Optimization of an RC Plane

Weston Pace

November 2022

1 Abstract

In this paper, I discuss the optimization process for an RC plane and also go through the analysis of how specific design variables affect the stability and efficiency of the aircraft. I use this analysis to validate the information I got from the optimization packaged that I used. Through the use of information obtained through isolated analysis of changing conditions on an aircraft, I was able to build a basis for what to expect an optimal plane might be even before the optimizer ran.

2 Introduction

The basis for this paper came from the assignments that I was given for my 497R class in the FLOW Lab at BYU. These assignments taught me how to use XFoil.jl, VortexLattice.jl, and SNOW.jl and the physical principles behind these packages. I wanted to extend some of the basic concepts that were over viewed in the assignments for my 497R and obtain a deeper understanding of how they apply to aircraft design and optimization. Additionally, I wanted to build intuition for how an aircraft is affected by changing specific components on the airframe. Having this intuition makes the optimization process more predictable and justifiable.

Another group who recently did an RC plane optimization was Joanneum Aeronautics who competed in the Design, Build, Fly competition for the 2021-2022 season. Their procedures were slightly different and more advanced than mine, but overall, the general principles were the same. Something different that they did was instead of starting with a random airframe configuration, they considered what requirements they needed from their plane for the competition and determined a reasonable initial design that they could work off of. An example of this was the decision to have a T-tail or conventional tail design. They weighted the pros and cons of different set ups and chose the most effective options that met their needs.

Once they had this design, they began to run simulations to test the aircraft's efficiency and stability. While I used Xfoil.jl and VortexLattice.jl for my analysis, they used XFLR5 instead. However, the results that they obtained from this program are obtainable in my chosen packages, as well.

A very different approach that they took compared to me was the analysis of the airfoil on their aircraft. I specifically focused on the airframe design without looking into the airfoil, which may have been something good to consider. In general, though, we were both looking for similar things in the aircraft. We both analyzed lift, drag, and the stability derivatives that affect different modes of flight. In this way, my optimization method for my RC plane was mostly in line with the principles of this professional organization. Granted, they went into much more depth and complexity with their analysis, but similarities in our processes are still note worthy.

3 Methods

In order to optimize the characteristics of an aircraft, it is crucial to have methods that solve for values that give critical information about how successful the aircraft will be. These values can be as simple as aerodynamic coefficients for lift, drag, and moments, but they can even go into the direction of stability derivatives and structural integrity. However, for this project I chose to keep the span of values relatively small and just focus on stability derivatives and coefficients.

Due to the previous assignments I did for my 497R, I learned how to use Xfoil.jl and VortexLattice.jl to find aerodynamic values. Though different in their solution methods, both of these programs are extremely useful for aircraft analysis.

Additionally, I used the optimization package SNOW.jl. Although a somewhat basic optimization problem like this could be done with some simple intuition, I chose to use SNOW because I felt like learning how to use an optimizer would be beneficial for future projects.

3.1 Vortex Lattice

VortexLattice.jl uses the principles of Lifting Line theory as the basis for its solution method. In brief, Lifting Line theory models the flow interacting with the body of an aircraft using vortex filaments. These filaments are simply lines containing an infinite amount of vortices that are used to describe the downwash effect and fluid movement that occurs with an aircraft. VortexLattice.jl is very useful because it not only describes the aerodynamic coefficients for a plane, but it can also give out the stability derivatives. For the purposes of this assignment, VortexLattice was the main package that I used for analysis.

An important thing to note about Lifting Line theory is that it is inviscid: meaning that fluid separation does not occur when using this analysis method. This means that stall is not calculated for, which can lead to misleading data for lift.

