The Use of Global Variables and Insert Part Feature to Create Adaptive Models in Solidworks

Martin Herrera mh866@cornell.edu
Cornell DBF Fall 2016

Abstract

Through the use of interdependent parts and variables in dimensions, it is possible to create a system of parts that adapt according to a set of easily accessible and modifiable numerical parameters. The scope of this document covers the use of global variables and the insert part feature in Solidworks to create adaptive parts. It is assumed that the reader is familiar with CAD design and has had exposure to Solidworks.

2016 Competition Overview

The 2016 DBF Competition centered around the design of a hand launched aircraft that could be stored in one piece within a tube for transport. The competition also required that teams design the storage tube to be used. The dimensions of the tube were factors in the rated aircraft cost of the design.

An optimized score could be achieved by minimizing the dimensions of the tube as much as possible while maintaining a diameter to length ratio of 4.Teams were permitted to use hinges and other such mechanisms to fold the plane such that it could fit in the smallest tube size possible.

Introduction

One of the designs pursued to accomplish the task of fitting the plane inside of a tube was a foldable carbon fiber wing. The intention was that the wing would wrap itself around the fuselage. The wing's cross section was the camberline of an Eppler 379 airfoil and the wing itself was a single surface. The camberline cross section would give the wing the rigidity needed to keep it's shape while providing lift but the flexibility needed to bend downwards and wrap around the fuselage. The inspiration for this design came from Prioria Robotic's Maveric UAV shown in figure 1.

Figure 1: Maveric UAV in flight ready and stored positions

Due to the wing being a surface with no internal structure, the resulting CAD model would not change drastically between iterations. Because of this, the use of parameterized design was employed to model the wing. Within the scope of this document, parameterized design is the use of user defined parameters or external part files to constrain a CAD model.

Dimensions and feature characteristics can be explicitly defined by global variables that can easily be accessed and modified. Additionally, external part files can be imported such that its geometry can be used to constrain dimensions and features.

Using these tools it was possible to iterate on a wing and wing- fuselage connection model simply by changing the values of a few parameters. Global variables for root chord, tip chord, span, and wing-fuselage connection width were used to explicitly define the model. All other dimensions were dependant upon these values. Therefore when any of these values was changed, the entire model shifted into the according shape without any modification on the user end. The effects of this can be seen in figures 1 and 2 which show the CAD model with two different sets of inputs.

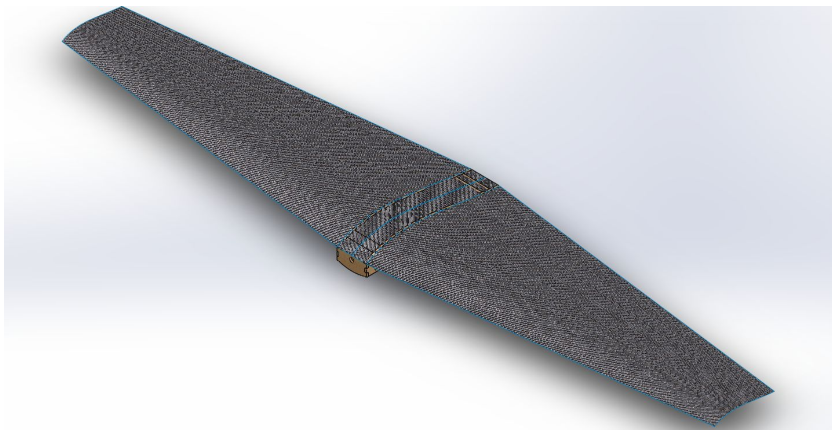

Figure 2: Adaptive Cad Example 1

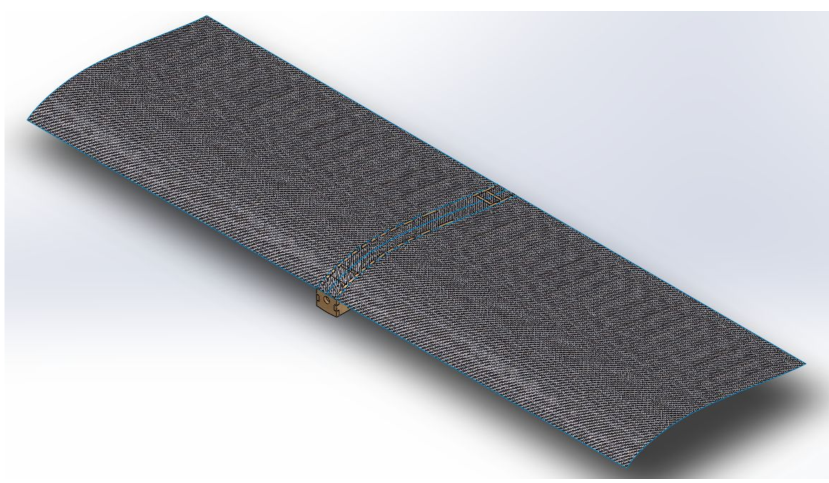

Figure 3: Adaptive CAD Example 2

Considerations

Parameterized design is useful for models in which small changes in overall geometry are expected but the general design is expected to remain static. In the camber-line wing example, one would expect that the span or taper of the wing may change but that the wing would remain a surface and that the cross section would remain the same. This would be a good opportunity to use parameterized design. Although it is possible to use these techniques on a more complex structure such as a balsa wing with ribs, the more complex the part, the more difficult it is to properly explicitly parameterize it.

