

# SUAVE: An Open-Source Environment Enabling Unconventional Vehicle Designs through Higher Fidelity

Timothy MacDonald\*, Emilio Botero\*, J. Michael Vegh\*,  
Anil Variyar\*,  
Juan J. Alonso<sup>†</sup>

*Stanford University, Stanford, CA 94305, USA*

Tarik H. Orra<sup>‡</sup>, Carlos R. Ilario da Silva<sup>§</sup>  
*EMBRAER, São José dos Campos, SP, 12277-901, Brazil.*

SUAVE is a conceptual level aircraft design environment that incorporates multiple information sources to analyze unconventional configurations. This work incorporates higher-fidelity tools to build upon previous efforts where SUAVE analyzed and optimized several types of aircraft using low-fidelity methods. This is done in an automated way that incorporates three external programs. The first is OpenVSP, which is used for geometry creation, area calculation, and surface meshing. The second is Gmsh, which uses these surface meshes to create volume meshes. The third is SU2, which is used to run Euler CFD simulations. Wetted areas from OpenVSP and lift from SU2 is used to enhance SUAVE's aerodynamic analyses. We present results for a verification case with the Onera M6 wing, then present mission results with a conventional narrow-body airliner, a supersonic jet, and a blended wing body.

## I. Introduction

SUAVE is an aircraft design tool created with the goal of providing an environment for both the analysis and the optimization of aerospace vehicles while allowing for the integration of unconventional configurations. In prior work, a description of some of the analysis capabilities available has been presented.<sup>1</sup> A follow-up work details the usage of this tool in an optimization context.<sup>2</sup> This paper investigates the integration of higher-fidelity analysis to further enable unconventional configurations.

Higher-fidelity analysis becomes incredibly important when the boundaries of aircraft design are pushed. The unconventional configurations and technologies that SUAVE was envisioned for lack the tried and true empirical correlations of most traditional aerospace design tools. This necessitates more physics-based methods to augment the current correlations. The goal is to reduce risk and uncertainty when developing these new configurations and designs as early as possible in the conceptual design stage.

Prior work using SUAVE has included using CFD and FEM analysis to analyze and optimize strut braced wing aircraft.<sup>3</sup> However, this work was done solely for a strut braced wing aircraft and could not easily be transferred to other vehicle types. In order to make these kind of capabilities available to a wide range of vehicles, and a wide variety of users, increasing generality and adaptability while maintaining simplicity are key. Other work has explored these MDO type problems with user interactions to do things such as mesh

---

\*Graduate Student, Department of Aeronautics and Astronautics, AIAA Student Member.

<sup>†</sup>Professor, Department of Aeronautics and Astronautics, AIAA Associate Fellow.

<sup>‡</sup>Aircraft Conceptual Design Engineer - Embraer, AIAA Senior Member.

<sup>§</sup>Technology Development Engineer - Embraer, AIAA Associate Member.

generation,<sup>4</sup> and collections of open-source tools have been combined to do similar jobs.<sup>5</sup> However, these all lack a unified design environment for the designer to work from.

This paper is organized as follows: Section II details the methodology and integration of vehicle geometry into the code. Section III gives details on the use of higher-fidelity aerodynamics in the context of an aircraft design analysis problem. Finally, these tools will be tested on the Onera M6 and applied to three different vehicles in Section IV.

## II. Geometry Output

The first step in utilizing SUAVE with high-fidelity tools is the ability to create a geometric model of the vehicle. Several parametric geometry engines for aircraft exist including commercial products such as RAGE<sup>6</sup> and AVID PAGE, as well as open-source tools such as OpenVSP<sup>7</sup> and GeoMACH.<sup>4</sup> As SUAVE is intended to provide a complete open-source package for aircraft design, and one that will continue to grow over time, we have decided to use OpenVSP. The open-source community is encouraged to develop SUAVE with these other tools as well.

### A. OpenVSP

OpenVSP is our preferred geometry generation tool due to its active development. This tool allows the creation of arbitrary aircraft geometries from a handful of component classes, and can be operated through a Python API. It also comes with a variety of other features that are useful in analysis. In particular, we make use of tools for wetted area computation to estimate drag, as well as surface mesh generation for CFD. These processes are described in more detail in Section III.

### B. Aircraft Parameters

SUAVE has its own data structure for defining a vehicle.<sup>1</sup> This data structure was crafted to include what we believed to be the best set of parameters to define any vehicle at the conceptual level of design. However, the inputs for OpenVSP differ significantly from the SUAVE data structure, and a separate OpenVSP vehicle must be built here. For ease of use, a separate script is provided to accomplish this task. Only one conversion script is required for the aircraft classes explored here (a narrow-body airliner, a supersonic transport, and a blended wing body aircraft). A narrow-body airliner created with this script is shown in Figure 1.

