid of wing in the horizontal direction

% Stringers
n = [2:4];   % number of stringers in top half (total = n*2)
nc = 2;      % corner stringers in top half
ni = n-nc;   % internal stringers in top half

% Moments
Mx = trapz(y,Lp)-W*5; % Nm
Mz = 0; % Nm

% Material Properties
Syield = 683E6; % Nm
rho = 2.85*((100)^3/1000); % kg/m^3

q=15;                           % Number of points

%Stiffner Dimensions
w = linspace(2,50,q)*10^(-3);       % Width of Hat section
h = linspace(2,20,q)*10^(-3);       % Height of stiffner
ts = linspace(1,10,q)*10^(-3);      % Stiffner Thickness
tw = linspace(1,20,q)*10^(-3);      % Skin Thickness

for v = 1:length(w)
    for t = 1:length(h)
        for j = 1:length(ts)
            if h(t)-ts(j) <=0
                break
            else
            %  Area of L Section Calculation
             A_hat(v,t,j) = 2*((h(t)*ts(j))+(h(t)-ts(j)).*ts(j))+ts(j)*(2*ts(j)+w(v));
            end
        end
    end

end

Astring=nonzeros(A_hat);      % Removal of Zero Valued Areas of the Stiffner
```

Idealization of Area of Each Stringer

```
for s = 1:length(n)

    if nc == n(s)
```

```
        dx(s) = Lwing;           % (m) no internal stringers
    else
        dx(s) = Lwing/(2*ni(s)); % (m) internal stringers
    end
        dz = Hwing;

    for k=1:length(tw)

        for i=1:length(Astring)
            %Idealization of the Stringer on the Corner
            Ac(s,i,k)=Astring(i)+(1/2)*dx(s)*tw(k)+(1/6)*dz*tw(k);

            if ni(s) < 0
                Ai(s,i,k)=0
            else
             %Idealization of the Internal Stringer
             Ai(s,i,k) = Astring(i) + dx(s)*tw(k);
             xi(s) = xc - dx(s);
            end

            %Moments of Inertia Calculation
         Ixx(s,i,k)=2*nc*Ac(s,i,k)*(dz/2)^2 + 2*ni(s)*Ai(s,i,k)*(dz/2)^2;
         Izz(s,i,k)=2*nc*Ac(s,i,k)*(xc)^2 + 2*ni(s)*Ai(s,i,k)*(xi(s))^2;
         Izx = 0;

         % % Design Stress
         Z = Hwing/2;
         Syy(s,i,k) = -Z*Mx/Ixx(s,i,k);
         sf(s,i,k) = abs(Syield/Syy(s,i,k));
        end
    end
end
```

Minimum Weight Design

```
%Stringers on the Corner (4 Total)
for i=1:length(Astring)
    for k=1:length(tw)
        if sf(1,i,k) >=1.5
        safety_factor(i,k) = sf(1,i,k);
        Acorner(i,k) = Ac(1,i,k);
        Atot(i,k)=(404.5E-3*98E-3)-(404.5E-3-2.*tw(k)).*(98E-3 - 2.*tw(k))+4*Acorner(i,k);
        mass_z(i,k)=rho*Atot(i,k)*10;
        end
    end
   if i == length(Astring) && k == length(tw)
       min_mass_zero=min(nonzeros(mass_z));
       [i,k]=find(mass_z==min_mass_zero);

       fprintf('Stringers on the Corner Only\n')
       fprintf('Wing Box Weight = %0.2f kg \n',min_mass_zero)
       fprintf('Safety Factor is %0.2f \n',safety_factor(i,k))
       fprintf('\nStringer Information\n')
```

```
        fprintf('Area of Hat Stringer is %0.2f mm^2\n',Astring(i)*10^6)


    for v=1:length(w)
        for t =1:length(h)
            for j = 1:length(ts)
             val = A_hat(v,j,t) == Astring(i);
             if  val == true
            fprintf('Total Height is %0.2f mm \n',(h(t)+ts(j))*10^3)
            fprintf('Width Hat is %0.2f mm\n',(w(v)+2*ts(j))*10^3)
            fprintf('Width of L is %0.2f mm\n',h(t)*10^3)
            fprintf('Thickness is %0.2f for mm\n',ts(j)*10^3)
            fprintf('The skin thickness is %0.2f mm\n',tw(k)*10^3)
            fprintf('-----------------------------------------------\n')

             end

            end
        end
    end


     end
   end




% One Internal Stringer (6 Total Stringers)
for i=1:length(Astring)
    for k=1:length(tw)

    if sf(2,i,k) >=1.5
        safety_factor1(i,k) = sf(2,i,k);
        Acorner1(i,k) = Ac(2,i,k);
        Aint1(i,k) = Ai(2,i,k);

        Atot1(i,k)=(404.5E-3*98E-3)-(404.5E-3-2.*tw(k)).*(98E-3 -
2.*tw(k))+4*Acorner1(i,k)+2*ni(2)*Aint1(i,k);
        mass_o(i,k)=rho*Atot1(i,k)*10;
    end

    end
   if i == length(Astring) && k == length(tw)
       min_mass_one=min(nonzeros(mass_o));
       [i,k]=find(mass_o==min_mass_one);

       fprintf('Stringers on the Corner and Two Internal\n')
       fprintf('Wing Box Weight = %0.2f kg \n',min_mass_one)
       fprintf('Safety Factor is %0.2f \n',safety_factor1(i,k))
       fprintf('\nStringer Information\n')
       fprintf('Area of Hat Stringer is %0.2f mm^2\n',Astring(i)*10^6)
```

```
    for v=1:length(w)
        for t =1:length(h)
            for j = 1:length(ts)
             val = A_hat(v,j,t) == Astring(i);
             if  val == true
            fprintf('Total Height is %0.2f mm \n',(h(t)+ts(j))*10^3)
            fprintf('Width Hat is %0.2f mm\n',(w(v)+2*ts(j))*10^3)
            fprintf('Width of L is %0.2f mm\n',h(t)*10^3)
            fprintf('Thickness is %0.2f for mm\n',ts(j)*10^3)
            fprintf('The skin thickness is %0.2f mm\n',tw(k)*10^3)
            fprintf('-----------------------------------------------\n')

             end

            end
        end
    end


     end
   end
% %Two Internal Stringers (8 Total Stringers)
 for i=1:length(Astring)
     for k=1:length(tw)

     if sf(3,i,k) >=1.5
         safety_factor3(i,k) = sf(3,i,k);
         Acorner2(i,k) = Ac(3,i,k);
         Aint2(i,k) = Ai(3,i,k);
         Atot2(i,k)=(404.5E-3*98E-3)-(404.5E-3-2.*tw(k)).*(98E-3 -
2.*tw(k))+4*Acorner2(i,k)+2*ni(3)*Aint2(i,k); %math probably needs to be checked
         mass_t(i,k)=rho*Atot2(i,k)*10;
     end
     end
    if i == length(Astring) && k == length(tw)
        min_mass_two=min(nonzeros(mass_t));
        [i,k]=find(mass_t==min_mass_two);

        fprintf('\nStringers on the Corner and four internally\n')
        fprintf('Wing Box Weight = %0.2f kg \n',min_mass_two)
        fprintf('Safety Factor is %0.2f \n',safety_factor3(i,k))
        fprintf('\nStringer Information\n')
        fprintf('Area of Hat Stringer is %0.2f mm^2\n',Astring(i)*10^6)

    for v=1:length(w)
        for t =1:length(h)
            for j = 1:length(ts)
             val = A_hat(v,j,t) == Astring(i);
             if  val == true
            fprintf('Total Height is %0.2f mm \n',(h(t)+ts(j))*10^3)
            fprintf('Width Hat is %0.2f mm\n',(w(v)+2*ts(j))*10^3)
            fprintf('Width of L is %0.2f mm\n',h(t)*10^3)
            fprintf('Thickness is %0.2f for mm\n',ts(j)*10^3)
            fprintf('The skin thickness is %0.2f mm\n',tw(k)*10^3)
```