3.2 Xfoil

While VortexLattice.jl describes the flow around the whole airframe and uses vortex filaments, Xfoil.jl focuses on the flow around an airfoil. The main theory behind Xfoil.jl comes from Potential Flow theory. This theory is extremely useful for describing the velocity field of an irrotational, incompressible fluid. The principles from Potential Flow theory are carried over to Thin Airfoil theory, which allows the boundary conditions from an airfoil to be imposed in Potential Flow theory without interfering with the integrity of the solution method. If the airfoil dimensions are too big, then Thin Airfoil theory doesn't stand as well, and the results from it are not as legitimate as they would be for a relatively thin airfoil.

Thin Airfoil theory sets up an infinite line of point vortices and sources along the chord line in order to model the flow around an airfoil. This method works by finding a distribution of vortex and source strengths that satisfy the boundary conditions for an airfoil. Once these distributions have been found, the fluid velocity at each point around the airfoil can be solved for. This allows for a pressure distribution to be created, which in turn allows for the coefficients of lift, drag, and moment to be found.

An important characteristic of Xfoil.jl is that it actually includes viscosity in its solution method. This allows for fluid separation to be considered, which leads to stall being a factor in Xfoil.jl's solutions.

3.3 SNOW

As was stated previously, an optimizer package is not necessarily needed for a relatively simple problem like this, but I used it nonetheless for the experience.

I initially was using Optim.jl as my optimization software. It worked pretty well in the beginning, but as I made the problem more complicated, the efficiency of Optim.jl and its constraint process made it frustrating to use. After some discussion about other approaches to take, I ended up using SNOW.jl for optimization. SNOW.jl is a wrapper that was created by Doctor Ning and utilizes IPOPT as its main solution method. IPOPT stands for interior point optimizer and is used for solving linear and nonlinear convex optimization problems.

SNOW.jl was fantastic for this assignment because of its user friendliness and efficiency. Additionally, I learned a lot of good skills while working through SNOW.jl: such as how to set up constraint functions, how important initial conditions are for optimization, and how to build an effective objective function.

3.3.1 Optimization Problem

maximize:

lift to drag ratio

by varying:

wing sweep: $0.1 \le x_1 \le 0.5$

wing span (half): $0.5 \le x_2 \le 0.75$

wing dihedral: $0.01 \le x_3 \le 0.2$

horizontal stabilizer sweep: $0.2 \le x_4 \le 0.5$

horizontal stabilizer span (half): $0.3 \le x_5 \le 0.5$

horizontal stabilizer dihedral: $0.0 \le x_6 \le 0.05$

vertical stabilizer sweep: $0.1 \le x_7 \le 0.5$

vertical stabilizer span: $0.1 \le x_8 \le 0.5$

horizontal and vertical stabilizer position: $1.7 \le x_9 \le 2.5$

subject to:

lift force > 4.9 N

$C_{ma} < 0$

$C_{nb} > 0$

$C_{lb} < 0$

4 Results and Discussion

Through the use of Xfoil.jl and VortexLattice.jl, I was able to collect aircraft data while changing parameters relating to airframe geometry and freestream conditions. Understanding how different parameters affect aerodynamic coefficients and stability aided me in the optimization process since it helped me legitimize my results.

4.1 Inviscid Efficiency

It is desirable for an aircraft to be able to produce a high amount a lift with respect to drag. This relationship is often called a lift-to-drag ratio, but another way that it can be represented is called inviscid efficiency. Inviscid efficiency is represented by the equation:

$$e = \frac{C_L^2}{\pi(AR)C_D}$$

where AR is the aspect ratio.

$AR = \frac{b^2}{S}$;

b = wing span;

S = wing area

What this shows is that a higher efficiency corresponds to a higher coefficient of lift compared to drag for the aspect ratio being held constant.

With the help of VortexLattice.jl, I was able to find how the efficiency changed with respect to the aspect ratio. This was important because it showed how the efficiency of the aircraft changed with adjustments to the wing geometry.