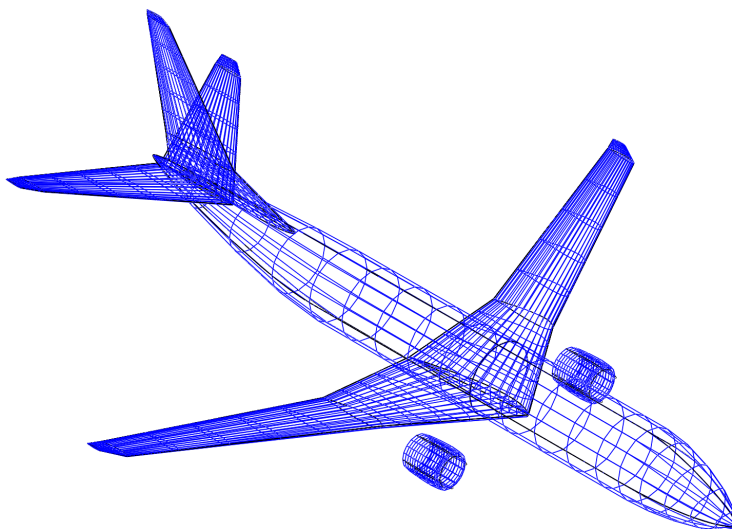


Figure 1: OpenVSP Wireframe Rendering of a Narrow-body Airliner

The user of the scripts is encouraged to double check the results before continuing with the analysis to ensure the suitability for their aircraft design. Even in more conventional cases and in cases where higher-fidelity methods may not be desired, this allows a quick visual check to ensure that the entered vehicle parameters produce the intended vehicle in SUAVE. In particularly unconventional cases, the user may need to create a new script to perform this conversion. In this event, the user should use the current method as a template. Additional vehicle classes of interest could include military fighters, rotorcraft, or launch vehicles. Users in the open-source community are encouraged to provide their conversion and vehicle scripts back to the community.

### 1. *Fuselage Parameters*

The fuselage is constructed using parameters for length, width, height, and fineness. Length, width, and height are used in a straight-forward way, with the length determined by the total length of the fuselage, and the width and height determining the width and height of the inner sections. By default, the fuselage is broken into four sections, with divisions at the end of nose, the main wing root quarter chord, and the beginning of the tail. Height can be specified individually for each of these. The position of the first and third divisions are determined by the nose and tail fineness respectively, which are ratios relating the nose and tail lengths to the width.

In its current form, SUAVE does not use parameters specifying nose or tail vertical position or other shape parameters. To account for this, the user must add a small data set if they wish to use custom values. This includes information on position, angles, and strength of those angles. For the tail, SUAVE enforces full symmetry due to peculiarities of the OpenVSP meshing tool. If no parameters are specified, the nose is centered and the tail is elevated at 20% of the fuselage diameter.

### 2. *Wing Parameters*

Overall wing parameters are specified as normal in SUAVE,<sup>1</sup> and can be translated to OpenVSP geometry fairly easily. However, these base parameters do not allow for cranked wing segments, airfoils, winglets, etc. To account for this, the ability to specify wing segments has been added to SUAVE. Creating a wing segment will allow the user to specify the airfoil, chord length, and twist at a particular spanwise position, along with the sweep and dihedral extending out to the next segment. Vertical wings and symmetry are specified in standard SUAVE parameters.

We recognize that different users will prefer to use the wing and segments in different ways. To account for this, the scripts that handle multiple sections are designed with three possible use cases in mind. The first is the basic case of a wing with no specified sections, which will output a simple trapezoidal planform. The second is a wing with only outboard sections specified. In this case, the innermost section will use values specified in the overall wing. This may be useful for cases where the inboard parameters match the parameters to be used in overall analysis, or for adding custom wingtips. The third case is a wing with all sections specified, which may be useful when the innermost section also does not match overall values. This last one is automatically used when the spanwise value of the first section is set to zero. Aside from adding segments in the vehicle setup, no additional modifications are necessary to pick between these options.

### 3. *Nacelle Parameters*

Within SUAVE, sizing routines exist to compute the length and diameter of a turbofan engine. The base routine requires the bypass ratio and thrust to compute these dimensions. With the length and diameter computed, the user can add extra length for the inlet nozzle and exit nozzle. Finally the nacelle is added as an OpenVSP fuselage component with loop type XSecs, a specific type of fuselage sections. The loop allows the nacelle to start and end with elliptical XSecs. The sizing of the XSecs creates the loft which allows flow through the engines.

## III. Aerodynamics with OpenVSP and SU2

Aerodynamic forces are often calculated using simplified models that do not accurately represent airfoil selection and other higher-order effects. These effects can become especially relevant with unconventional configurations. In these cases, there may be no data available to validate low-fidelity methods, so if results

are to be trusted, it is critical that they may be run with higher-fidelity CFD methods. In this work, lower-fidelity drag methods are augmented with wetted area calculation in OpenVSP, and lift is determined with Euler simulations in SU2. Here SU2 is not used to compute drag, and all drag calculations come from SUAVE methods discussed in previous publications.<sup>1</sup> However, this framework may be further enhanced to include this.

### A. Link with SUAVE

The overall tool-chain works as follows. The user inputs vehicle parameters and (if desired) mesh specific parameters in the vehicle setup file in SUAVE, and the SU2 analysis method is added to the analysis setup. This information is then converted to an OpenVSP vehicle. Once the geometry has been created, surface areas can be extracted and the vehicle surface mesh can be created. These surface areas are read back into SUAVE, and the surface mesh is passed to Gmsh where a volume mesh is created. An SU2 configuration file is then constructed for every point desired for a surrogate model, and SU2 is run with these settings. Finally, a surrogate is created and the mission is run using the standard SUAVE framework. Although in this case we are using only a lift surrogate, additional forces could be added without changing the general process. This process is shown in Figure 2.

