NACA 2412 Data and Discussion

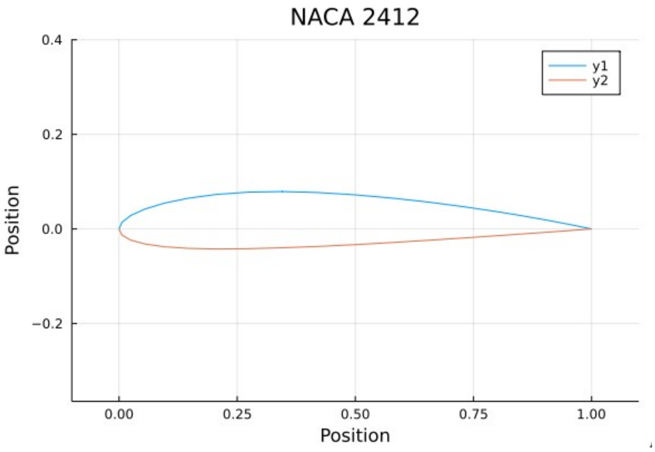

Figure 1: A Plot of the NACA 2412 Airfoil. Generated using Cosine spacing and 20 points on both the top and bottom (for the source code see Here)

NACA 2412 Polar Plots Under Various Conditions

Note: As smoothing a polar plot requires extensive user input and trial and error, only the first plot/case will be extended and smoothed, the rest will simply be given as is.

M = 0, Re = 10,000, Ncrit = 7

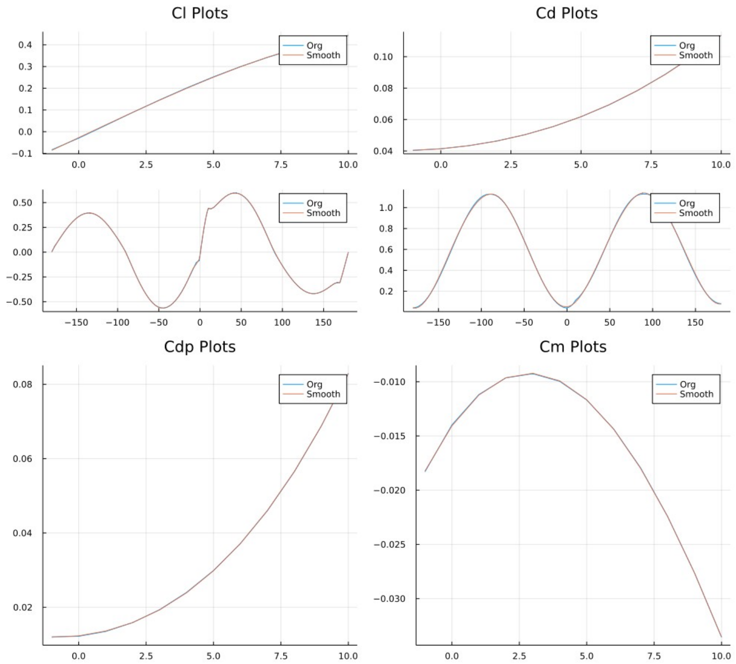

Figure 2: Plot under the given conditions (M = 0, Re = 10,000, Ncrit = 7) at different angles of attack given in degrees (Originally -1 to 10 deg before extending). All the plots were smoothed, and the Cm and Cl plots were extended and smoothed from -pi to pi. For methods, see the source code linked in the appendix.