```
            fprintf('-----------------------------------------------\n')

             end

            end
        end
    end
    end

  end
```

```
Stringers on the Corner Only
Wing Box Weight = 143.02 kg
Safety Factor is 1.50
Safety Factor is 1.50
Safety Factor is 1.50
Safety Factor is 1.50

Stringer Information
Area of Hat Stringer is 785.71 mm^2
Area of Hat Stringer is 785.71 mm^2
Total Height is 27.43 mm
Width Hat is 34.00 mm
Width of L is 20.00 mm
Thickness is 7.43 for mm
The skin thickness is 1.00 mm
The skin thickness is 1.00 mm
-----------------------------------------------
Total Height is 26.14 mm
Width Hat is 41.71 mm
Width of L is 20.00 mm
Thickness is 6.14 for mm
The skin thickness is 1.00 mm
The skin thickness is 1.00 mm
-----------------------------------------------
Stringers on the Corner and Two Internal
Wing Box Weight = 142.91 kg
Safety Factor is 1.50
Safety Factor is 1.50
Safety Factor is 1.50
Safety Factor is 1.50

Stringer Information
Area of Hat Stringer is 523.18 mm^2
Area of Hat Stringer is 523.18 mm^2
Total Height is 19.07 mm
Width Hat is 51.57 mm
Width of L is 9.71 mm
Thickness is 9.36 for mm
The skin thickness is 1.00 mm
The skin thickness is 1.00 mm
-----------------------------------------------
Total Height is 17.79 mm
```

```
Width Hat is 59.29 mm
Width of L is 9.71 mm
Thickness is 8.07 for mm
The skin thickness is 1.00 mm
The skin thickness is 1.00 mm
-----------------------------------------------

Stringers on the Corner and four internally
Wing Box Weight = 142.94 kg
Safety Factor is 1.50
Safety Factor is 1.50
Safety Factor is 1.50
Safety Factor is 1.50
Safety Factor is 1.50
Safety Factor is 1.50
Safety Factor is 1.50
Safety Factor is 1.50
Safety Factor is 1.50
Safety Factor is 1.50
Safety Factor is 1.50
Safety Factor is 1.50
Safety Factor is 1.50
Safety Factor is 1.50
Safety Factor is 1.50
Safety Factor is 1.50

Stringer Information
Area of Hat Stringer is 417.82 mm^2
Area of Hat Stringer is 417.82 mm^2
Area of Hat Stringer is 417.82 mm^2
Area of Hat Stringer is 417.82 mm^2
```

Part 5 L and Hat Stiffners

```
clear;clc;

% Length Across Wingspan
b = 10; % m
c = 1; % m
y = linspace(0,b,1000); % m

% Lift Across Wingspan
for i = 1:length(y)
    if y(i) <= 0.65*b
        Lp(i) = y(i)*5000;% (N)

    else
         Lp(i) = y(i)*(b-y(i))/b*5000; % (N)
    end
end

% Wing Box Dimentions
Lwing = 404.5E-3;           % (m) total length of box fitted inside airfoil
Hwing = 98E-3;              % (m) total height of box fitted inside airfoil
```

```
W = 808*9.81;              % (kg) weight of airfoil
xc = Lwing/2;              % centroid of wing in the horizontal direction

% Stringers
n = [3,4];   % number of stringers in top half (total = n*2) (I)
nc = 2;      % corner stringers in top half
ni = n-nc;   % internal stringers in top half

% Moments
Mx = trapz(y,Lp)-W*5; % Nm
Mz = 0; % Nm

% Material
Syield = 683E6;             % Nm
rho = 2.85*((100)^3/1000); % kg/m^3

q=9;                                % Number of points

%Stiffner Dimensions
w = linspace(2,50,q)*10^(-3);       % Width of Hat section
h_L=linspace(2,20,q)*10^(-3);       % Height of L stiffner
ts = linspace(1,10,q)*10^(-3);      % Stiffner Thickness
tw = linspace(1,20,q)*10^(-3);      % Skin Thickness


for v = 1:length(w)
    for t = 1:length(w)
        for j = 1:length(ts)
            if h_L(t)-ts(j) <=0
                break
            else
              %Area of L Section Calculation
              A_L(v,t,j) = (h_L(t)*ts(j))+(h_L(t)-ts(j)).*ts(j);
              %Area of I Section Calculation
              A_hat(v,t,j) = 2*((h_L(t)*ts(j))+(h_L(t)-ts(j)).*ts(j))+ts(j)*(2*ts(j)+w(v));
    %              fprintf('t = %f\nj=%f\n \n',t,j)
            end
        end
    end
end
Astring_corner=nonzeros(A_L);       % Removal of Zero Valued Areas of the L-Stiffner
Astring_internal=nonzeros(A_hat);     % Removal of Zero Valued Areas of the I-Stiffner
```

Idealization of Area of Each Stringer

```
for s = 1:length(n)
          for k=1:length(tw)
              for i=1:length(Astring_corner)
            dx(s) = Lwing/(2*ni(s)); % (m) internal stringers
            dz = Hwing;

            %Idealization of the Stringer on the Corner
             Ac(s,i,k)=Astring_corner(i)+(1/2)*dx(s)*tw(k)+(1/6)*dz*tw(k);
```

```
            %Idealization of the Internal Stringer
             Ai(s,i,k) = Astring_internal(i) + dx(s)*tw(k);
             xi(s) = xc - dx(s);

             %Moments of Inertia Calculation
             Ixx(s,i,k)=2*nc*Ac(s,i,k)*(dz/2)^2 + 2*ni(s)*Ai(s,i,k)*(dz/2)^2;
             Izz(s,i,k)=2*nc*Ac(s,i,k)*(xc)^2 + 2*ni(s)*Ai(s,i,k)*(xi(s))^2;
             Izx = 0;

             % % Design Stress
             Z = Hwing/2;
             Syy(s,i,k) = -Z*Mx/Ixx(s,i,k);
             sf(s,i,k) = abs(Syield/Syy(s,i,k));
        end
    end
end
```

Minimum Weight Design

One Internal Stringer (6 Total Stringers)

```
for i=1:length(Astring_corner)
    for k=1:length(tw)

    if sf(1,i,k) >=1.5
        safety_factor1(i,k) = sf(1,i,k);
        Acorner1(i,k) = Ac(1,i,k);
        Aint1(i,k) = Ai(1,i,k);