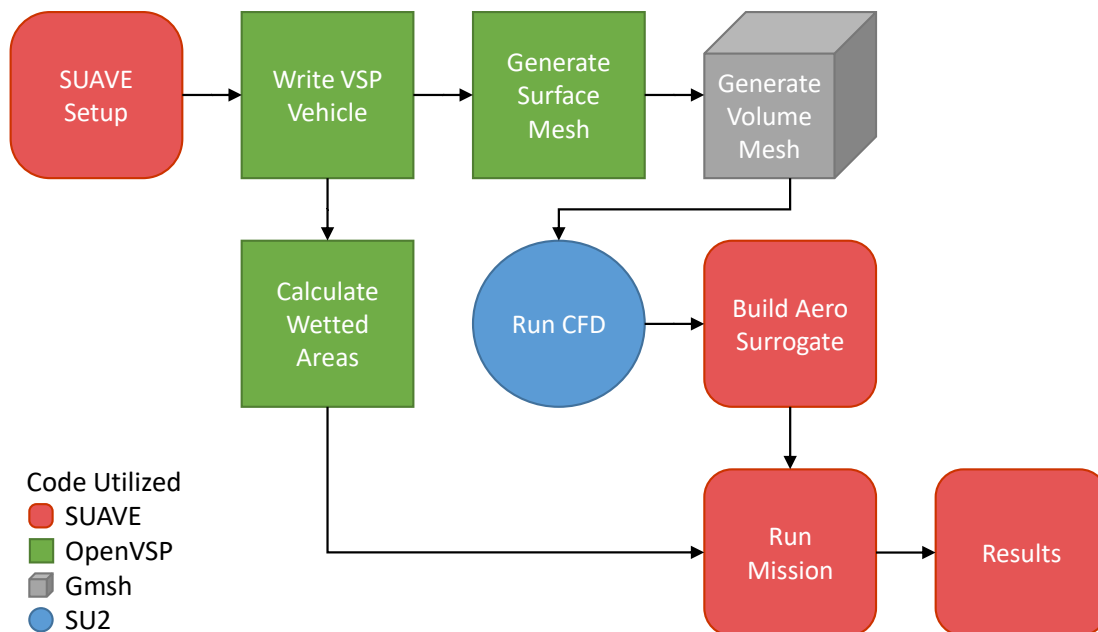


Figure 2: Tool Chain

### B. OpenVSP

#### 1. Wetted Area Computation

OpenVSP includes several tools that are useful for aerodynamic analysis. The first that we use here is computation of wetted areas. This is performed by determining component intersections and then finding the area of the outer surfaces. Data for each component is then output and read in SUAVE. A wing sizing script that executes this method through the OpenVSP Python API has been added to SUAVE. This can be included in a vehicle analysis by simply calling the sizing method in the vehicle setup. A visualization of this method is shown in Figure 3, with different colors representing different surfaces.

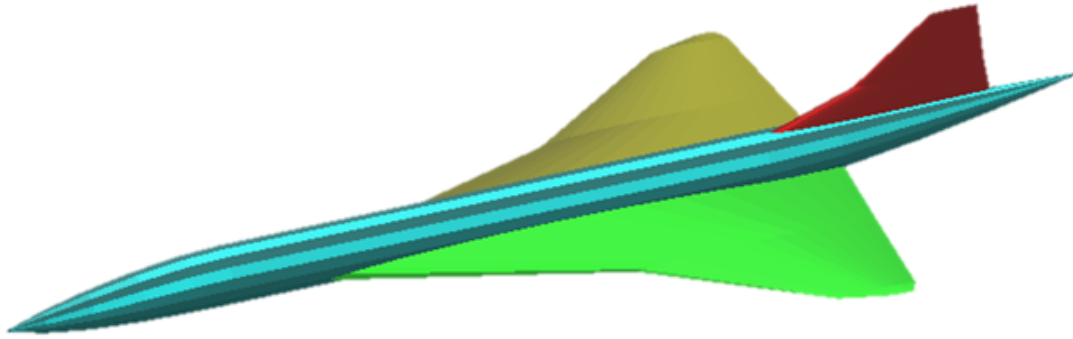


Figure 3: Supersonic Transport Split by Areas in OpenVSP

## 2. CFD Meshing

The surface meshing tool automatically creates a surface mesh that can be exported as a multi-solid STereoLithography file (.stl) for use in generating a volume mesh. Many other types of outputs are also possible if desired for other uses. This includes not just a vehicle surface mesh but also far-field and symmetry-plane meshes. The symmetry-plane mesh is created by default in the SUAVE analysis script, but can be disabled if desired. When the symmetry plane is set to be generated, symmetry plane splitting is activated to improve the robustness of the mesh generation. Without this, mesh generation is more likely to fail when thin components such as the vertical stabilizer are split.