M = 0, Re = 100000, Ncrit = 7

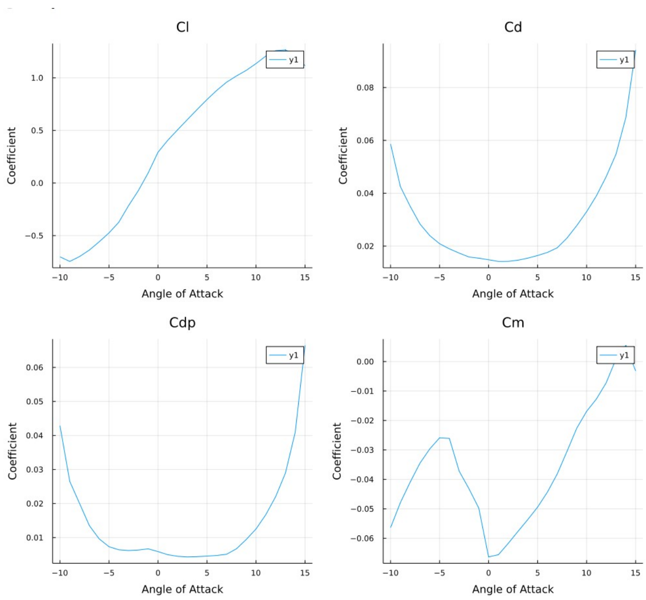

Figure 3: Plot under the given conditions (M = 0, Re = 100000, Ncrit = 7) at different angles of attack given in degrees (from -10 to 15 deg)

M = 0, Re = 1000000, Ncrit = 7

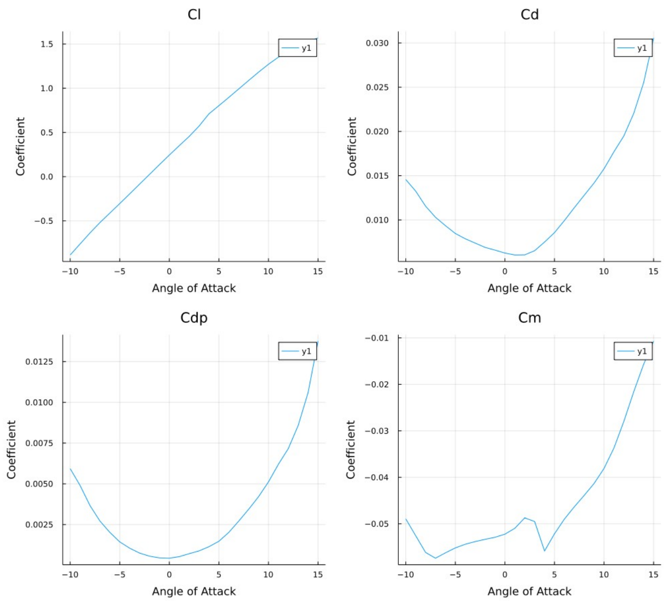

Figure 4: Plot under the given conditions (M = 0, Re = 1000000, Ncrit = 7) at different angles of attack given in degrees (from -10 to 15 deg)