        Atot1(i,k)=(404.5E-3*98E-3)-(404.5E-3-2.*tw(k)).*(98E-3 -
2.*tw(k))+4*Acorner1(i,k)+2*ni(1)*Aint1(i,k);
        mass_o(i,k)=rho*Atot1(i,k)*10;
    end

    end
   if i == length(Astring_corner) && k == length(tw)
       min_mass_one=min(nonzeros(mass_o));
       [i,k]=find(mass_o==min_mass_one);

       fprintf('Stringers on the Corner and Two Internal\n')
       fprintf('Wing Box Weight = %0.2f kg \n',min_mass_one)
       fprintf('Safety Factor is %0.2f \n',safety_factor1(i,k))
       fprintf('\nStringer Information\n')
       fprintf('Area of L Stringer is %0.2f mm^2\n',Astring_corner(i)*10^6)
       fprintf('Area of Hat Stringer is %0.2f mm^2\n',Astring_internal(i)*10^6)


   for v=1:length(w)
       for t =1:length(h_L)
           for j = 1:length(ts)
            val = A_hat(v,j,t) == Astring_internal(i);
               if  val == true
```

```
                fprintf('\nL Stringer Dimensions\n')
                fprintf('Height is %0.2f mm \n',h_L(t)*10^3)
                fprintf('Thickness is %0.2f for mm\n',ts(j)*10^3)
                fprintf('The skin thickness is %0.2f mm\n',tw(k)*10^3)
                 end

             val1 = A_hat(v,j,t) == Astring_internal(i);
             if  val1 == true
            fprintf('\nHat Stringer Dimensions\n')
            fprintf('Total Height is %0.2f mm \n',(h_L(t)+ts(j))*10^3)
            fprintf('Width Hat is %0.2f mm\n',(w(v)+2*ts(j))*10^3)
            fprintf('Width of L is %0.2f mm\n',h_L(t)*10^3)
            fprintf('Thickness is %0.2f for mm\n',ts(j)*10^3)
            fprintf('The skin thickness is %0.2f mm\n',tw(k)*10^3)
            fprintf('-----------------------------------------------\n')

             end

            end
        end
    end


     end
   end
```

```
Stringers on the Corner and Two Internal
Wing Box Weight = 144.63 kg
Safety Factor is 1.52

Stringer Information
Area of L Stringer is 276.23 mm^2
Area of Hat Stringer is 1047.25 mm^2

L Stringer Dimensions
Height is 17.75 mm
Thickness is 10.00 for mm
The skin thickness is 1.00 mm

Hat Stringer Dimensions
Total Height is 27.75 mm
Width Hat is 58.00 mm
Width of L is 17.75 mm
Thickness is 10.00 for mm
The skin thickness is 1.00 mm
-----------------------------------------------
```

Two Internal Stringers (8 Total Stringers)

```
 for i=1:length(Astring_corner)
    for k=1:length(tw)

    if sf(2,i,k) >=1.5
```

```
        safety_factor3(i,k) = sf(2,i,k);
        Acorner2(i,k) = Ac(2,i,k);
        Aint2(i,k) = Ai(2,i,k);
        Atot2(i,k)=(404.5E-3*98E-3)-(404.5E-3-2.*tw(k)).*(98E-3 -
2.*tw(k))+4*Acorner2(i,k)+2*ni(2)*Aint2(i,k); %math probably needs to be checked
        mass_t(i,k)=rho*Atot2(i,k)*10;
    end
    end
   if i == length(Astring_corner) && k == length(tw)
       min_mass_two=min(nonzeros(mass_t));
       [i,k]=find(mass_t==min_mass_two);

       fprintf('\nStringers on the Corner and four internally\n')
       fprintf('Wing Box Weight = %0.2f kg \n',min_mass_two)
       fprintf('Safety Factor is %0.2f \n',safety_factor3(i,k))
       fprintf('\nStringer Information\n')
       fprintf('Area of L Stringer is %0.2f mm^2\n',Astring_corner(i)*10^6)
       fprintf('Area of Hat Stringer is %0.2f mm^2\n',Astring_internal(i)*10^6)

   for v=1:length(w)
       for t =1:length(h_L)
           for j = 1:length(ts)

             val = A_hat(v,j,t) == Astring_internal(i);
                if  val == true
               fprintf('\nL Stringer Dimensions\n')
               fprintf('Height is %0.2f mm \n',h_L(t)*10^3)
               fprintf('Thickness is %0.2f for mm\n',ts(j)*10^3)
               fprintf('The skin thickness is %0.2f mm\n',tw(k)*10^3)
                end

            val1 = A_hat(v,j,t) == Astring_internal(i);
            if  val1 == true
           fprintf('\nHat Stringer Dimensions\n')
           fprintf('Total Height is %0.2f mm \n',(h_L(t)+ts(j))*10^3)
           fprintf('Width Hat is %0.2f mm\n',(w(v)+2*ts(j))*10^3)
           fprintf('Width of L is %0.2f mm\n',h_L(t)*10^3)
           fprintf('Thickness is %0.2f for mm\n',ts(j)*10^3)
           fprintf('The skin thickness is %0.2f mm\n',tw(k)*10^3)
           fprintf('-----------------------------------------------\n')

            end

           end
       end
   end
   end

 end
```

```
Stringers on the Corner and four internally
Wing Box Weight = 142.87 kg
```

```
Safety Factor is 1.50

Stringer Information
Area of L Stringer is 165.00 mm^2
Area of Hat Stringer is 670.00 mm^2

L Stringer Dimensions
Height is 20.00 mm
Thickness is 6.62 for mm
The skin thickness is 1.00 mm

Hat Stringer Dimensions
Total Height is 26.62 mm
Width Hat is 27.25 mm
Width of L is 20.00 mm
Thickness is 6.62 for mm
The skin thickness is 1.00 mm
-----------------------------------------------
```

Published with MATLAB® R2021b

Optimization Code - Composite:

MAE 526 Design Project - Group 2 - Part 2: Composite Design
Aishwarya Ledalla, Alexander Scalco, Lakshya Tiwari, Rohan Ravishekar

```
clc;clear;
```

Material Properties & Laminate (Unidirectional Carbon Fiber/Epoxy)

```
Vf = [0.1:0.1:0.9];                     % fiber volume fraction
Vm = 1 - Vf;                            % matrix volume fraction
Em = 3.34; Ef = 233;                    % (GPa) Young's Modulus
v12m = 0.321; v12f = 0.2;
G12m = 2.1; G12f = 8.963;               % (GPa) Shear Modulus
E1 = Ef.*Vf + Em.*Vm;                   % GPa
E2 = Ef.*Em./(Vm.*Ef + Vf.*Em);         % GPa
G12 = G12f*G12m./(Vm*G12f + Vf*G12m);   % GPa
v12 = v12m.*Vm + v12f.*Vf;
theta = [-45,0,45,90];                  % degrees
NM = [0;0;0;8.9503e-5;0;0];             % N, GNm
pf = 1760;                              % kg/m3
pm = 1440;                              % kg/m3
pc = (pf.*Vf + pm.*Vm)';                % kg/m3

Lwing = 0.404;           % (m) total length of box fitted inside airfoil
twing = 0.049;           % (m) thickness of box fitted inside airfoil
s = 10;                  % (m) half span length
Awing = Lwing*twing      % (m^2) cross-section of the airfoil
```

```
% Tasi-Hill
s1 = 1.447;  % GPa
s2 = 0.0517; % GPa
t12 = 0.93;  % GPa
f = 1.5;     % factor of safety

% Weight
M = Awing.*s.*pc;  % (kg) weight of the laminate
t = [0.049; 0.035; 0.025; 0.020; 0.017; 0.015]; % (m) Varying widths