To create the far-field mesh, SUAVE checks which dimension is the largest and sets the far-field parameters such that all sides of the box will be ten times that dimension apart. The size of this bounding box can be easily changed by the user if desired. This is a bit different than the OpenVSP default, which would create the far-field mesh such that it is four times the size of the vehicle across in each dimension respectively.

Other meshing parameters are also exposed through the link between these codes. One of the most useful of these, and the primary one available in the SUAVE vehicle setup, is the sources used on components. Sources are used to refine the mesh in a specified area by a specified amount. OpenVSP allows the user to create default sources based on what the developers believe is the most probable use case. SUAVE mimics this type of functionality for the wings and fuselages if no component specific sources are specified, and otherwise adjusts the cell edge length and effective radius of these sources based on user input. These sources still maintain the positioning of the default sources. Since the default analysis setup does not contain OpenVSP specific values, a special data set must be added to include these source modifications. More customized sources may be also added if needed, but must be done through a user created method outside of the analysis setup.

## C. Gmsh

Gmsh<sup>8</sup> is an open-source meshing tool commonly used for a variety of problems in aerospace and beyond, and has the ability to output meshes in SU2 format. It was chosen as the first meshing tool for integration over commercially developed tools in order to keep this tool chain completely open-source software, in line with the goal of having SUAVE as a tool that anyone is able to use and contribute to. However, users are encouraged to create and share links to other tools including commercial tools such as Pointwise that would be valuable to the community.

Gmsh also has the advantage of relatively simple use. The .geo files used to define Gmsh geometries are straight-forward and small, and can be run through Gmsh to generate a mesh without the need to build Gmsh from source. SUAVE builds these .geo files using data from the surface mesh and the vehicle to properly label mesh surfaces as the vehicle, the symmetry plane, and the far field. These labels are read into SU2 later on. The file contains the commands necessary to build a volume with surface mesh. Gmsh is then called via the command line to construct a volume mesh and output it in SU2 format.

## D. SU2

SU2 was chosen as the CFD solver for this work. This is a well-developed and validated open-source code suite which fits well with our goal of having all capabilities available through open source software.<sup>9</sup> SU2 is also developed heavily by the same research group responsible for SUAVE, making tight integration between codes much easier.

In the mission analysis process, SU2 is used to generate a surrogate for lift based on angle of attack and mach number. When the user chooses SU2 as an analysis method, they have the option of specifying the angles of attack and mach numbers they wish to use to build a surrogate. If no values are selected, defaults are chosen and are based on what would typically be useful for a subsonic transport. A supersonic analysis mode is also available and will create separate surrogates for the subsonic and supersonic regimes.

Once the analysis is chosen, this module will take the selected points and build and run an SU2 config file for each test point. When the run is complete, a surrogate is created using scikit-learn's Gaussian Process Regression module by default, which models each data point as a normalized Gaussian variable.<sup>10</sup> The resulting multivariate Gaussian distribution is then queried to interpolate between the data points. The data used to build the surrogate is also saved so that it can be used without rerunning SU2 if desired. The ability to use old data is an option in the analysis settings.

In the configuration of SU2 we default several parameters that users may tweak. To accelerate convergence we set an adaptive CFL number as a default. With the Euler simulations performed here we have noticed a tremendous change in computational time to convergence without sacrificing final results.

The large number of samples required to populate the surrogate led the team to consider using restarts to also speed convergence. However, it was decided against using restarts by default because of the possibility of introducing hysteresis effects in the results from the changing initial conditions.

The convergence of the problem was set to be a Cauchy residual on Lift coefficient. The tolerance is defaulted to 10e-6, as that was reasonable for SUAVE's default mission solver tolerance. The defaults also set a maximum of 1500 iterations. This number was decided on as we noticed that most cases converge within 300 iterations.

If the installation of SU2 on the system is compiled with parallel support SUAVE can launch a parallel process of SU2. This is done through SU2's Python interface. This is different from a serial process of SU2 where command line inputs are used to execute the process to reduce overhead. If a parallel installation is present the user must set the number of processes desired.

## IV. Cases

### A. Onera M6

The Onera M6 wing is a common reference case used to validate CFD results. It serves as a check that this new tool chain is performing as expected. The first check is to ensure that the mesh is of the appropriate resolution for transonic problems. A top view of the mesh can be seen in Figure 4. One can see that the CFD sources create finer cells around the leading and trailing edges but leave the mid-chord region cells larger in size. The corresponding volume mesh has 127,200 elements.

In this case we are not interested in running a mission with SUAVE, rather we only want to execute the aerodynamic analysis. Here we just run the tool-chain and query the results. A symmetry plane is established at the wing root as performed in the original study. The Onera test case<sup>11</sup> is run at Mach 0.8395 and an angle of attack of 3.06 degrees. This geometry is created entirely from SUAVE using dimensions provided in reference 11 along with the airfoil coordinates.

The lift coefficient for the Onera M6 are 0.272 and the drag coefficient is 0.0167. This is similar to other Euler CFD results.<sup>12,13</sup> The pressure distribution is plotted in Figure 5. Slices also show reasonable correlation with experimental data, as shown in Figures 6 and 7. Though this was used to test functionality, full mesh refinement studies were not conducted.