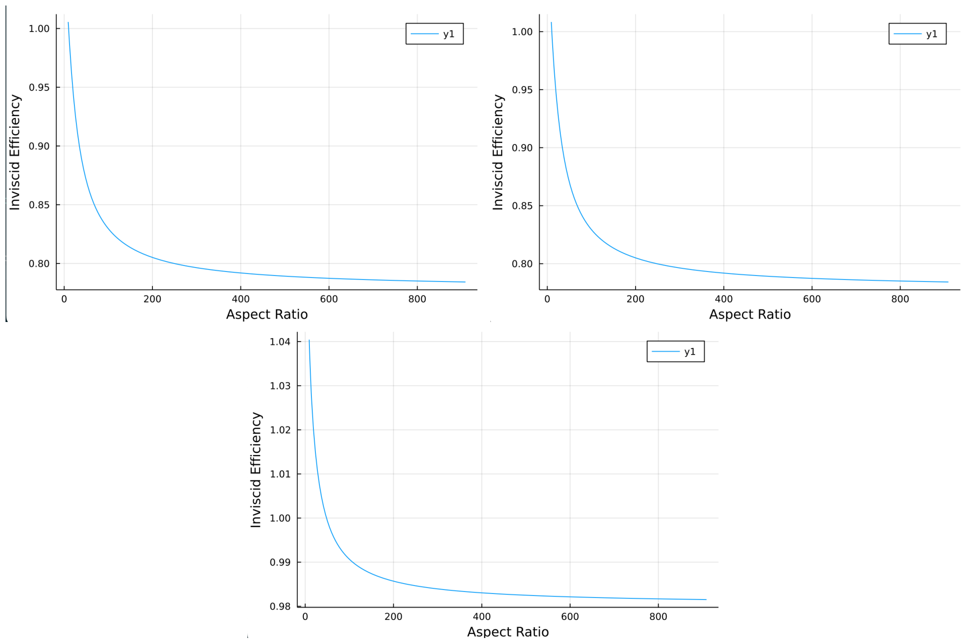

Figure 1: These all had aspect ratio change with respect to the span of the wing. The top left is of a Hershey bar wing; the top right is a dihedral Hershey bar wing; the bottom is a non-Hershey bar dihedral wing

The important thing to notice about figure 1 is that as the aspect ration of a wing increases, the efficiency of the aircraft decreases. Additionally, an aircraft with dihedral can achieve a higher efficiency at higher aspect ratios than a planar wing. In terms of optimization, this means that adding dihedral to a wing can help with the lift-to-drag ratio of my aircraft, and it also implies that there is balance that must be met between the chord length and the span of the wing in order to reduce the aspect ratio to a lower value. Since I didn't make the chord lengths design variables in my optimization, I had the ability to choose what I wanted them to be. Since aspect ratio should be reduced in order to increase efficiency, it would be in the best interest to increase the reference area in the aspect ratio equation, so the numerator term gets divided by a greater value and reduce the aspect ratio. A larger chord length would be ideal, but since I'm not considering the internal forces that could arise within a very large wing, it is my job to use a little intuition to choose a chord length that decreases aspect ratio but also would be integrious.

4.2 Stability Derivatives

Stability derivatives are extremely important for an aircraft. They are the difference between whether a plane can stabilize itself when undesired motion arises or if it will continuously get more out of control until it falls out of the sky. Stability derivatives come with a real and imaginary component, and as differential equations show, the real part reflects the dampening aspect while the imaginary part shows the rate of oscillation of the motion. Based off what stability derivative is being evaluated, the real component needs to either be positive or negative for the aircraft to stabilize itself. The basis for whether the sign should be positive or negative comes from the coordinate system assigned to the aircraft.

VortexLattice automatically only gives the real component of the derivatives, and for me, this was more than enough for determining whether my aircraft would be stable or not.

There are quite a few stability derivatives for an aircraft, but I wanted to only focus on a few of them due to my skill level in this field. I decided to pick three derivatives that I felt were applicable to an RC plane and the type of flight that it would see. I decided on three of them, and they were C_{ma}, C_{nb}, and C_{lb}.