% Laminate Stiffness Matrix (GPa)
for i = 1:length(Vf)
Q_45(:,3*i-2:i*3)= OrthoMatrix_Offaxis(theta(1),E1(i),E2(i),v12(i),G12(i));
Q0(:,3*i-2:i*3) = OrthoMatrix_Offaxis(theta(2),E1(i),E2(i),v12(i),G12(i));
Q45(:,3*i-2:i*3) = OrthoMatrix_Offaxis(theta(3),E1(i),E2(i),v12(i),G12(i));
Q90(:,3*i-2:i*3) = OrthoMatrix_Offaxis(theta(4),E1(i),E2(i),v12(i),G12(i));
end
TM_45 = TransMatrix(theta(1));
TM0 = TransMatrix(theta(2));
TM45 = TransMatrix(theta(3));
TM90 = TransMatrix(theta(4));
```

```
Awing =

    0.0198
```

[90/0]

```
% Plies
Q = [Q90; Q0]; % GPa
TM = [TM90; TM0;];
P = size(Q,1)/3;                      % number of plies
t = twing/P                           % thickness of each ply
zk = linspace(-t*(P/2),t*(P/2),P)/2; % m

% ABD Matrix
M_ABD_1 = ABD(t,Q,1,3);
M_ABD_2 = ABD(t,Q,4,6);
M_ABD_3 = ABD(t,Q,7,9);
M_ABD_4 = ABD(t,Q,10,12);
M_ABD_5 = ABD(t,Q,13,15);
M_ABD_6 = ABD(t,Q,16,18);
M_ABD_7 = ABD(t,Q,19,21);
M_ABD_8 = ABD(t,Q,22,24);
M_ABD_9 = ABD(t,Q,25,27);

% Strains and Curvature
e0k_1 = (M_ABD_1^(-1))*NM;
e0k_2 = (M_ABD_2^(-1))*NM;
e0k_3 = (M_ABD_3^(-1))*NM;
```

```
e0k_4 = (M_ABD_4^(-1))*NM;
e0k_5 = (M_ABD_5^(-1))*NM;
e0k_6 = (M_ABD_6^(-1))*NM;
e0k_7 = (M_ABD_7^(-1))*NM;
e0k_8 = (M_ABD_8^(-1))*NM;
e0k_9 = (M_ABD_9^(-1))*NM;

% Stresses
for i = 1:P
    e_xy_1(:,i)= e0k_1(1:3) + zk(i).*e0k_1(4:6);
    e_xy_2(:,i)= e0k_2(1:3) + zk(i).*e0k_2(4:6);
    e_xy_3(:,i)= e0k_3(1:3) + zk(i).*e0k_3(4:6);
    e_xy_4(:,i)= e0k_4(1:3) + zk(i).*e0k_4(4:6);
    e_xy_5(:,i)= e0k_5(1:3) + zk(i).*e0k_5(4:6);
    e_xy_6(:,i)= e0k_6(1:3) + zk(i).*e0k_6(4:6);
    e_xy_7(:,i)= e0k_7(1:3) + zk(i).*e0k_7(4:6);
    e_xy_8(:,i)= e0k_8(1:3) + zk(i).*e0k_8(4:6);
    e_xy_9(:,i)= e0k_9(1:3) + zk(i).*e0k_9(4:6);

    sigma_xy_1(:,i) = Q(3*i-2:i*3,1:3)*e_xy_1(:,i);        % GPa
    sigma_xy_2(:,i) = Q(3*i-2:i*3,1:3)*e_xy_2(:,i);        % GPa
    sigma_xy_3(:,i) = Q(3*i-2:i*3,1:3)*e_xy_3(:,i);        % GPa
    sigma_xy_4(:,i) = Q(3*i-2:i*3,1:3)*e_xy_4(:,i);        % GPa
    sigma_xy_5(:,i) = Q(3*i-2:i*3,1:3)*e_xy_5(:,i);        % GPa
    sigma_xy_6(:,i) = Q(3*i-2:i*3,1:3)*e_xy_6(:,i);        % GPa
    sigma_xy_7(:,i) = Q(3*i-2:i*3,1:3)*e_xy_7(:,i);        % GPa
    sigma_xy_8(:,i) = Q(3*i-2:i*3,1:3)*e_xy_8(:,i);        % GPa
    sigma_xy_9(:,i) = Q(3*i-2:i*3,1:3)*e_xy_9(:,i);        % GPa

    sigma_12_1(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_1(:,i);   % GPa
    sigma_12_2(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_2(:,i);   % GPa
    sigma_12_3(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_3(:,i);   % GPa
    sigma_12_4(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_4(:,i);   % GPa
    sigma_12_5(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_5(:,i);   % GPa
    sigma_12_6(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_6(:,i);   % GPa
    sigma_12_7(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_7(:,i);   % GPa
    sigma_12_8(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_8(:,i);   % GPa
    sigma_12_9(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_9(:,i);   % GPa
end

sig_1 = f.*[sigma_12_1(1,:); sigma_12_2(1,:); sigma_12_3(1,:); ...
         sigma_12_4(1,:); sigma_12_5(1,:); sigma_12_6(1,:); ...
         sigma_12_7(1,:); sigma_12_8(1,:); sigma_12_9(1,:)];
sig_2 = f.*[sigma_12_1(2,:); sigma_12_2(2,:); sigma_12_3(2,:); ...
         sigma_12_4(2,:); sigma_12_5(2,:); sigma_12_6(2,:); ...
         sigma_12_7(2,:); sigma_12_8(2,:); sigma_12_9(2,:)];
tau_12 = f.*[sigma_12_1(3,:); sigma_12_2(3,:); sigma_12_3(3,:); ...
         sigma_12_4(3,:); sigma_12_5(3,:); sigma_12_6(3,:); ...
         sigma_12_7(3,:); sigma_12_8(3,:); sigma_12_9(3,:)];

n = ((sig_1./s1).^2 - sig_1.*sig_2./(s1^2) + (sig_2./s2).^2 + ...
    (tau_12./t12).^2)
```

```
t =

    0.0245


n =

    5.9708    0.0076
    3.1785    0.0015
    1.9157    0.0005
    1.2398    0.0003
    0.8329    0.0002
    0.5642    0.0001
    0.3713    0.0001
    0.2206    0.0001
    0.0937    0.0001
```

[90/0/90]

```
% Plies
Q = [Q90; Q0; Q90]; % GPa
TM = [TM90; TM0; TM90];
P = size(Q,1)/3;                    % number of plies
t = twing/P                         % thickness of each ply
zk = linspace(-t*(P/2),t*(P/2),P)/2; % m

% ABD Matrix
TM = [TM90; TM45; TM0; TM0; TM45; TM90];
M_ABD_1 = ABD(t,Q,1,3);
M_ABD_2 = ABD(t,Q,4,6);
M_ABD_3 = ABD(t,Q,7,9);
M_ABD_4 = ABD(t,Q,10,12);
M_ABD_5 = ABD(t,Q,13,15);
M_ABD_6 = ABD(t,Q,16,18);
M_ABD_7 = ABD(t,Q,19,21);
M_ABD_8 = ABD(t,Q,22,24);
M_ABD_9 = ABD(t,Q,25,27);

% Strains and Curvature
e0k_1 = (M_ABD_1^(-1))*NM;
e0k_2 = (M_ABD_2^(-1))*NM;
e0k_3 = (M_ABD_3^(-1))*NM;
e0k_4 = (M_ABD_4^(-1))*NM;
e0k_5 = (M_ABD_5^(-1))*NM;
e0k_6 = (M_ABD_6^(-1))*NM;
e0k_7 = (M_ABD_7^(-1))*NM;
e0k_8 = (M_ABD_8^(-1))*NM;
e0k_9 = (M_ABD_9^(-1))*NM;