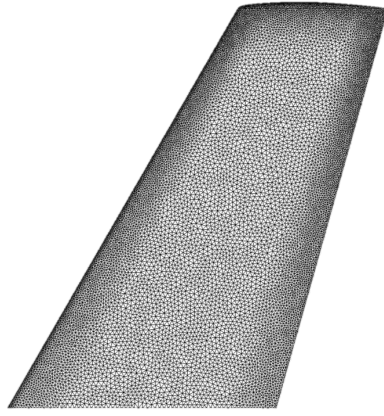


Figure 4: Onera M6 Mesh Top View

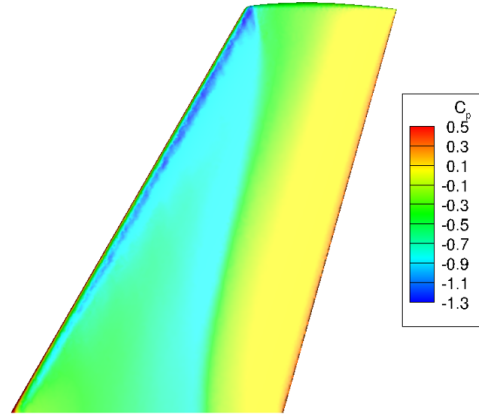


Figure 5: Onera M6 Cp Top View

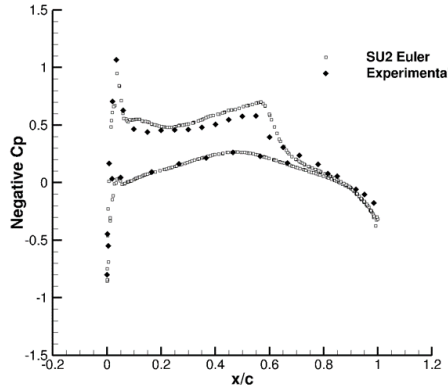


Figure 6: 20% Span

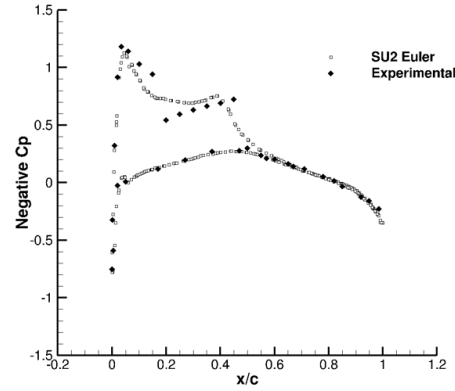


Figure 7: 65% Span

## B. Conventional Narrow-body Airliner

The first example case is a conventional, tube-and-wing jet, similar to the type used in previous SUAVE publications.<sup>1,2</sup> This is used to demonstrate SUAVE's ability to analyze and interface geometry for a well-validated configuration while leveraging higher-fidelity analysis. The 737 Classic airfoil was chosen to define the wing section. A vehicle mesh can be seen in Figure 8. The corresponding volume mesh has 768,069 elements. Nacelles are not used in the CFD calculations, but they are still included in the drag buildup.

The surrogate is constructed with three angles of attack and four Mach numbers. The angles of attack are 0, 3, and 6 degrees and the Mach numbers are 0.3, 0.5, 0.7, and 0.85. These 12 points cover the main flight envelope. We can visualize the space by plotting the results of the surrogate over a large domain as shown in Figure 9. This plot extends beyond the points interrogated by SU2 (marked) to show the functional behavior. We see that the aircraft exhibits strong effects on lift due to compressibility. Only lower angle of attack values are expected to be physical. We also look at the individual CFD results to judge the results of each run. Shown in Figure 10 is the pressure coefficient plot at Mach 0.7 and 3 degrees angle of attack.

The vehicle was then analyzed using the reference mission shown in previous SUAVE publications, but this time using the CFD data and OpenVSP wetted areas.<sup>1,2</sup> The profile of the mission is shown in Figure 11. The aerodynamic coefficients corresponding to the profile are shown in Figure 12. The lift to drag ratios as well as the angles of attack are representative of what one might expect from aircraft in this class. They also represent an increase in L/D from 16.2 to 17.0 versus the base SUAVE model, which translates to a decrease in fuel burn from 40,300 lb to 38,200 lb.