It is important to consider how many variables would be needed to fully describe the model and whether any dimensions can be expressed in terms of these variables. Likewise it is important to consider whether the design is expected to change drastically between iterations. If this is the case then the time needed to produce a properly constrained and adaptive model is not justified and modeling should be done using traditional methods.

Global Variables

Global variables are useful tools for creating easily accessible user defined dimensions. Any field that requires a numerical value, such as sketch dimensions or extrusion lengths can be replaced with a global variable or an expression utilizing a global variable. By using a common global variable amongst several different dimensions the need to individually modify each one is eliminated. Changing the value of a global variable automatically updates all dependent dimensions and features accordingly.

Global variables can be accessed from the equations folder in the featuremanager design tree. In a new part, the equations folder must be created by adding a new global variable or equation. To do this, the equations window must be opened by selecting tools from the menu bar and then clicking on equations from the drop down menu. In the equations window (figure 4) one can enter the name of the desired variable and the corresponding value, clicking ok will save and close the equations window.

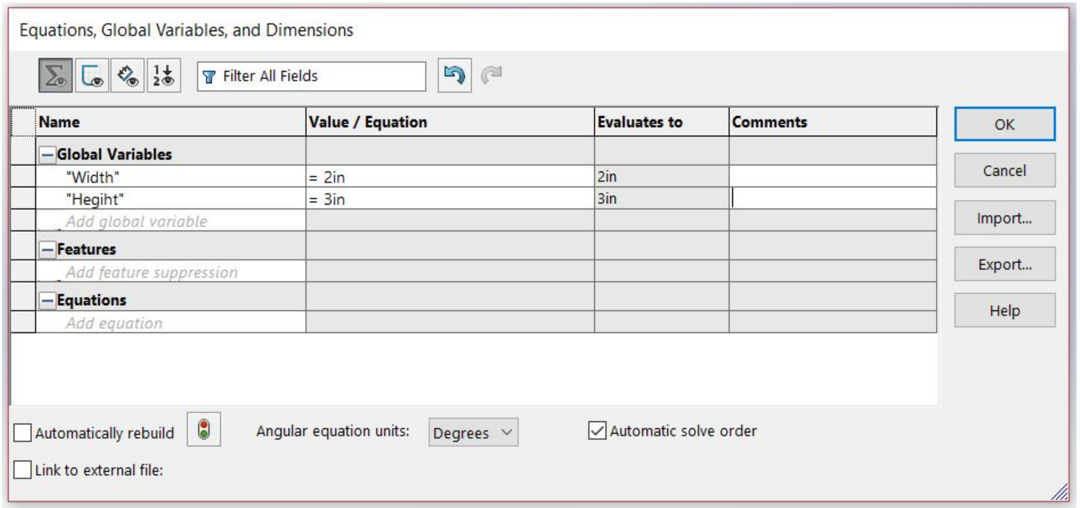

Figure 4: The Equation Manager Window

Once a global variable has been created, the Equations folder will appear in the featuremanager design tree. The folder can be expanded to show all the saved global variables and equations (figure 5). Right clicking an item in the expanded folder reveals a manage equations option that can be used to navigate to the equations window to modify, add, or delete variables.

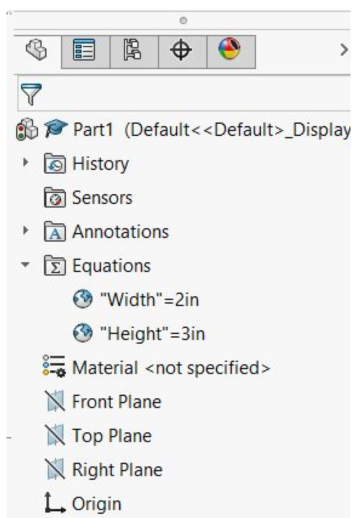

Figure 5: Expanded Equation Folder in the FeatureManager Design Tree

To reference a variable within a sketch first dimension the desired feature. In the modify window for the dimension object give the dimension a value of “= VariableName” where VariableName is the name of the desired global variable within quotation marks. Once the changes have been applied a red epsilon will be displayed next to the numerical value of the dimension. This denotes that a dimension has a global variable or equation dependency.