M = 0.02, Re = 10000, Ncrit = 11

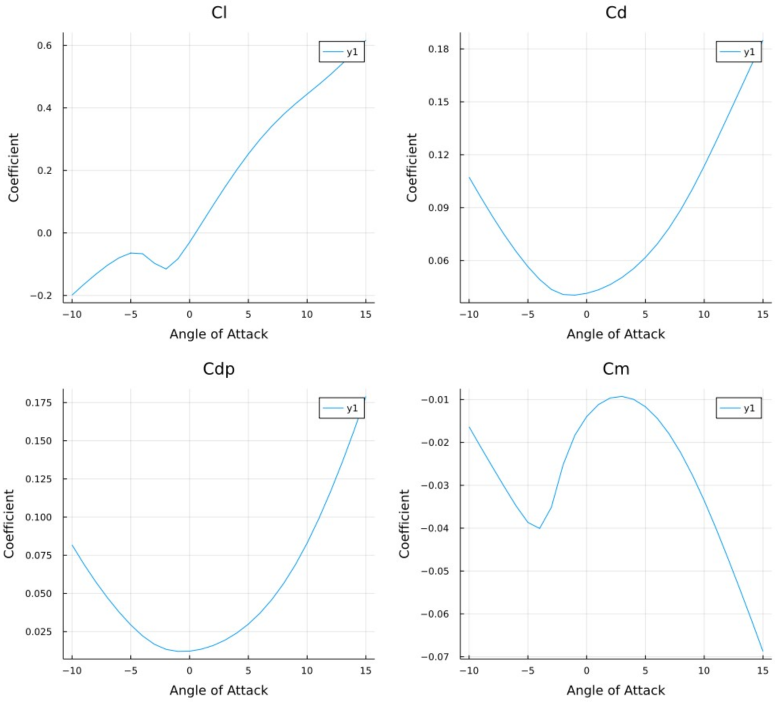

Figure 5: Plot under the given conditions (M = 0.02, Re = 10000, Ncrit = 11) at different angles of attack given in degrees (from -10 to 15 deg)

M = 0.02, Re = 100000, Ncrit = 11

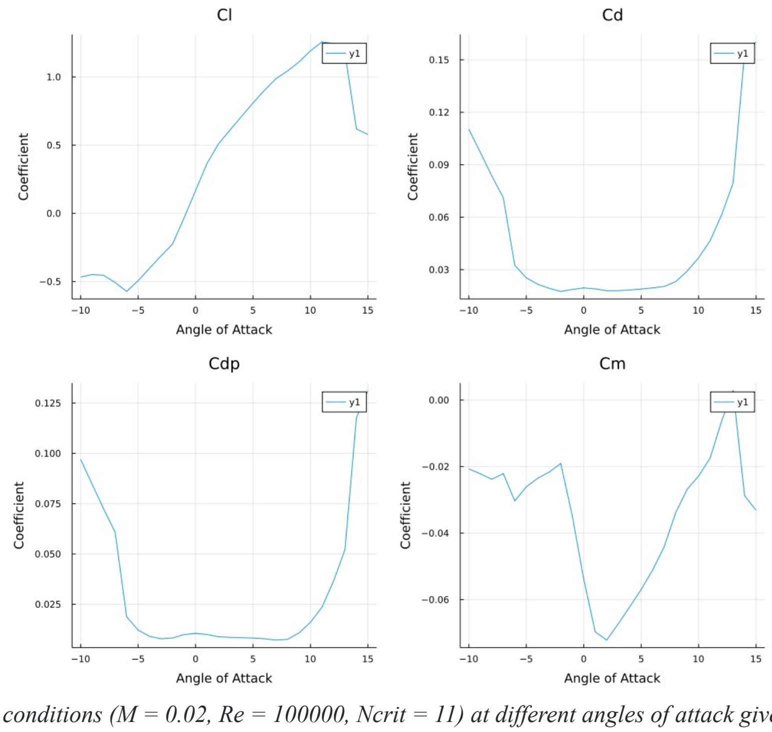

Figure 6: Plot under the given conditions (M = 0.02, Re = 100000, Ncrit = 11) at different angles of attack given in degrees (from -10 to 15 deg). It should be noted that My code failed under these conditions when not using the same coordinates as Xfoil, ie, it was necessary to have more coordinate points than the default settings in PolarPlotter.jl to have converged results.