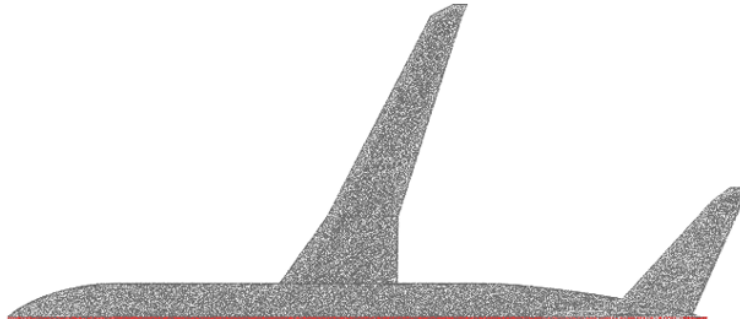


Figure 8: Narrow-Body Mesh Top View

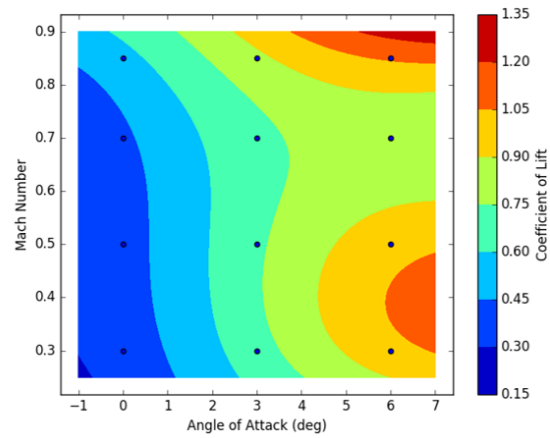


Figure 9: Narrow-Body Lift Coefficient Surrogate

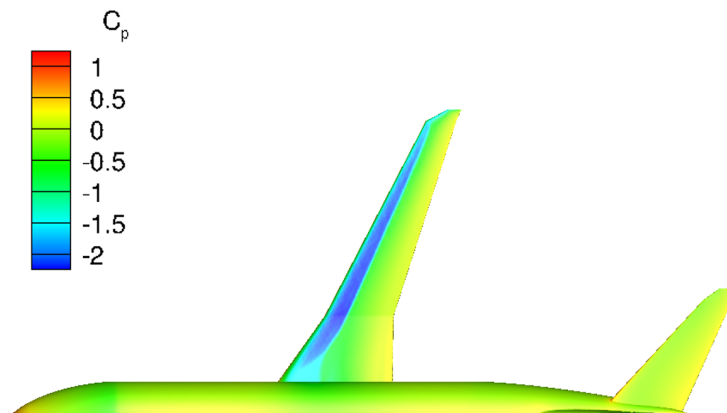


Figure 10: Narrow-Body Coefficient of Pressure



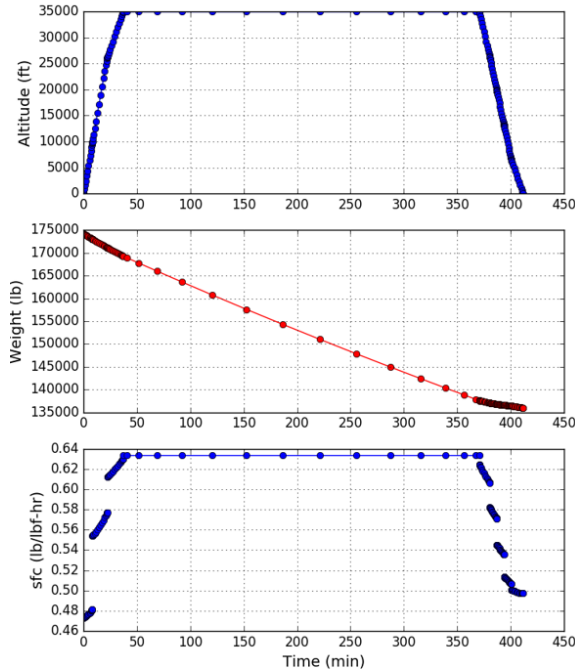


Figure 11: Narrow-Body Flight Profile

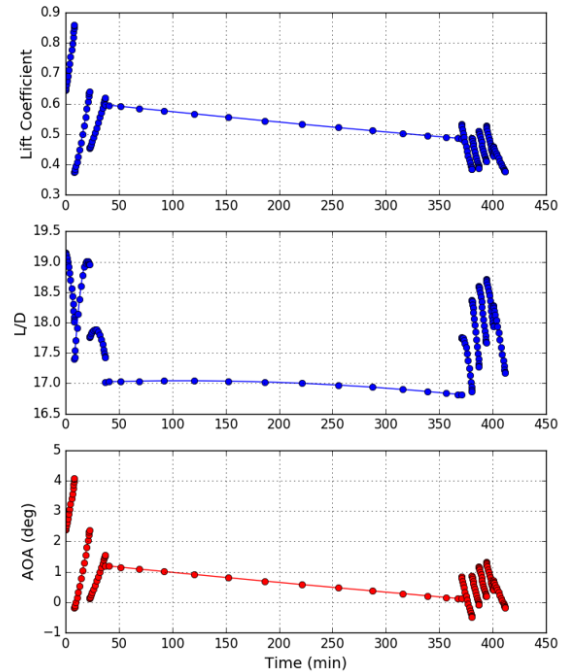


Figure 12: Narrow-Body Aerodynamic Data

### C. Supersonic Transport

The second case is a supersonic transport utilizing a delta wing, based on the Concorde. This demonstrates the ability of the link for use in supersonic design, and shows the surrogate created by merging two piece-wise surrogates from the subsonic and supersonic range to more accurately capture the function shape. Wing dimensions are set by measurements taken from available three-views of the Concorde. The airfoil used for the main wing is a NACA 65-203. This airfoil is somewhat different from the actual airfoil on the Concorde, but has been previously proposed for aircraft designed for supersonic flight.<sup>14</sup> A version of this was then modified to zero camber and 4% thickness to chord for use on the vertical stabilizer. Nacelles are not included in the CFD model, but are still used in the drag computations. Due to the very thin vertical tail, a full mesh was required in order to avoid meshing errors in OpenVSP. This mesh is shown in Figure 13. The volume mesh has 401,397 elements.