Referencing a variable within a feature can be done in a similar nature. However, this is only supported by features that require a numerical field such as boss extrude, extruded cut, etc. When creating or modifying the feature simply place in the desired field “= VariableName”, where VariableName is the name of the variable

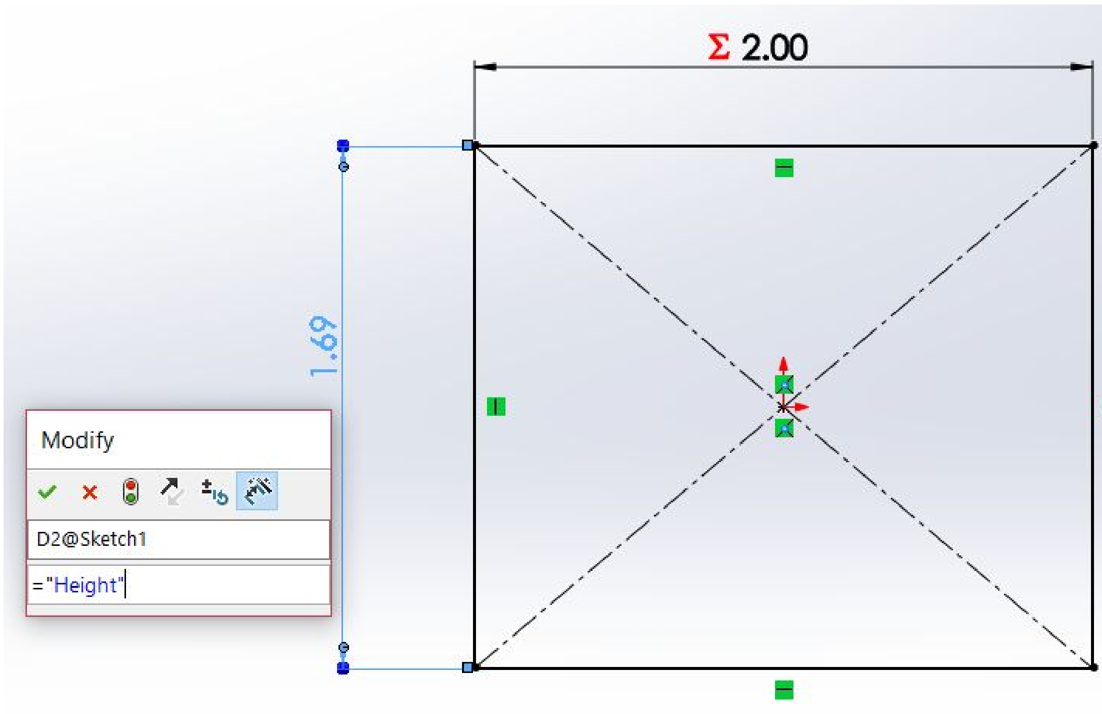

Figure 6: Dimensioning Using Global Variables

Dimensions and features can also be defined by expressions using global variables. For example, inserting “=0.5*VariableName” will give the specified dimension half the value of the variable VariableName. This is useful for creating several dimensions that are all dependant off of a singular variable but need to have different values.

It is important to note that although sketches and features will adapt to changes in variable values, it is easy to break sketches and features and create errors if one is not careful. When using variables within a sketch it is worthwhile to consider whether other dimensions and features can use the variable in order to increase adaptability. For example, if the length of a part with a slot is to be defined by a variable, it may be worthwhile to make the length of the slot also dependant off of the variable by some expression. If the feature is not constrained by some other parameter, such as having to have specific dimensions, this is a good way to maintain the adaptability of the part and minimize the effort of having to modify a drawing.

Insert Part Feature

The insert part feature is a useful tool for referencing geometry from another part file while modeling a new part. The use of this feature builds off of the concept of a multi-body part. A multi-body part is a part file containing several independent solid bodies. This method is useful for modeling sub assemblies with several parts that interface together and require the referencing of each other's geometry. This makes the resulting collection of parts much more adaptive to changes than an equivalent assembly made of several independent part files.

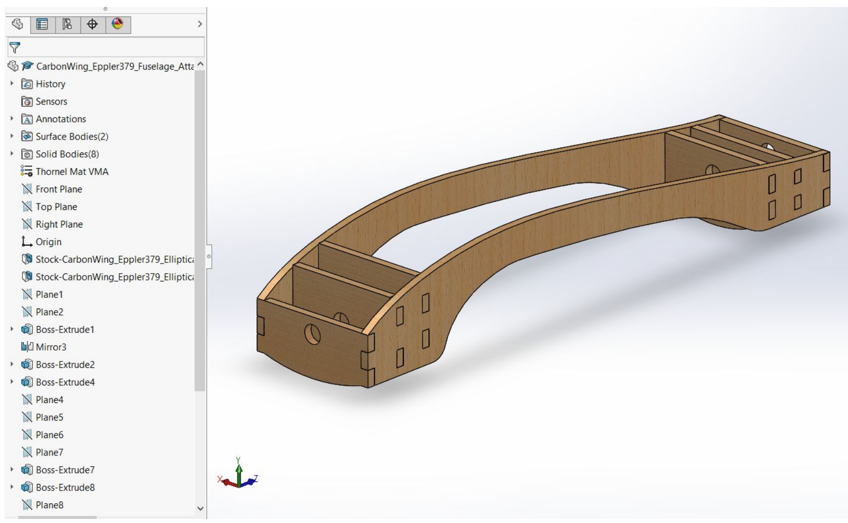

Figure 7: Example of Multi-Body Part

The insert part feature allows for parts to be dependant off of one another and increase their adaptability, much as a multi-body part would. Inserting a part into another part file creates a one way link between the parts. Any changes made to the inserted part file will be reflected in all the part files into which the part has been inserted. However, when editing within the dependent part files it is not possible to make changes to the inserted part file.