% Stresses
for i = 1:P
```

```
    e_xy_1(:,i)= e0k_1(1:3) + zk(i).*e0k_1(4:6);
    e_xy_2(:,i)= e0k_2(1:3) + zk(i).*e0k_2(4:6);
    e_xy_3(:,i)= e0k_3(1:3) + zk(i).*e0k_3(4:6);
    e_xy_4(:,i)= e0k_4(1:3) + zk(i).*e0k_4(4:6);
    e_xy_5(:,i)= e0k_5(1:3) + zk(i).*e0k_5(4:6);
    e_xy_6(:,i)= e0k_6(1:3) + zk(i).*e0k_6(4:6);
    e_xy_7(:,i)= e0k_7(1:3) + zk(i).*e0k_7(4:6);
    e_xy_8(:,i)= e0k_8(1:3) + zk(i).*e0k_8(4:6);
    e_xy_9(:,i)= e0k_9(1:3) + zk(i).*e0k_9(4:6);

    sigma_xy_1(:,i) = Q(3*i-2:i*3,1:3)*e_xy_1(:,i);        % GPa
    sigma_xy_2(:,i) = Q(3*i-2:i*3,1:3)*e_xy_2(:,i);        % GPa
    sigma_xy_3(:,i) = Q(3*i-2:i*3,1:3)*e_xy_3(:,i);        % GPa
    sigma_xy_4(:,i) = Q(3*i-2:i*3,1:3)*e_xy_4(:,i);        % GPa
    sigma_xy_5(:,i) = Q(3*i-2:i*3,1:3)*e_xy_5(:,i);        % GPa
    sigma_xy_6(:,i) = Q(3*i-2:i*3,1:3)*e_xy_6(:,i);        % GPa
    sigma_xy_7(:,i) = Q(3*i-2:i*3,1:3)*e_xy_7(:,i);        % GPa
    sigma_xy_8(:,i) = Q(3*i-2:i*3,1:3)*e_xy_8(:,i);        % GPa
    sigma_xy_9(:,i) = Q(3*i-2:i*3,1:3)*e_xy_9(:,i);        % GPa

    sigma_12_1(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_1(:,i);   % GPa
    sigma_12_2(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_2(:,i);   % GPa
    sigma_12_3(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_3(:,i);   % GPa
    sigma_12_4(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_4(:,i);   % GPa
    sigma_12_5(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_5(:,i);   % GPa
    sigma_12_6(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_6(:,i);   % GPa
    sigma_12_7(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_7(:,i);   % GPa
    sigma_12_8(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_8(:,i);   % GPa
    sigma_12_9(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_9(:,i);   % GPa
end

sig_1 = f.*[sigma_12_1(1,:); sigma_12_2(1,:); sigma_12_3(1,:); ...
         sigma_12_4(1,:); sigma_12_5(1,:); sigma_12_6(1,:); ...
         sigma_12_7(1,:); sigma_12_8(1,:); sigma_12_9(1,:)];
sig_2 = f.*[sigma_12_1(2,:); sigma_12_2(2,:); sigma_12_3(2,:); ...
         sigma_12_4(2,:); sigma_12_5(2,:); sigma_12_6(2,:); ...
         sigma_12_7(2,:); sigma_12_8(2,:); sigma_12_9(2,:)];
tau_12 = f.*[sigma_12_1(3,:); sigma_12_2(3,:); sigma_12_3(3,:); ...
         sigma_12_4(3,:); sigma_12_5(3,:); sigma_12_6(3,:); ...
         sigma_12_7(3,:); sigma_12_8(3,:); sigma_12_9(3,:)];

n = ((sig_1./s1).^2 - sig_1.*sig_2./(s1^2) + (sig_2./s2).^2 + ...
    (tau_12./t12).^2)
```

```
t =

    0.0163


n =

    7.0025         0    0.0090
```

```
    4.2491         0     0.0117
    2.7545         0     0.0115
    1.8505         0     0.0094
    1.2576         0     0.0072
    0.8421         0     0.0052
    0.5338         0     0.0035
    0.2948         0     0.0020
    0.1099         0     0.0008
```

[90/0]s

```
% Plies
Q = [Q90; Q0; Q0; Q90]; % GPa
TM = [TM90; TM0; TM0; TM90];
P = size(Q,1)/3;                     % number of plies
t = twing/P                          % thickness of each ply
zk = linspace(-t*(P/2),t*(P/2),P)/2; % m

% ABD Matrix
M_ABD_1 = ABD(t,Q,1,3);
M_ABD_2 = ABD(t,Q,4,6);
M_ABD_3 = ABD(t,Q,7,9);
M_ABD_4 = ABD(t,Q,10,12);
M_ABD_5 = ABD(t,Q,13,15);
M_ABD_6 = ABD(t,Q,16,18);
M_ABD_7 = ABD(t,Q,19,21);
M_ABD_8 = ABD(t,Q,22,24);
M_ABD_9 = ABD(t,Q,25,27);

% Strains and Curvature
e0k_1 = (M_ABD_1^(-1))*NM;
e0k_2 = (M_ABD_2^(-1))*NM;
e0k_3 = (M_ABD_3^(-1))*NM;
e0k_4 = (M_ABD_4^(-1))*NM;
e0k_5 = (M_ABD_5^(-1))*NM;
e0k_6 = (M_ABD_6^(-1))*NM;
e0k_7 = (M_ABD_7^(-1))*NM;
e0k_8 = (M_ABD_8^(-1))*NM;
e0k_9 = (M_ABD_9^(-1))*NM;

% Stresses
for i = 1:P
    e_xy_1(:,i)= e0k_1(1:3) + zk(i).*e0k_1(4:6);
    e_xy_2(:,i)= e0k_2(1:3) + zk(i).*e0k_2(4:6);
    e_xy_3(:,i)= e0k_3(1:3) + zk(i).*e0k_3(4:6);
    e_xy_4(:,i)= e0k_4(1:3) + zk(i).*e0k_4(4:6);
    e_xy_5(:,i)= e0k_5(1:3) + zk(i).*e0k_5(4:6);
    e_xy_6(:,i)= e0k_6(1:3) + zk(i).*e0k_6(4:6);
    e_xy_7(:,i)= e0k_7(1:3) + zk(i).*e0k_7(4:6);
    e_xy_8(:,i)= e0k_8(1:3) + zk(i).*e0k_8(4:6);
    e_xy_9(:,i)= e0k_9(1:3) + zk(i).*e0k_9(4:6);
```