C_{ma} deals with the pitching moment of an aircraft as the angle of attack changes, and it is useful in determining the stability in a Phugoid and short oscillation mode of flight. This derivative should be negative.

C_{nb} deals with the yaw moment created as the side slip angle of an aircraft increases. It is good for determining the stability in a slow spiral mode. This derivative should be positive due to the coordinate system of the side slip angle and the yaw moment.

C_{lb} is a derivative that shows the stability for the rolling moment as the side slip angle changes. This is extremely important for seeing how a plane reacts to a dutch roll mode of flight. This derivative should be negative.

4.2.1 Horizontal Tail Volume Ratio's Effect on Derivatives

Horizontal tail ratio is a nondimensionalized value that relates the size and position of the horizontal stabilizer to the size and position of the wing. The equation to represent this is given by:

$$\text{Horizontal Tail Volume Ratio} = \frac{S_t x_t}{S_w x_w}$$

S_t and S_w refer to the area of the horizontal stabilizer and wing respectively, and x_t is the distance of the aerodynamic center of the stabilizer to the center of mass while x_w is the wing's aerodynamic center's distance from this point. I simplified x_w by making it the quarter chord length of the wing for all calculations.

In order to see how the horizontal stabilizer affected the stability derivatives that I chose to evaluate, I made adjustments to the tail in two ways. Firstly, I changed the span of the stabilizer, and secondly, I pushed the tail farther back on the airframe. This led to the two graphs below.

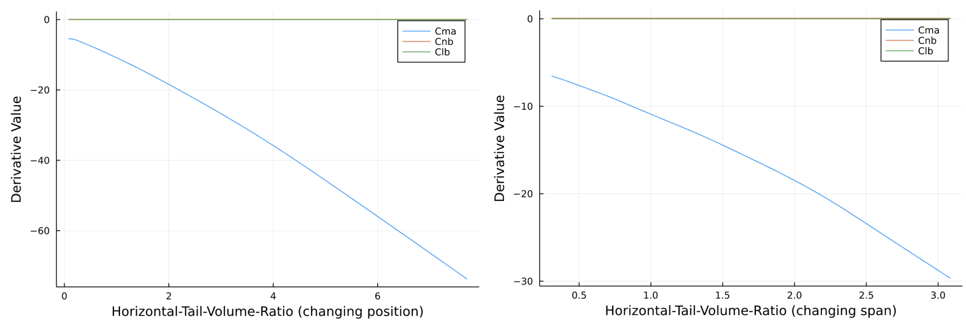

Figure 2: The image on the left shows the derivative values with respect to changing the horizontal tail position, and the image on the right shows the derivatives when the span of the stabilizer changes.

Figure 2 shows that changing the horizontal tail volume ratio really only plays a big part in changing the C_{ma} derivative. This is very useful information, though, because it can be seen that moving the tail or increasing its span makes the phugoid and short oscillation modes more stable by making the C_{ma} derivative more negative. As for C_{nb} and C_{lb}, they are not changed significantly by the horizontal tail volume ratio, so they must be affected by other parameter shifts.

It makes sense that changing parameters on the horizontal stabilizer would have an effect on a derivative that plays a part in pitch control since this stabilizer's main job is to correct the moment that is naturally created by the wing. A larger stabilizer means a greater counter moment, and a farther back stabilizer leads to a larger moment arm for the counter moment. These both will lead to a more stable plane

4.2.2 Vertical Tail Volume Ratio's Effect on Derivatives

Similar to horizontal tail volume ratio, vertical tail volume ratio is calculated in a similar way except that instead of the horizontal stabilizer's area and position, it is the the vertical stabilizer's. I did a similar analysis approach to see how the vertical tail volume ratio affected my chosen derivatives by changing its span and position. The effects can be seen below.