Five angles of attacks and eight mach numbers were used to generate the surrogate for this case. The angles were -1, 1, 3, 5, and 7 degrees, with Mach numbers of 0.25, 0.5, 0.75, 1.0, 1.25, 1.5, 1.75, and 2.0. The surrogate (CFD results marked) and SU2 results at 1 degree and Mach 2.0 are shown in Figures 14 and 15 respectively. OpenVSP wetted area calculation has also been incorporated. The mission used here is based on the mission flown by the Concorde, and is explained in more detail in a previous publication.<sup>1</sup> The mission profile is shown in Figure 16, while the aerodynamics results for this profile are shown in Figure 17. This is consistent with the base SUAVE analysis, indicating a good original set of areas.

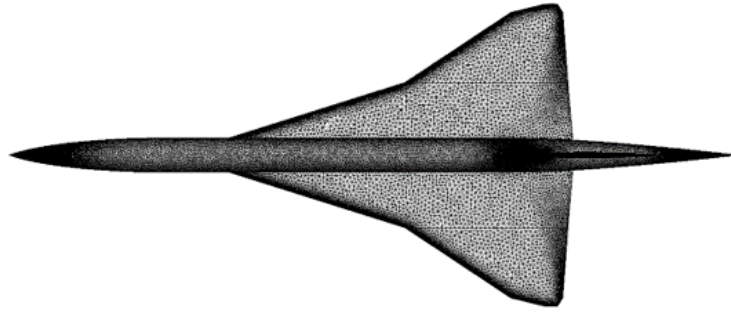


Figure 13: Supersonic Transport Mesh Top View

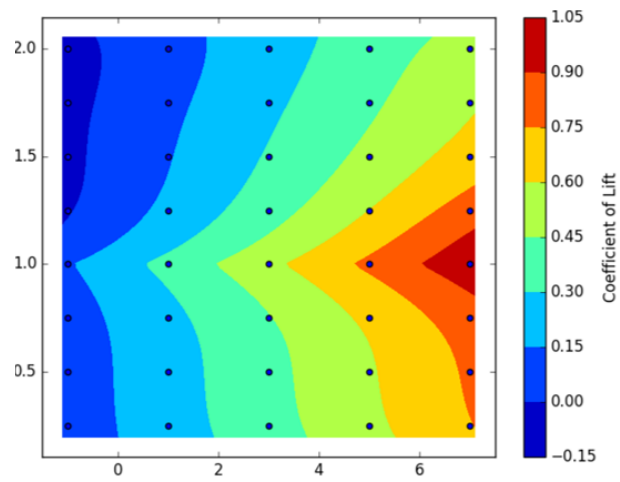


Figure 14: Supersonic Transport Lift Coefficient Surrogate

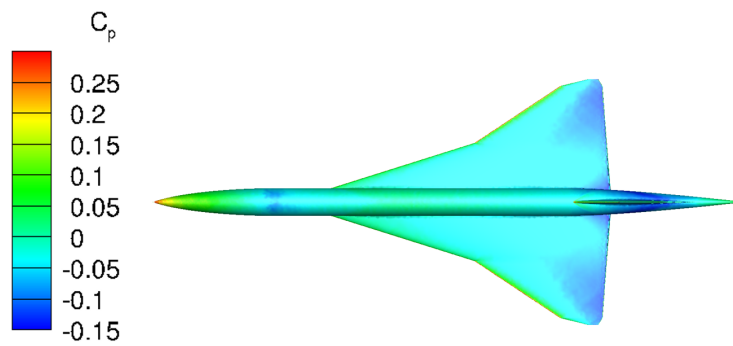


Figure 15: Supersonic Transport Coefficient of Pressure

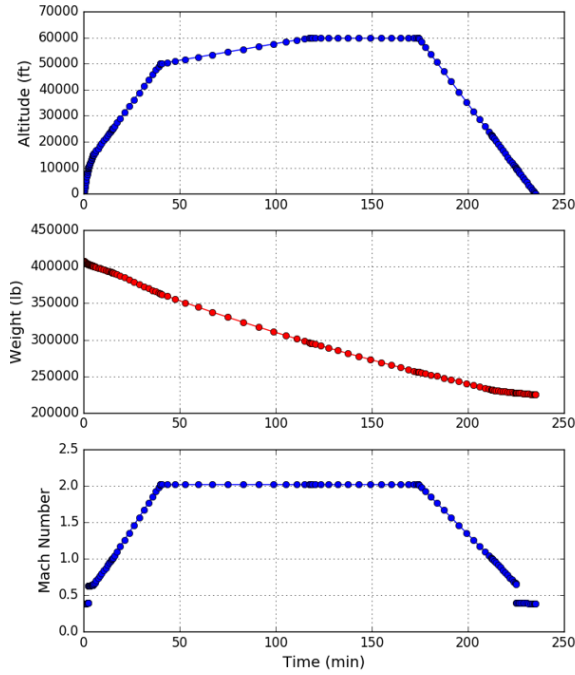


Figure 16: Supersonic Transport Flight Profile

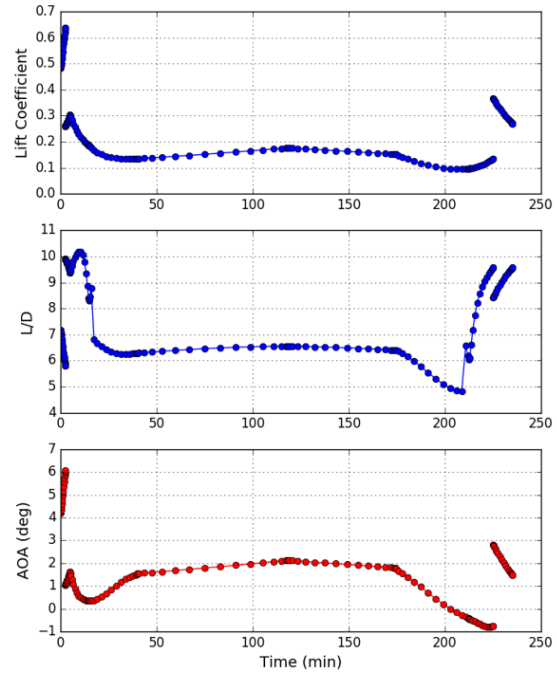


Figure 17: Supersonic Transport Aerodynamic Data

#### D. Blended Wing Body Transport Jet

The final test case is a blended wing body transport (BWB). To demonstrate the ability to analyze unconventional configurations we chose a design based on the Boeing BWB-450 design from Liebeck.<sup>15</sup> Here we make two changes that cause the design to differ from the original. First we forego the embedded nacelles with boundary layer ingestion due to a lack of knowledge of the internal ducting. This change is necessary to simplify the design for modeling, even though the