```
    sigma_xy_1(:,i) = Q(3*i-2:i*3,1:3)*e_xy_1(:,i);        % GPa
    sigma_xy_2(:,i) = Q(3*i-2:i*3,1:3)*e_xy_2(:,i);        % GPa
    sigma_xy_3(:,i) = Q(3*i-2:i*3,1:3)*e_xy_3(:,i);        % GPa
    sigma_xy_4(:,i) = Q(3*i-2:i*3,1:3)*e_xy_4(:,i);        % GPa
    sigma_xy_5(:,i) = Q(3*i-2:i*3,1:3)*e_xy_5(:,i);        % GPa
    sigma_xy_6(:,i) = Q(3*i-2:i*3,1:3)*e_xy_6(:,i);        % GPa
    sigma_xy_7(:,i) = Q(3*i-2:i*3,1:3)*e_xy_7(:,i);        % GPa
    sigma_xy_8(:,i) = Q(3*i-2:i*3,1:3)*e_xy_8(:,i);        % GPa
    sigma_xy_9(:,i) = Q(3*i-2:i*3,1:3)*e_xy_9(:,i);        % GPa

    sigma_12_1(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_1(:,i);   % GPa
    sigma_12_2(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_2(:,i);   % GPa
    sigma_12_3(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_3(:,i);   % GPa
    sigma_12_4(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_4(:,i);   % GPa
    sigma_12_5(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_5(:,i);   % GPa
    sigma_12_6(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_6(:,i);   % GPa
    sigma_12_7(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_7(:,i);   % GPa
    sigma_12_8(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_8(:,i);   % GPa
    sigma_12_9(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_9(:,i);   % GPa
end

sig_1 = f.*[sigma_12_1(1,:); sigma_12_2(1,:); sigma_12_3(1,:); ...
         sigma_12_4(1,:); sigma_12_5(1,:); sigma_12_6(1,:); ...
         sigma_12_7(1,:); sigma_12_8(1,:); sigma_12_9(1,:)];
sig_2 = f.*[sigma_12_1(2,:); sigma_12_2(2,:); sigma_12_3(2,:); ...
         sigma_12_4(2,:); sigma_12_5(2,:); sigma_12_6(2,:); ...
         sigma_12_7(2,:); sigma_12_8(2,:); sigma_12_9(2,:)];
tau_12 = f.*[sigma_12_1(3,:); sigma_12_2(3,:); sigma_12_3(3,:); ...
         sigma_12_4(3,:); sigma_12_5(3,:); sigma_12_6(3,:); ...
         sigma_12_7(3,:); sigma_12_8(3,:); sigma_12_9(3,:)];

n = ((sig_1./s1).^2 - sig_1.*sig_2./(s1^2) + (sig_2./s2).^2 + ...
    (tau_12./t12).^2)
```

```
t =

    0.0123


n =

    3.3860    0.0259    0.0259    3.3860
    1.5040    0.0116    0.0116    1.5040
    0.8316    0.0065    0.0065    0.8316
    0.5164    0.0040    0.0040    0.5164
    0.3426    0.0027    0.0027    0.3426
    0.2350    0.0018    0.0018    0.2350
    0.1612    0.0013    0.0013    0.1612
    0.1044    0.0008    0.0008    0.1044
    0.0532    0.0004    0.0004    0.0532
```

[45/-45]s

```
% Plies
Q = [Q45; Q_45; Q_45; Q45]; % GPa
TM = [TM45; TM_45; TM_45; TM45];
P = size(Q,1)/3;                    % number of plies
t = twing/P                         % thickness of each ply
zk = linspace(-t*(P/2),t*(P/2),P)/2; % m

% ABD Matrix
M_ABD_1 = ABD(t,Q,1,3);
M_ABD_2 = ABD(t,Q,4,6);
M_ABD_3 = ABD(t,Q,7,9);
M_ABD_4 = ABD(t,Q,10,12);
M_ABD_5 = ABD(t,Q,13,15);
M_ABD_6 = ABD(t,Q,16,18);
M_ABD_7 = ABD(t,Q,19,21);
M_ABD_8 = ABD(t,Q,22,24);
M_ABD_9 = ABD(t,Q,25,27);

% Strains and Curvature
e0k_1 = (M_ABD_1^(-1))*NM;
e0k_2 = (M_ABD_2^(-1))*NM;
e0k_3 = (M_ABD_3^(-1))*NM;
e0k_4 = (M_ABD_4^(-1))*NM;
e0k_5 = (M_ABD_5^(-1))*NM;
e0k_6 = (M_ABD_6^(-1))*NM;
e0k_7 = (M_ABD_7^(-1))*NM;
e0k_8 = (M_ABD_8^(-1))*NM;
e0k_9 = (M_ABD_9^(-1))*NM;

% Stresses
for i = 1:P
    e_xy_1(:,i)= e0k_1(1:3) + zk(i).*e0k_1(4:6);
    e_xy_2(:,i)= e0k_2(1:3) + zk(i).*e0k_2(4:6);
    e_xy_3(:,i)= e0k_3(1:3) + zk(i).*e0k_3(4:6);
    e_xy_4(:,i)= e0k_4(1:3) + zk(i).*e0k_4(4:6);
    e_xy_5(:,i)= e0k_5(1:3) + zk(i).*e0k_5(4:6);
    e_xy_6(:,i)= e0k_6(1:3) + zk(i).*e0k_6(4:6);
    e_xy_7(:,i)= e0k_7(1:3) + zk(i).*e0k_7(4:6);
    e_xy_8(:,i)= e0k_8(1:3) + zk(i).*e0k_8(4:6);
    e_xy_9(:,i)= e0k_9(1:3) + zk(i).*e0k_9(4:6);

    sigma_xy_1(:,i) = Q(3*i-2:i*3,1:3)*e_xy_1(:,i);        % GPa
    sigma_xy_2(:,i) = Q(3*i-2:i*3,1:3)*e_xy_2(:,i);        % GPa
    sigma_xy_3(:,i) = Q(3*i-2:i*3,1:3)*e_xy_3(:,i);        % GPa
    sigma_xy_4(:,i) = Q(3*i-2:i*3,1:3)*e_xy_4(:,i);        % GPa
    sigma_xy_5(:,i) = Q(3*i-2:i*3,1:3)*e_xy_5(:,i);        % GPa
    sigma_xy_6(:,i) = Q(3*i-2:i*3,1:3)*e_xy_6(:,i);        % GPa
    sigma_xy_7(:,i) = Q(3*i-2:i*3,1:3)*e_xy_7(:,i);        % GPa
    sigma_xy_8(:,i) = Q(3*i-2:i*3,1:3)*e_xy_8(:,i);        % GPa
    sigma_xy_9(:,i) = Q(3*i-2:i*3,1:3)*e_xy_9(:,i);        % GPa
```

```
    sigma_12_1(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_1(:,i);   % GPa
    sigma_12_2(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_2(:,i);   % GPa
    sigma_12_3(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_3(:,i);   % GPa
    sigma_12_4(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_4(:,i);   % GPa
    sigma_12_5(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_5(:,i);   % GPa
    sigma_12_6(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_6(:,i);   % GPa
    sigma_12_7(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_7(:,i);   % GPa
    sigma_12_8(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_8(:,i);   % GPa
    sigma_12_9(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_9(:,i);   % GPa
end

sig_1 = f.*[sigma_12_1(1,:); sigma_12_2(1,:); sigma_12_3(1,:); ...
         sigma_12_4(1,:); sigma_12_5(1,:); sigma_12_6(1,:); ...
         sigma_12_7(1,:); sigma_12_8(1,:); sigma_12_9(1,:)];
sig_2 = f.*[sigma_12_1(2,:); sigma_12_2(2,:); sigma_12_3(2,:); ...
         sigma_12_4(2,:); sigma_12_5(2,:); sigma_12_6(2,:); ...
         sigma_12_7(2,:); sigma_12_8(2,:); sigma_12_9(2,:)];
tau_12 = f.*[sigma_12_1(3,:); sigma_12_2(3,:); sigma_12_3(3,:); ...
         sigma_12_4(3,:); sigma_12_5(3,:); sigma_12_6(3,:); ...
         sigma_12_7(3,:); sigma_12_8(3,:); sigma_12_9(3,:)];

n = ((sig_1./s1).^2 - sig_1.*sig_2./(s1^2) + (sig_2./s2).^2 + ...
    (tau_12./t12).^2)
```