To insert a part the insert part feature must first be selected using the commands search. The commands search box can be accessed from the search box located at the top right of the window next to the help icon. The default setting for the search box is Solidworks search, this can be changed by selecting the drop down arrow next to the search box and selecting the commands option from the drop down menu. Once this is done, the insert part feature can be accessed by typing it into the search box and selecting it from the dropdown menu.

Figure 8: Commands Search for Insert Part

Once the feature has been selected, a file browser window will appear and prompt for the part file to import. Once the file has been selected and confirmed, options for selecting the degree to which the part is inserted can be found in the left panel. For a standard part it is recommended that the minimum be to import solid bodies and surface bodies. Confirming the insert parameters by pressing ok will insert the part such that the origin corresponds to the origin of the current part file.

It is recommended that if the insert part feature is to be used, it be done before any other features are made. Once the part has been inserted any of its geometry can be used to create sketches or features. If the geometry being referenced is changed within the original part the corresponding sketch or feature will be updated accordingly. Afterwards, if only the new part is to be displayed the bodies and surfaces of the inserted part can be hidden by expanding the solid bodies or solid surfaces folder in the featuremanager design tree and hiding them.

Application on Carbon Fiber Wing Design

Both global variables and the insert part feature were used to construct the CAD for the first prototype of the carbon fiber camberline airfoil. The finished CAD included the wing as well as a balsa wing to fuselage connection piece. Both parts were fully adaptive to four input parameters, wing span, root chord, tip chord, and fuselage connection width. Changing any of these global variables would change all dependant features within the CAD, creating a new iteration within a matter of seconds. The effects of this can be seen in figures 1 and 2. The creation of a fully adaptive CAD such as this allowed for a single user to make changes to the prototype in a fraction of the time. The following section outlines the methods used to create this method as an example of the possibilities of implementing parameterized design.

Firstly,the desired parameters that could be changed were defined as global variables. The airfoil curve was then imported into solidworks on a sketch plane. Instructions on how to do this can be found in appendix A. Importing the airfoil with this method creates curve objects in the feature manager. In a new sketch, on a parallel plane to the curves, the curves can be projected using the convert entities feature. On the same sketch and using the projected airfoil as a guide, the camber line can be made using the spline feature. The resulting sketch is shown in figure 9. This sketch need not be fully defined as its purpose is to be copied and scaled into another sketch.

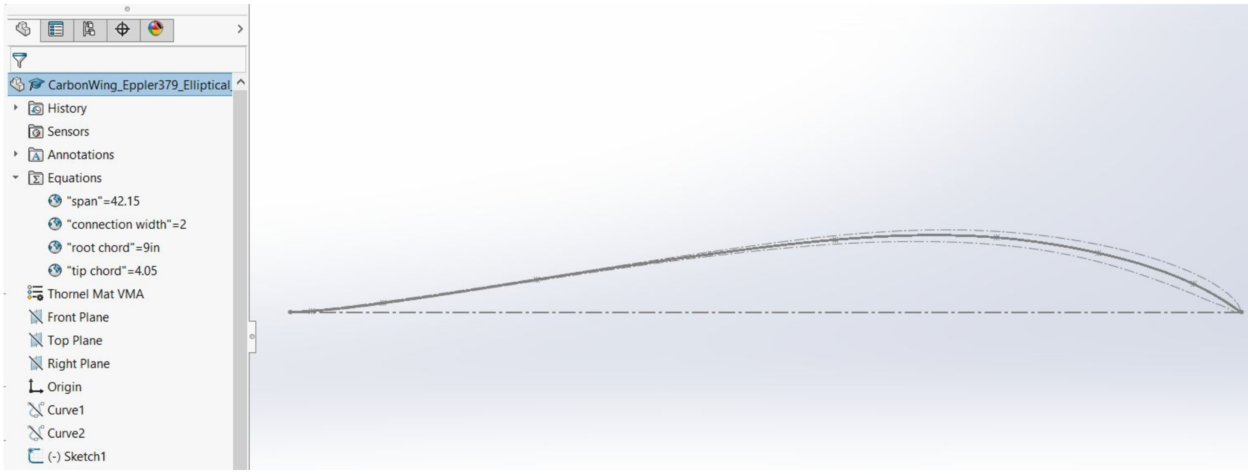

Figure 9: Reference Airfoil Sketch

In a new sketch, the camber line was projected using the convert entities feature. These projected features would then be turned into construction lines. In order to rescale the airfoil to the desired chord the scale entities feature was used on the projected lines. The scale was set at an arbitrary value as the actual scale would be constrained by dimensioning the new lines. Drawing a construction line between the leading and trailing edge allowed the chord of the scaled curves to be dimensioned. This sketch was dimensioned using the global variable for the root chord and can be shown in figure 10. A reference plane was then created offset from the sketch of the root. The offset distance was set as half of the global variable for span. On this new plane another sketch was made onto which the original camberline was projected, new curves were made by scaling the projected curves, and a construction line drawn between the leading and trailing edge, and the chord dimensioned using the tip chord global variable. Once both sketches were made a surface loft between the two sketches was used to create one half of the wing.