M = 0.02, Re = 1000000, Ncrit = 11

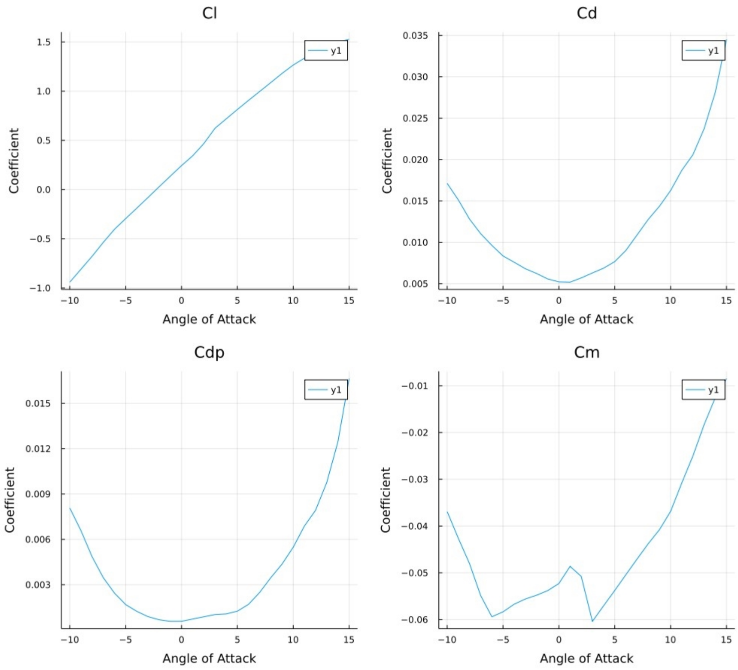

Figure 7: Plot under the given conditions (M = 0.02, Re = 1000000, Ncrit = 11) at different angles of attack given in degrees (from -10 to 15 deg)

Validation and Verification through Comparison of Results

This was done between Xfoil, my adapted Xfoil.jl code, Airfoiltools.com, and experimental data.

Validation with known theoretical data

- Example From Airfoiltools.com
  - M = 0, Ncrit = 9, Re = 50,000 , iter = 500 (their iteration limit is unknown)
  - Using downloaded coordinate file from airfoiltools.com- NACA2412.dat
  - *Note: Xfoil.jl uses 140 panels by default, Xfoil uses 160, when I updated the panels it matched*

Table 1: Coefficient of Lift Comparison between Xfoil, My code, and Airfoiltools.com at various angles of attack

Angle	Xfoil	MyCode	Online
0	-.0589	-.05884	-.0589
1	.1375	.13749	.1375
10	1.1591	1.15908	1.1591

Error Analysis

```
mycode = m
xfoil = x
m = -.05884
x = -.0589
maxtrue = 1.1591

RelativeError = (m-x)/x*100 => -0.10186757%
MaxRelativeError = (m-x)/maxtrue*100 => 0.00517643%
```

Discussion

Using the same coordinate file, and changing the number of panes in my adapted xfoil.jl code led to very little error. Double checking the Xfoil results with online validated that the test was being run with all of the correct inputs. While there is very little error, it is important to discuss the source of what error there is. My code does not use the exact same solver that xfoil does. After discussion with Judd Mehr (*is that who it was?*) it was learned that Xfoil.jl has additional methods built in to help the solver converge to a solution. Further digging that my code converged on all given output points, however Xfoil did not. Thus the source of the "error" was revealed and can be safely assumed to be within allowed tolerances with a relative error of less than a percent.

Verification through comparison with experimental data

The Experimental data and plots that were used can be found here

- As the Experimental data was not tabulated, but was instead plotted, the estimated stall angle and Coefficient of lift at that angle will be reported and compared

Table 2: Comparison of the estimated stall angles and coefficients of lift under these conditions: Re = 3,100,000, M = 0, and Ncrit = 9. (Note a mach number was not specified, so M = 0 was assumed and used)

	XFoil	MyCode	Experimental
Estimated Stall Angle (deg)	18	18	15
Coefficient of Lift at estimated stall angle	1.7689	1.7733	1.6