```
t =

    0.0123


n =

    0.8737    0.0179    0.0179    0.8737
    0.3924    0.0071    0.0071    0.3924
    0.2191    0.0039    0.0039    0.2191
    0.1373    0.0025    0.0025    0.1373
    0.0918    0.0017    0.0017    0.0918
    0.0633    0.0012    0.0012    0.0633
    0.0437    0.0009    0.0009    0.0437
    0.0285    0.0006    0.0006    0.0285
    0.0148    0.0004    0.0004    0.0148
```

[90/45/-45/90]

```
% Plies
Q = [Q90; Q45; Q_45; Q90]; % GPa
TM = [TM90; TM45; TM_45; TM90];
P = size(Q,1)/3;                    % number of plies
t = twing/P                         % thickness of each ply
zk = linspace(-t*(P/2),t*(P/2),P)/2; % m

% ABD Matrix
```

```
M_ABD_1 = ABD(t,Q,1,3);
M_ABD_2 = ABD(t,Q,4,6);
M_ABD_3 = ABD(t,Q,7,9);
M_ABD_4 = ABD(t,Q,10,12);
M_ABD_5 = ABD(t,Q,13,15);
M_ABD_6 = ABD(t,Q,16,18);
M_ABD_7 = ABD(t,Q,19,21);
M_ABD_8 = ABD(t,Q,22,24);
M_ABD_9 = ABD(t,Q,25,27);

% Strains and Curvature
e0k_1 = (M_ABD_1^(-1))*NM;
e0k_2 = (M_ABD_2^(-1))*NM;
e0k_3 = (M_ABD_3^(-1))*NM;
e0k_4 = (M_ABD_4^(-1))*NM;
e0k_5 = (M_ABD_5^(-1))*NM;
e0k_6 = (M_ABD_6^(-1))*NM;
e0k_7 = (M_ABD_7^(-1))*NM;
e0k_8 = (M_ABD_8^(-1))*NM;
e0k_9 = (M_ABD_9^(-1))*NM;

% Stresses
for i = 1:P
    e_xy_1(:,i)= e0k_1(1:3) + zk(i).*e0k_1(4:6);
    e_xy_2(:,i)= e0k_2(1:3) + zk(i).*e0k_2(4:6);
    e_xy_3(:,i)= e0k_3(1:3) + zk(i).*e0k_3(4:6);
    e_xy_4(:,i)= e0k_4(1:3) + zk(i).*e0k_4(4:6);
    e_xy_5(:,i)= e0k_5(1:3) + zk(i).*e0k_5(4:6);
    e_xy_6(:,i)= e0k_6(1:3) + zk(i).*e0k_6(4:6);
    e_xy_7(:,i)= e0k_7(1:3) + zk(i).*e0k_7(4:6);
    e_xy_8(:,i)= e0k_8(1:3) + zk(i).*e0k_8(4:6);
    e_xy_9(:,i)= e0k_9(1:3) + zk(i).*e0k_9(4:6);

    sigma_xy_1(:,i) = Q(3*i-2:i*3,1:3)*e_xy_1(:,i);        % GPa
    sigma_xy_2(:,i) = Q(3*i-2:i*3,1:3)*e_xy_2(:,i);        % GPa
    sigma_xy_3(:,i) = Q(3*i-2:i*3,1:3)*e_xy_3(:,i);        % GPa
    sigma_xy_4(:,i) = Q(3*i-2:i*3,1:3)*e_xy_4(:,i);        % GPa
    sigma_xy_5(:,i) = Q(3*i-2:i*3,1:3)*e_xy_5(:,i);        % GPa
    sigma_xy_6(:,i) = Q(3*i-2:i*3,1:3)*e_xy_6(:,i);        % GPa
    sigma_xy_7(:,i) = Q(3*i-2:i*3,1:3)*e_xy_7(:,i);        % GPa
    sigma_xy_8(:,i) = Q(3*i-2:i*3,1:3)*e_xy_8(:,i);        % GPa
    sigma_xy_9(:,i) = Q(3*i-2:i*3,1:3)*e_xy_9(:,i);        % GPa

    sigma_12_1(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_1(:,i);   % GPa
    sigma_12_2(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_2(:,i);   % GPa
    sigma_12_3(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_3(:,i);   % GPa
    sigma_12_4(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_4(:,i);   % GPa
    sigma_12_5(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_5(:,i);   % GPa
    sigma_12_6(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_6(:,i);   % GPa
    sigma_12_7(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_7(:,i);   % GPa
    sigma_12_8(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_8(:,i);   % GPa
    sigma_12_9(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_9(:,i);   % GPa
end
```

```
sig_1 = f.*[sigma_12_1(1,:); sigma_12_2(1,:); sigma_12_3(1,:); ...
         sigma_12_4(1,:); sigma_12_5(1,:); sigma_12_6(1,:); ...
         sigma_12_7(1,:); sigma_12_8(1,:); sigma_12_9(1,:)];
sig_2 = f.*[sigma_12_1(2,:); sigma_12_2(2,:); sigma_12_3(2,:); ...
         sigma_12_4(2,:); sigma_12_5(2,:); sigma_12_6(2,:); ...
         sigma_12_7(2,:); sigma_12_8(2,:); sigma_12_9(2,:)];
tau_12 = f.*[sigma_12_1(3,:); sigma_12_2(3,:); sigma_12_3(3,:); ...
         sigma_12_4(3,:); sigma_12_5(3,:); sigma_12_6(3,:); ...
         sigma_12_7(3,:); sigma_12_8(3,:); sigma_12_9(3,:)];

n = ((sig_1./s1).^2 - sig_1.*sig_2./(s1^2) + (sig_2./s2).^2 + ...
    (tau_12./t12).^2)
```

```
t =

    0.0123


n =

    8.3340    0.7886    0.7886    8.3340
    6.0979    0.7017    0.7017    6.0979
    4.4808    0.5564    0.5564    4.4808
    3.2436    0.4173    0.4173    3.2436
    2.2750    0.2970    0.2970    2.2750
    1.5107    0.1972    0.1972    1.5107
    0.9119    0.1171    0.1171    0.9119
    0.4583    0.0563    0.0563    0.4583
    0.1472    0.0160    0.0160    0.1472
```

[45/-45/45]s

```
% Plies
Q = [Q45; Q_45; Q45; Q45; Q_45; Q45]; % GPa
TM = [TM45; TM_45; TM45; TM45; TM_45; TM45];
P = size(Q,1)/3;                     % number of plies
t = twing/P                          % thickness of each ply
zk = linspace(-t*(P/2),t*(P/2),P)/2; % m

% ABD Matrix
M_ABD_1 = ABD(t,Q,1,3);
M_ABD_2 = ABD(t,Q,4,6);
M_ABD_3 = ABD(t,Q,7,9);
M_ABD_4 = ABD(t,Q,10,12);
M_ABD_5 = ABD(t,Q,13,15);
M_ABD_6 = ABD(t,Q,16,18);
M_ABD_7 = ABD(t,Q,19,21);
M_ABD_8 = ABD(t,Q,22,24);
M_ABD_9 = ABD(t,Q,25,27);

% Strains and Curvature
```

```
e0k_1 = (M_ABD_1^(-1))*NM;
e0k_2 = (M_ABD_2^(-1))*NM;
e0k_3 = (M_ABD_3^(-1))*NM;
e0k_4 = (M_ABD_4^(-1))*NM;
e0k_5 = (M_ABD_5^(-1))*NM;
e0k_6 = (M_ABD_6^(-1))*NM;
e0k_7 = (M_ABD_7^(-1))*NM;
e0k_8 = (M_ABD_8^(-1))*NM;
e0k_9 = (M_ABD_9^(-1))*NM;