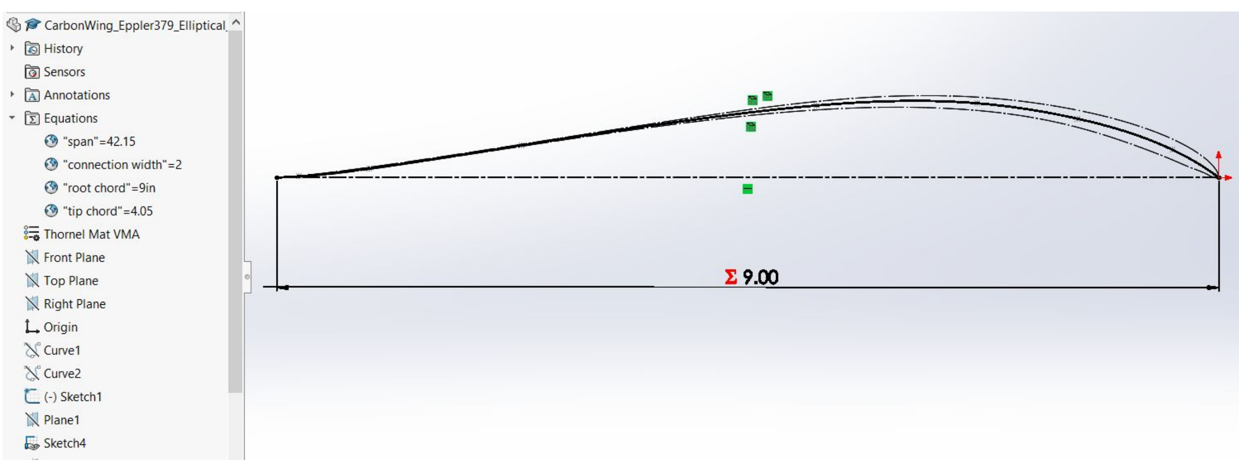

Figure 10: Root Airfoil Sketch

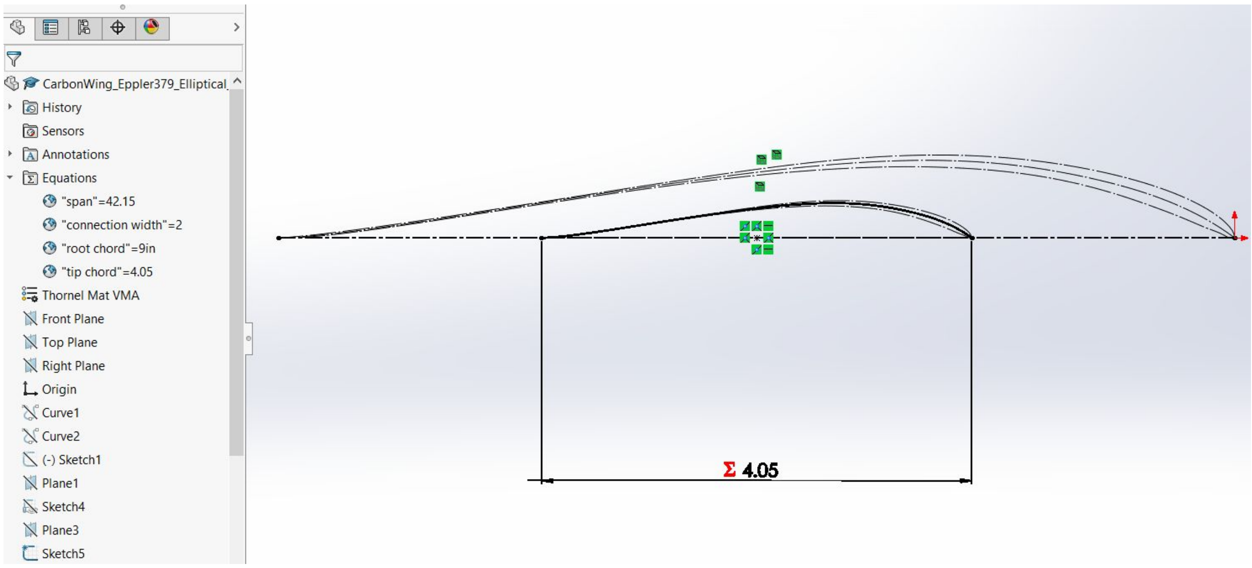

Figure 11: Tip Airfoil Sketch

The middle portion of the wing surface that would connect to the fuselage was made using another surface loft feature. A reference plane was created offset from the root by a distance defined as half of the fuselage connection width variable. This would then be the middle plane of the wing. On this plane another sketch was created onto which the the root camberline sketch used in the first surface loft was projected. Using both these sketches another surface loft was made. All surfaces were then mirrored over the middle plane to create a full wing and wing fuselage connection surface. The final wing is shown in figure 12.