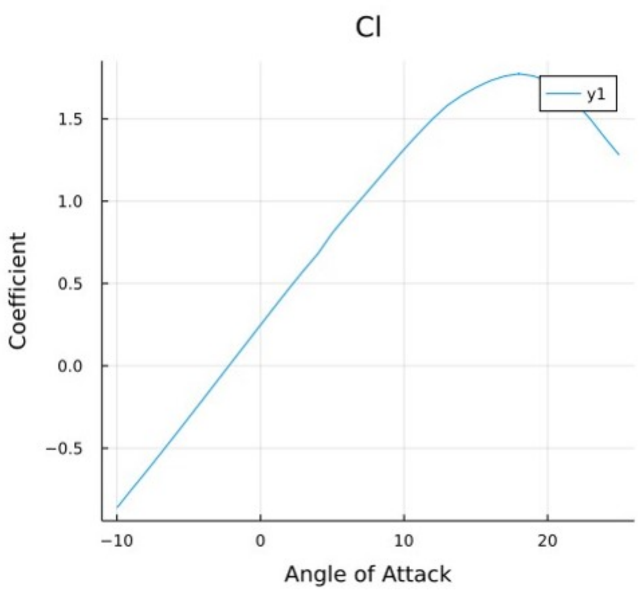

Figure 8: Theoretical output calculated from the PolarPlotter.jl code

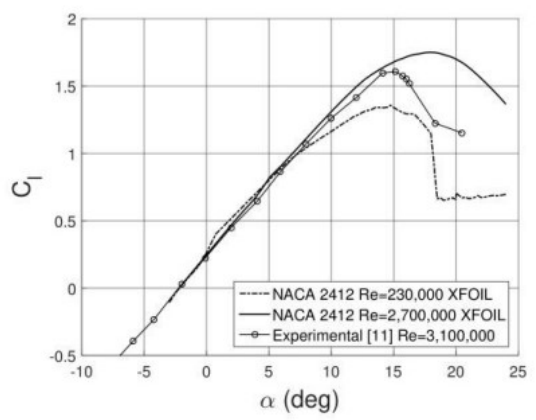

Figure 9: Experimental output found in a wind tunnel and cited in the link above

Error Analysis

```
mycode = m
experimental =  e

%Stall angle error
RelativeStallAngleError = (m - e) / e * 100
= 20.0%

%Cl error
RelativeClError = (m - e) / e * 100
= 10.83%
```

Discussion

The error between the theoretical and experimental data is quite large. It must be remembered however, that the Experimental data points were estimated off of a plot, it was assumed that M = 0, and there were unknown assumptions made by the original researchers. It follows that the aggregation of these error sources would lead to a quite large error overall. The general trends that we see with the theoretical data match what we would expect to see- the theoretical model performs better (not stalling until later), than the real life model, and the linear region and shape of the curve also generally match between cases.

Overall Discussion

Reynolds Number Trends

The Mach number and Ncrit values were left constant as the Reynolds number was varied under two different sets of conditions. A high Reynolds number means that the effects of inertia are greater then the effects of the velocity of the fluid. In simple terms this means the flow / boundary layer will become turbulent quicker. In the case of airfoils, turbulent boundary layers keep the flow attached to the airfoil, thus delaying flow separation longer at higher Reynolds numbers. When comparing the Coefficients of lift seen under the first set of conditions (for example), this effect can be seen. Tor an angle of attack of 10 degrees, Cl at Re = 10,000 is apx .4 and at Re = 100000 it is apx 1.2, and at Re = 1000000 it is apx 1.5. This shows that higher Reynolds numbers allow for the generation of more lift as the flow stays attached to the airfoil longer.

These effects are even more apparent when looking at the Coefficient of Drag. A large portion of drag comes from "pressure drag", or the drag that comes because of the pressure differential between the front and trailing edge of the airfoil. This pressure differential is caused by flow separation, so it follows that if there is less flow separation there is less drag. These results are seen when comparing the Cd plots under the second condition at a 10 degree angle of attack (for example). At Re = 10000, Cd is approximately 1.1, at Re = 100000 it is approximately .04, and at Re = 1000000 it is approximately .016. Thus, it can be seen that drag is drastically reduced with an increase of the Reynolds number.