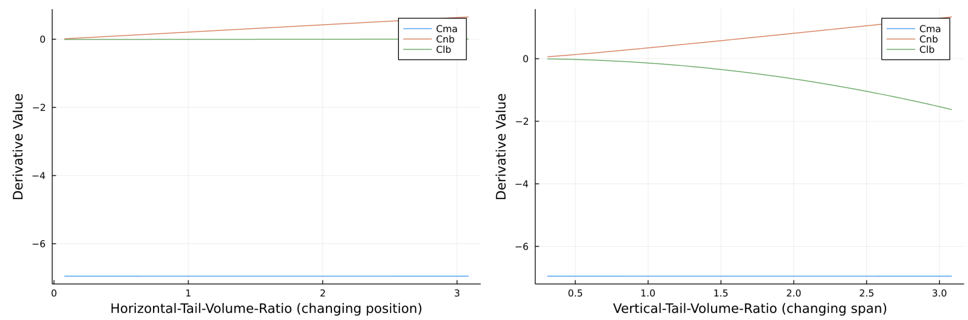

Figure 3: The image on the left shows the derivative values with respect to changing the vertical tail position (pushing it farther back on the airframe), and the image on the right shows the derivatives when the span of the stabilizer changes.

Figure 3 shows that changing the position of the stabilizer makes C_{nb} slightly more positive, which is desirable for this value. However, it doesn't change C_{ma} or C_{lb} significantly. In terms of changing the span, though, both the C_{lb} and C_{nb} derivatives change in a desirable manner significantly while C_{ma} still didn't changed.

Just like how increasing the span of the horizontal stabilizer led to a more stable corresponding derivative, increasing the span of the vertical stabilizer did the same. A larger area leads to a larger moment in the yaw axis, which leads to more stability when rotation in this axis arises. Additionally, this larger moment could also play a part in stabilizing an aircraft when it starts to roll, which is why C_{lb} also got more stable. Changing the position of the vertical stabilizer didn't have as strong of an effect on stability, but it was still effective in making C_{nb} more positive nonetheless.

4.2.3 Dihedral's Effect on Roll Stability

The major thing that dihedral helps with is roll stability in a dutch roll. In order to show that increasing dihedral would help with this mode of flight, I graphed dihedral vs. the C_{lb} derivative to see if it made it more negative.

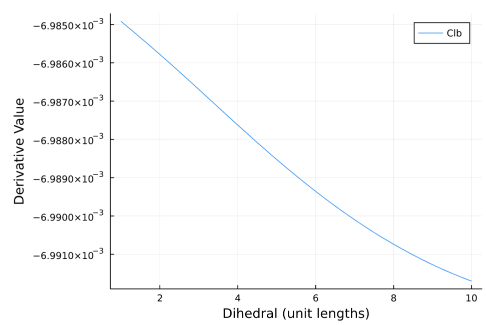

Figure 4: Increasing dihedral vs. C_{lb}. The dihedral value has units that are arbitrary with respect to the rest of the aircraft.

Figure 4 shows that dihedral definitely helps with dutch roll stability. Even though these values are small, the change is still relatively significant since the C_{lb} derivative is usually only between 0 and -0.1 as Doctor Ning stated in his Flight Vehicle Design textbook. Additionally, the most important aspect is whether or not this derivative is negative, and this graph shows that a larger dihedral helps make it more and more negative.

4.2.4 Sweep's Effect on the Stability Derivatives

Since sweep was one of the variables that I was adjusting, I thought that it would be valuable to see how a changing sweep in the wing affects the stability derivatives. In the following figure, the sweep value has an arbitrary unit with respect to the rest of the plane's geometry.