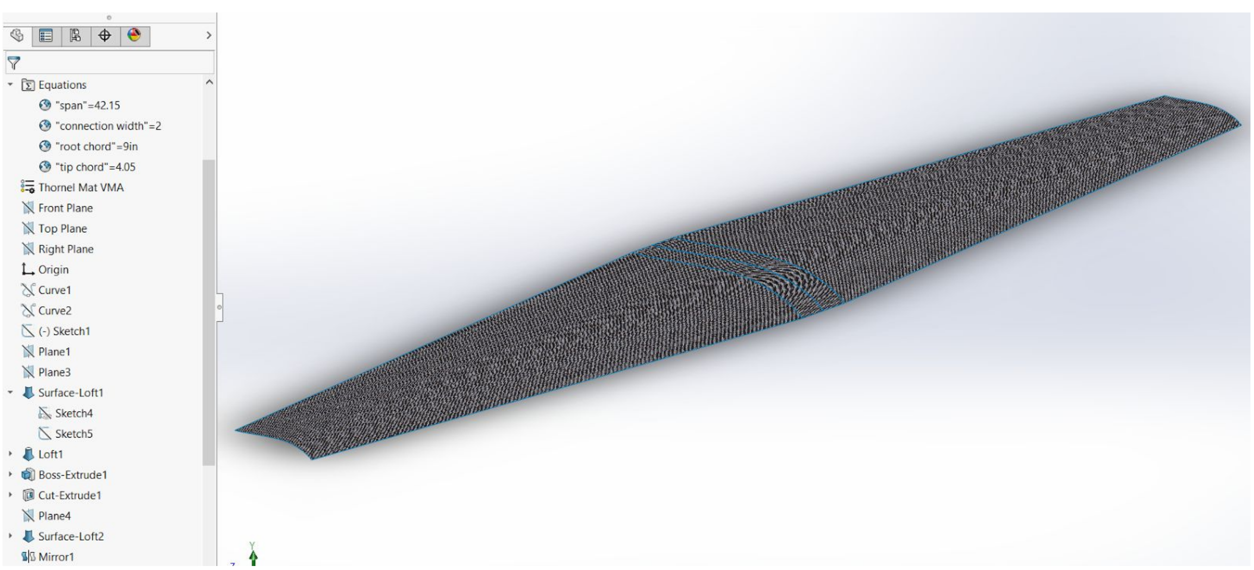

Figure 12: Mirrored Surface Lofts and Final Wing

Once this was done, the part could be saved as it's own independent file. This file would then be used to build the fuselage connection, whose width and length were dependant on the width of the middle of the wing. In a new part, the wing was imported using the insert part feature. This created several stock features, of which all but those corresponding to the middle of the wing were deleted as they were not needed. A new reference plane that was parallel to the right plane but in plane with one of the outer edges of the middle of the wing was created. A mid plane for the middle of the wing parallel to this plane was also made. The result of this can be seen in figure 13.

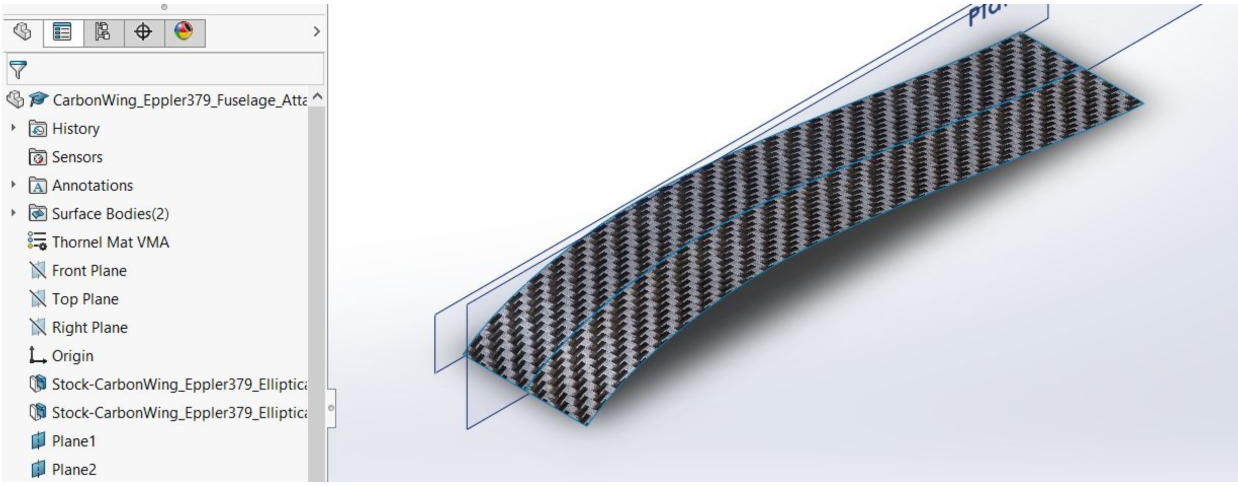

Figure 13: Inserted Part and Reference Planes

The side of the fuselage connection was then sketched on the first reference plane. This allowed for the use of the edge of the wing to be used within the sketch using the convert entities feature, shown in figure 14. Now if the root chord were to change, the length of the fuselage connection would also change accordingly. Once an acceptable side piece was sketched and extruded, the extrusion was mirrored over the middle plane. Now if the width of the middle of the wing were to change then the model would also dapat accordingly without any user input.