Angle of Attack Trends

As the angle of attack was varied from negative (nose down) to positive (nose up), regardless of the Reynolds number and other conditions, a few general trends could be seen. The Coefficient of lift generally grows more positive as the angle of attack is increased until it eventually reaches stall (where the airfoil stops producing as much lift). It should be noted that at some negative angles of attack the Coefficient of lift briefly stops growing more positive (at this point Cl is already negative) but has a negative slope for a few degrees before reversing again and becoming more positive as seen in Figure 10. This is due to the streamline/flow being largely unattached from the airfoil, and then reattaching as the negative angle of attack is less extreme and briefly generating more negative lift.

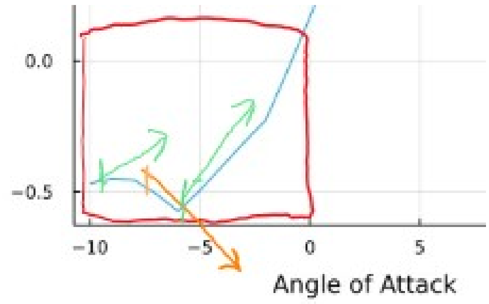

Figure 10: Coefficient of lift with the slopes outlined at negative angles of attack. Taken from the M = 0.02, Re = 1000000, Ncrit = 11 Condition

The Coefficient of Drag displays more intuitive behavior in that as the angle of attack travels from negative to positive angle at which the least drag occurs is typically within a few degrees of 0. This makes sense as that is when the airfoil would be the most streamlined and the flow would be the least separated, thus generating the least friction and pressure drag respectively. Keep in mind that as was already shown above the effects of pressure drag are much larger than friction drag, and thus the angles of attack at which flow separation is larger are where the Coefficient of Drag is also much larger.

The Coefficient of Moment in all cases stays negative within our range of angles of attack. A negative Coefficient of Moment means that the airfoil is wanting to pitch nose down *verify this*. This is desirable behavior as it means that the airfoil will resist stall (the angle at which an airfoil no longer produces more lift). The moment Coefficient is closely related to the other coefficients. In general both Lift and Drag give a positive moment to the airfoil (ie cause the nose to pitch up), *so where does the negative coefficient of moment come from?* This can be explained by the pressure distribution along the airfoil. At a higher angle of attack, both drag and lift will be trying to rotate the airfoil nose up, but a closer look at the pressure distribution shows that the locations and magnitudes of the pressure actually cause a negative moment coefficient. At higher Reynolds numbers as the angle of attack increases, the coefficient of moment generally becomes less negative, and at lower Reynolds numbers as the angle of attack increases, the coefficient of moment generally becomes more negative.

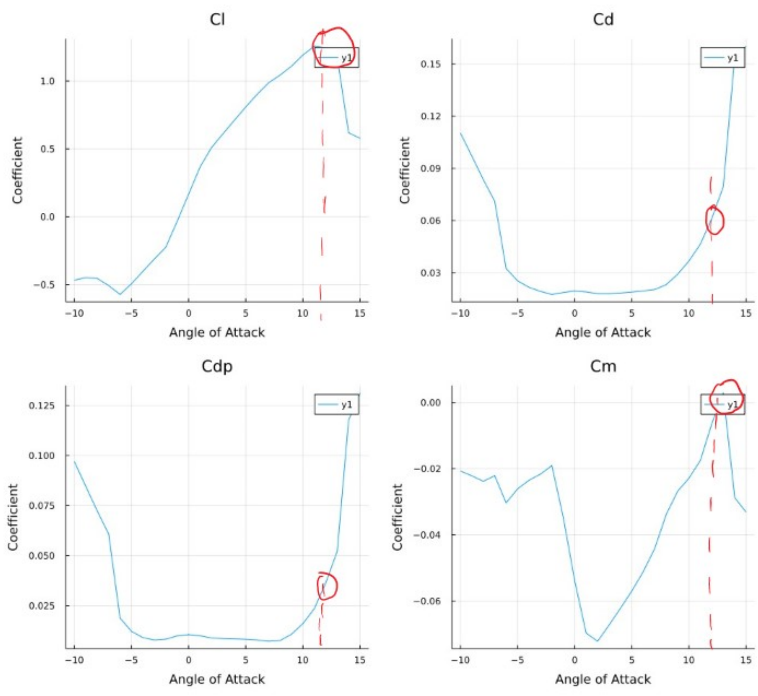

Figure 11: Example of the location of positive stall as outlined in the different plots. Taken from the M = 0.02, Re = 1000000, Ncrit = 11 Condition