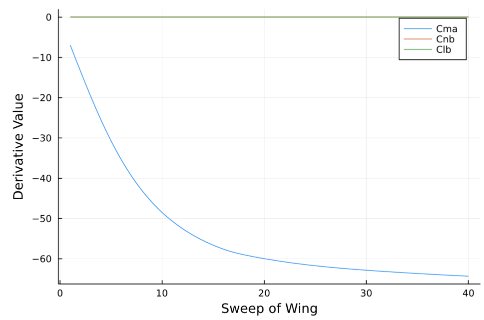

Figure 5: Sweep Value vs. Stability Derivatives

As figure 5 shows, sweep values for the wing only really affect the C_{ma} derivative while it leaves C_{lb} and the C_{nb} unchanged. The reason that sweep makes C_{ma} more stable is due to the fact that sweep actually lowers the maximum possible lift produced by the wing. This is caused because only the component of wind that is perpendicular to the leading edge of the wing generates lift, so if the sweep is at a higher angle, there will be less apparent wind that is perpendicular.

If there is less lift produced, then the moment produced by the wing will also be decreased. This means that the aircraft's horizontal stabilizer won't have to work as hard in order to produce a counter moment to keep stability with pitch. This is why C_{ma} becomes more stable with higher sweep.

4.2.5 Additional Note for why Xfoil was Helpful

Xfoil.jl has not been mentioned very much through this results section. The main reason for this is because in terms of airframe optimization, Xfoil.jl isn't as particularly helpful as VortexLattice.jl. Xfoil.jl focuses primarily on the airfoil, which was not an important part of this optimization process. However, Xfoil.jl did help me with the design variable selection process.

As was stated in the methods section, VortexLattice.jl uses an inviscid analysis method while Xfoi.jll takes viscosity into account. This means that where fluid separation and stall will be apparent in an Xfoil.jl analysis, it won't be accounted for in VortexLattice.jl. What Xfoil.jl can show can be seen in the following figure.

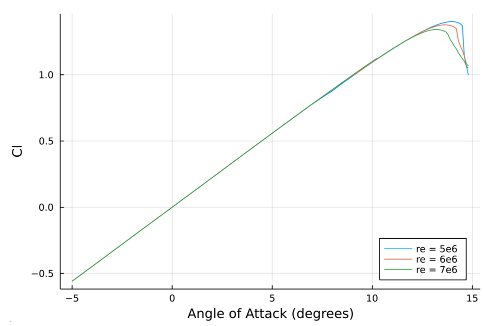

Figure 6: Coefficient of lift with changing Reynold's values and changing angle of attack from Xfoil.jl

What figure 6 shows is that there is some max angle of attack before the coefficient of lift actually begins to decrease. This is called stall and is an effect of fluid viscosity. This can be compared to what a lift to angle of attack graph from VortexLattice.jl looks like.

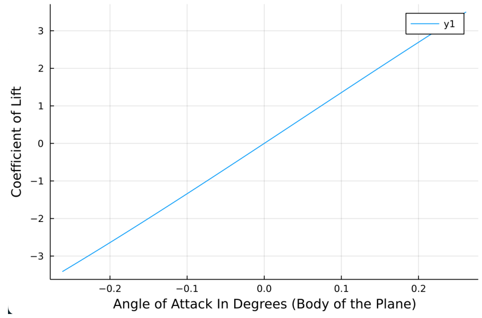

Figure 7: Coefficient of lift vs. angle of attack from VortexLattice.jl

Since VortexLattice.jl doesn't care about fluid viscosity, the lift polar is just a linear line with slope around 2π that would continue to increase indefinitely. The optimizer would just want to push angle of attack to the max. This is obviously unrealistic since at some angle of attack, stall will occur. This is why I didn't have angle of attack as a design variable and why Xfoil.jl was still useful for this analysis.

4.3 Airframe Design

At the beginning of the optimization process, I chose an initial design of an airframe that I would use as the basis for my analysis. In this section, I will show the change from the original to the optimized design, and analyze the difference in effectiveness between the two.