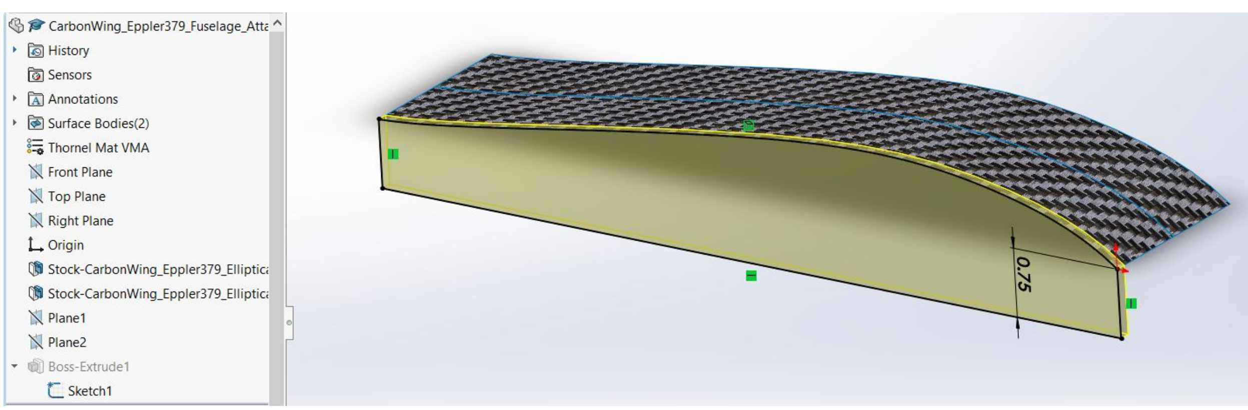

Figure 14: Use of Inserted Part Geometry to Constrain a Sketch

The rest of the fuselage connection was made using standard multi-body CADing techniques. However, because the sides of the connection housing referenced geometry from the inserted part, all other features were ultimately dependant off the inserted part. This allowed for the connection to change automatically as global variables in the wing CAD model were adjusted. All that was required to do so was press the rebuild button at the top of the tool panel. The effects of this can be seen in the examples shown in figures 15 and 16.