Stall Trends

Stall occurs when at an angle of attack when the airfoil generates less and less lift rather than more than before. This typically happens at higher angles off attack, with relatively high (or rapidly increasing) coefficient of drag, and a very small moment coefficient (ie not a strong desire to pitch the nose down). These trends can be seen in Figure 11. Stall can also occur in the negative Coefficient of Lift region. This can happen when the airfoil is beginning to generate more positive lift (ie still negative lift but with a positive slope) and then suddenly becoming more negative again as previously shown in Figure 10. Negative stall also occurs when Cl is negative, but typically comes from the airfoil being inverted (extremely high or low angles of attack), and then just rotating out of its inversion as shown in Figure 12.

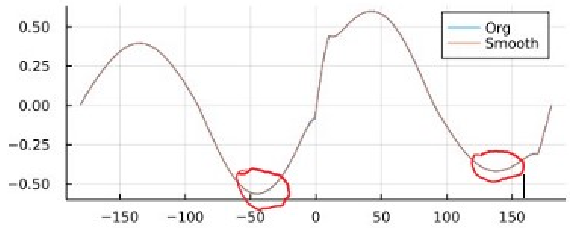

Figure 12: Negative stall example as shown on the extended plot. Taken from the M = 0.0, Re = 10000, Ncrit = 7 Condition

Conclusion

A much deeper understanding of the Coefficients of Lift, Drag, and Moment were gained, as well as a more intuitive grasp of how they vary with angle of attack. A proficiency at basic airfoil analysis, plotting, and comparison was gained, but with XFoil, and Xfoil.jl. Julia coding skills were improved with an increased knowledge of how to use functions in different files and how to use different math packages. The code was both validated and verified and will be safe to use (within normal conditions) in the future.

Appendix

A. High Level Coding Approach

- Figure 1 was generated through the AirFoilCoordinatesFunction.jl file, with the lines to make a function commented out and the lines to plot uncommented. The same output can come from the AirFoilCoordinatesArraysActivity.jl file with no changes as that was made to stand alone. Both plots are validated.

- Figure 2 was generated through the ExtenderNSmoother.jl file, which called on the PolarPlotter.jl function to generate the coefficient values to be extended and smoothed. PolarPlotter.jl called on the AirFoilCoordinatesFunction.jl function to generate the airfoil coordinates to be fed to the xfoil solver in PolarPlotter. The resulting data from PolarPlotter.jl was smoothed by creating an overlayed 1-D spline and evaluating it. That was made possible through the Dierckx package. The smoothed Cl and Cd data was then extended through the Viterna function from -pi to pi, made possible by the CCBlade package. The extended data was then smoothed and plotted.

- Figures 3-7 (ie non extended and smoothed plots) were generated through the TempPolarPlotter.jl file which was virtually identical to PolarPlotter.jl, but is not a function and was made to output the needed plots and data slightly differently than ExtenderNSmoother.jl did.

B. Source Code

ExtenderNSmoother.jl

PolarPlotter.jl

AirFoilCoordinatesFunction.jl

TempPolarPlotter.jl

For all other code, output images, and this document (in Markdown) see the XfoilProject folder on my GitHub.

Note: All of these links currently take the user to a branch off of main. Once everything in the project has been validated and verified, a pull request will be created and this branch will be merged with Main. It is yet to be seen if the links will work once that happens.