4.3.1 Initial Design

Wing	Value	Horizontal Stabilizer	Value	Vertical Stabilizer	Value
xle	0.1	xle	0.2	xle	0.1
yle	0.5	yle	0.3	yle	0.0
zle	0.01	zle	0.001	zle	0.1
chord	[0.3, 0.2]	chord	[0.2, 0.2]	chord	[0.2, 0.1]
theta	0.0	theta	0.0	theta	0.0
phi	0.0	phi	0.0	phi	0.0
position	0.0	position	2.0	position	2.0

4.3.2 Final Design

Wing	Value	Horizontal Stabilizer	Value	Vertical Stabilizer	Value
xle	0.5	xle	0.5	xle	0.3
yle	0.75	yle	0.3	yle	0.0
zle	0.2	zle	0.0	zle	0.3
chord	[0.3, 0.2]	chord	[0.2, 0.2]	chord	[0.2, 0.1]
theta	0.0	theta	0.0	theta	0.0
phi	0.0	phi	0.0	phi	0.0
position	0.0	position	1.7	position	1.7

Each of the data in the table refers to half of the wing and horizontal stabilizer. The other half of these components are just mirrored when analyzed by VortexLattice.jl. Additionally, xle refers to a length going parallel with the aircraft's body; yle refers to a length going perpendicular and away from the airframe (like the wing span); and zle refers to a length going perpendicular and away from the yle's axis (like the span of vertical stabilizer). The chord vectors show the length of the root chord and the tip chord respectively. Theta is the twist of each aerodynamic component, and phi is the twist of the airfoils. Additionally, the position of each of these airframe components is with respect to the center of mass of the aircraft.

4.3.3 Changes In Design

There were some obvious changes from the initial wing design to the final. The sweep increased from 0.1 to 0.5; the half span increased from 0.5 to 0.75; and the dihedral went from 0.01 to 0.2. For the horizontal stabilizer, the only two things that really changed were the sweep, which went from 0.2 to 0.5, and the position, which went from 2.0 to 1.7. Finally, the vertical stabilizer changed in its position from 2.0 to 1.7; the sweep went from 0.1 to 0.3; and the span changed from 0.1 to 0.3.

Overall, there were pretty significant adjustments made through the optimization process, but there is a chance that some of these values could have changed even more if I adjusted the constraint domain on my variables.

4.3.4 Comparison Between Initial and Final Design

Final Design	Values	Initial Design	Values
Lift to Drag Ratio	124.249	Lift to Drag Ratio	83.06
Lift Force	17.484 N	Lift Force	11.5783 N
C_{ma}	-0.0384	C_{ma}	1.4
C_{nb}	0.282	C_{nb}	0.117
C_{lb}	-0.0199	C_{lb}	0.0007

These values were calculated with an aircraft body angle of attack of 2^o and freestream velocity of 22 $\frac{m}{s}$.

Basically all of the changes in value from the initial design to the final design were expected as a product of the change in the design variables. The lift to drag ratio increased since the span of the wing increased, leading to more lift. Related to the previous statement, the lift force also increased since the coefficient of lift and wing area increased due to the increase in span. The C_{ma} derivative got more stable since wing sweep increased. I would have expected the tail to push back more in order to make this derivative better, but a farther tail must lead to less efficiency for the aircraft, which the optimizer had to work around. C_{nb} also got more stable as a result of an increase in the span of the vertical stabilizer, leading to a higher vertical tail volume ratio. And finally, C_{lb} improved due to an increase in the wing dihedral and an increase in the vertical tail span.

5 Conclusion

Through analysis of how changing isolated components of an airframe affect the efficiency and stability of a plane, I was able to build up enough knowledge of what parameters will lead to a successful plane without solely relying upon the use of an optimizer to guide my thinking. Having this foundation is crucial since the optimizer is just a tool to direct an analysis, but without a deep understanding of what is really going on, the information that an optimizer outputs can't really be validated.

Due to the foundational evidence that I built up, the changes from my initial design to my final design were not surprising. Additionally, the changes that were made led to an RC plane that had significant improvements from the original with respect to the stability derivatives and efficiency.