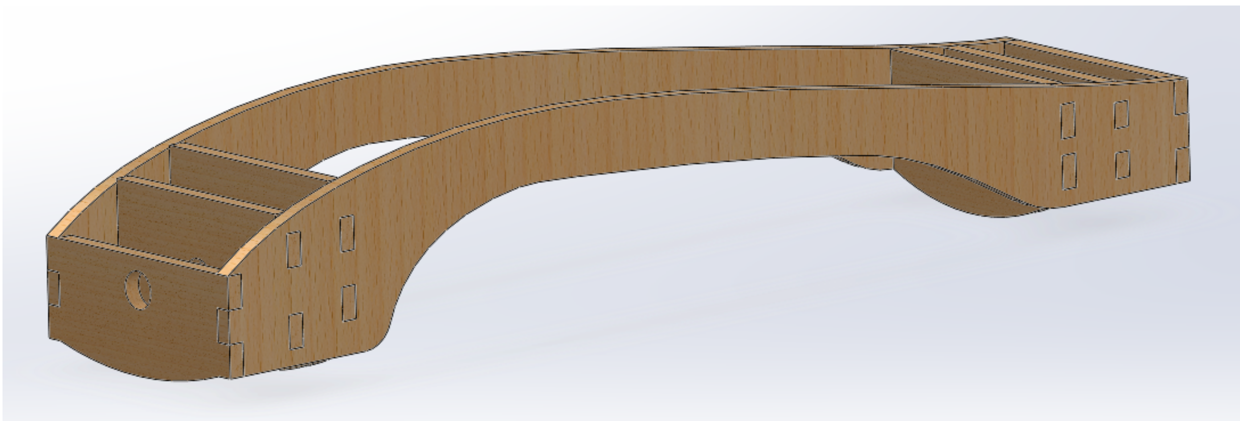

Figure 15: Wing Fuselage Connection Variant 1

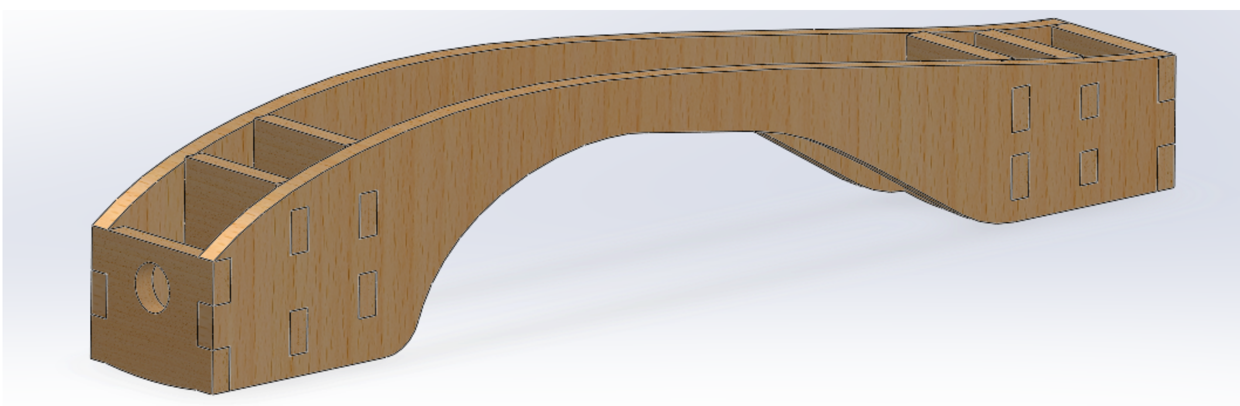

Figure 16: Wing Fuselage Connection Variant 2

Conclusion

Parameterized design can be an excellent tool for creating robust and adaptable CAD models. However, they are heavily limited by their initial design and intended scope. It is important to anticipate any possible changes that may need to be made as well as all the required dependencies on other parts before beginning to model. However, with careful consideration and under the appropriate circumstances the endeavor can prove to be efficient and immensely time saving.

Appendices

Appendix A

Making a Curve in SolidWorks From XYZ Points
(Making an Airfoil)

1. Get your data into Excel. If you're using data from World Of Krauss.
    1. Download the DAT file from WoK (World of Krauss), do this by right clicking and doing "save link as".
    2. Import this into Excel in a way that each space only contains one number. You can do this by any of the following methods:
        - Opening up the DAT file, copying it and then clicking the little arrow under the paste to get "Use Text Import Wizard". Make sure you select "Delimited" and on the next window select the "space" option so that each row is a single number.
        - Or going to the data tab and then click on the "Import Text" button. After you select your file (you may need to make it so the browser shows all files) just follow the steps above.
        - You can also paste the DAT file into notepad and then going through the other two options. Might make it easier?
2. Add zeros in one of the columns, as necessary. There needs to be exactly three columns, even if one column is all zeros. For airfoils, I recommend moving the x coordinates to the z column, then making the x column all zeros. This will insert the airfoil shape on the YZ plane, which is what you will eventually want if x is forward.
3. **THIS IS THE MOST IMPORTANT PART: Make sure there are no ADJACENT duplicate points EXCEPT THE FIRST AND LAST POINT, WHICH MUST BE THE SAME FOR A CLOSED CURVE.** World of Krauss often has a duplicate point somewhere around the origin, which is in the middle of the data. Delete it. Solidworks will draw the curve through the points in order, top to bottom. So you CAN have duplicate points, just not adjacent ones.
4. Scale your data as necessary SW will assume the coordinates are in whatever ever the base length unit is for your part file. Meaning if my data goes from 0 to 1 (like World of Krauss) and my file is in MetersKilogramsSeconds, SW will make my airfoil have a chord of 1m. Easiest way to do this is to have two new columns where the first one is =A1*[chord length] and the second one is =B1*[chord length]. You can always make a scatter plot to ensure it looks right. Note you can also scale the curve in Solidworks using tools>sketch tools>scale. You can also scale solids too under insert>features>scale.
5. Fix up your Excel sheet, making sure you only have X coordinates in the A column, Y coordinates in the B column, and Z coordinates in the C column, and nothing else anywhere on the entire sheet (no graphs, no column headers, nothing). Save this as a Comma Delimited (CSV) file. Click Yes at the warning about losing functionality or whatever.

6. Make sure file extensions are turned on (For windows you can do this by going to control panel, search for folder options, then under view uncheck the box that says “Hide extensions…”)
7. Then manually rename the .csv file as .txt.
8. In SW, go to Insert -> Curve -> Curve Through XYZ Points (or equivalently, Features -> Curves -> Curve Through XYZ Points). Browse to your file and open it (you will have to change it so that it looks for .txt files). You should see a preview of your curve in the workspace. Click OK. If you get an error, go back and check your points in Excel, looking for adjacent duplicates. This would also be a good time to double-check the units of the document, since SW defaults to MKS and most aerospace design is in IPS.
9. Note that at this point, we have a curve (which is a feature), not a sketch. Click the same plane your curve is on and create a new sketch. Click Convert Entities, then select the curve you just made and click OK. Then you can finish editing the sketch. Tada! Now you can extrude your airfoil (or whatever else you made)!