% Stresses
for i = 1:P
    e_xy_1(:,i)= e0k_1(1:3) + zk(i).*e0k_1(4:6);
    e_xy_2(:,i)= e0k_2(1:3) + zk(i).*e0k_2(4:6);
    e_xy_3(:,i)= e0k_3(1:3) + zk(i).*e0k_3(4:6);
    e_xy_4(:,i)= e0k_4(1:3) + zk(i).*e0k_4(4:6);
    e_xy_5(:,i)= e0k_5(1:3) + zk(i).*e0k_5(4:6);
    e_xy_6(:,i)= e0k_6(1:3) + zk(i).*e0k_6(4:6);
    e_xy_7(:,i)= e0k_7(1:3) + zk(i).*e0k_7(4:6);
    e_xy_8(:,i)= e0k_8(1:3) + zk(i).*e0k_8(4:6);
    e_xy_9(:,i)= e0k_9(1:3) + zk(i).*e0k_9(4:6);

    sigma_xy_1(:,i) = Q(3*i-2:i*3,1:3)*e_xy_1(:,i);        % GPa
    sigma_xy_2(:,i) = Q(3*i-2:i*3,1:3)*e_xy_2(:,i);        % GPa
    sigma_xy_3(:,i) = Q(3*i-2:i*3,1:3)*e_xy_3(:,i);        % GPa
    sigma_xy_4(:,i) = Q(3*i-2:i*3,1:3)*e_xy_4(:,i);        % GPa
    sigma_xy_5(:,i) = Q(3*i-2:i*3,1:3)*e_xy_5(:,i);        % GPa
    sigma_xy_6(:,i) = Q(3*i-2:i*3,1:3)*e_xy_6(:,i);        % GPa
    sigma_xy_7(:,i) = Q(3*i-2:i*3,1:3)*e_xy_7(:,i);        % GPa
    sigma_xy_8(:,i) = Q(3*i-2:i*3,1:3)*e_xy_8(:,i);        % GPa
    sigma_xy_9(:,i) = Q(3*i-2:i*3,1:3)*e_xy_9(:,i);        % GPa

    sigma_12_1(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_1(:,i);   % GPa
    sigma_12_2(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_2(:,i);   % GPa
    sigma_12_3(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_3(:,i);   % GPa
    sigma_12_4(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_4(:,i);   % GPa
    sigma_12_5(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_5(:,i);   % GPa
    sigma_12_6(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_6(:,i);   % GPa
    sigma_12_7(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_7(:,i);   % GPa
    sigma_12_8(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_8(:,i);   % GPa
    sigma_12_9(:,i) = TM(3*i-2:i*3,1:3)*sigma_xy_9(:,i);   % GPa
end

sig_1 = f.*[sigma_12_1(1,:); sigma_12_2(1,:); sigma_12_3(1,:); ...
         sigma_12_4(1,:); sigma_12_5(1,:); sigma_12_6(1,:); ...
         sigma_12_7(1,:); sigma_12_8(1,:); sigma_12_9(1,:)];
sig_2 = f.*[sigma_12_1(2,:); sigma_12_2(2,:); sigma_12_3(2,:); ...
         sigma_12_4(2,:); sigma_12_5(2,:); sigma_12_6(2,:); ...
         sigma_12_7(2,:); sigma_12_8(2,:); sigma_12_9(2,:)];
tau_12 = f.*[sigma_12_1(3,:); sigma_12_2(3,:); sigma_12_3(3,:); ...
         sigma_12_4(3,:); sigma_12_5(3,:); sigma_12_6(3,:); ...
         sigma_12_7(3,:); sigma_12_8(3,:); sigma_12_9(3,:)];
```

```
n = ((sig_1./s1).^2 - sig_1.*sig_2./(s1^2) + (sig_2./s2).^2 + ...
    (tau_12./t12).^2)
```

```
t =

    0.0082


n =

    0.4254    0.0519    0.0170    0.0170    0.0519    0.4254
    0.1595    0.0183    0.0064    0.0064    0.0183    0.1595
    0.0834    0.0098    0.0033    0.0033    0.0098    0.0834
    0.0511    0.0062    0.0020    0.0020    0.0062    0.0511
    0.0342    0.0043    0.0014    0.0014    0.0043    0.0342
    0.0240    0.0031    0.0010    0.0010    0.0031    0.0240
    0.0172    0.0023    0.0007    0.0007    0.0023    0.0172
    0.0122    0.0016    0.0005    0.0005    0.0016    0.0122
    0.0077    0.0011    0.0003    0.0003    0.0011    0.0077
```

Functions

```
    type OrthoMatrix_Offaxis.m
    type TransMatrix.m
    type ABD.m
```

```
%% Orthotropic Off-axis Matrix

function Q_ = OrthoMatrix_Offaxis(theta,Q11,Q22,Q12,Q66)

% Theta in degrees

%% Trigonometry
    s = sind(theta);
    c = cosd(theta);

%% New Stiffness Coefficients
    q11 = Q11*c^4 + Q22*s^4 + 2*(Q12+2*Q66)*(s^2)*(c^2);
    q22 = Q11*s^4 + Q22*c^4 + 2*(Q12+2*Q66)*(s^2)*(c^2);
    q12 = (Q11 + Q22 - 4*Q66)*(s^2)*(c^2) + Q12*(c^4 + s^4);
    q66 = (Q11 + Q22 - 2*Q12- 2*Q66)*(s^2)*(c^2) + Q66*(s^4 + c^4);
    q16 = (Q11 - Q12 - 2*Q66)*(c^3)*s - (Q22 - Q12 - 2*Q66)*c*(s^3);
    q26 = (Q11 - Q12 - 2*Q66)*c*(s^3) - (Q22 - Q12 - 2*Q66)*(c^3)*s;

%% New stiffness Matrix
    Q_ = [q11 q12 q16; q12 q22 q26; q16 q26 q66];

end

%% Transformation Matrix
```

```
function T = TransMatrix(theta)

% Theta in degrees

%% Trigonometry
    s = sind(theta);
    c = cosd(theta);

%% Matrix
    T = [c^2 s^2 2*c*s; s^2 c^2 -2*c*s; -c*s c*s c^2-s^2];

end

%% ABD Matrix
function M = ABD(t,Q,v1,v2)

    P = size(Q,1)/3; % number of plies
    z = linspace(-t*(P/2),t*(P/2),P+1); % m

    % initialize A, B, and D
    A(:,:,1) = zeros(3,3); B(:,:,1) = A; D(:,:,1) = A;

    for p = 1:P
        A(:,:,p+1) = A(:,:,p) + Q(3*p-2:p*3,v1:v2)*(z(p+1)-z(p));
        B(:,:,p+1) = B(:,:,p)+(1/2)*Q(3*p-2:p*3,v1:v2)*((z(p+1))^2-(z(p))^2);
        D(:,:,p+1) = D(:,:,p)+(1/3)*Q(3*p-2:p*3,v1:v2)*((z(p+1))^3-(z(p))^3);
    end

    A = A(:,:,P+1); B = B(:,:,P+1); D = D(:,:,P+1); % Pa
    M = [A B; B D]; % Pa

end
```

Published with MATLAB® R2021b