nacelles still do not appear for CFD. Second, we replace the proprietary airfoils with symmetric airfoils of the correct thickness to chord ratios. Reasonable replacements that would provide reflex while performing well at transonic speeds are relatively difficult to obtain and do not affect the analysis process. A top view of the final mesh is shown in Figure 18. The volume mesh has 217,023 elements.

A surrogate is generated with the same angles of attack and Mach numbers as the narrow-body case, 0, 3, 6 and 0.3, 0.5, 0.7, 0.85 respectively. The surrogate plot for the BWB is thought to be more predictable as seen in Figure 19 (CFD runs marked) because there are no wing-fuselage interactions like in a conventional design. The zero lift angle is also at zero degrees angle of attack as expected for a symmetric airfoil with no twist. As an example of the SU2 CFD results the pressure contours at a freestream Mach number of 0.7 and an angle of attack of 3 degrees are plotted in Figure 20.

The BWB mission is that of a conventional airliner but with a design range of 7000 nautical miles. The cruise Mach number is set to 0.78. For propulsion it is assumed that the vehicle has three GE-90 class turbofans installed. The climb and descent profiles shown in Figure 21 match that of the narrow-body shown earlier.

One of the main reasons for using a BWB type aircraft is the high lift to drag ratios; we can see this when plotting the aerodynamic results from SUAVE in Figure 22. When compared against the narrow-body airliner we see a significant difference in lift to drag ratio. This is similar to the cruise lift to drag ratio predicted by Liebeck of 23.

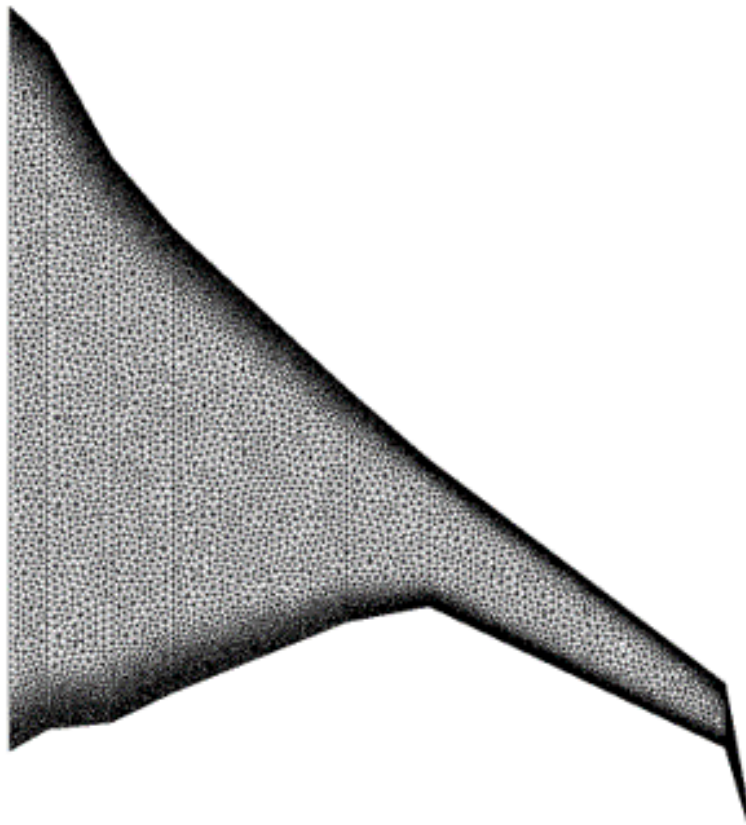


Figure 18: BWB Mesh

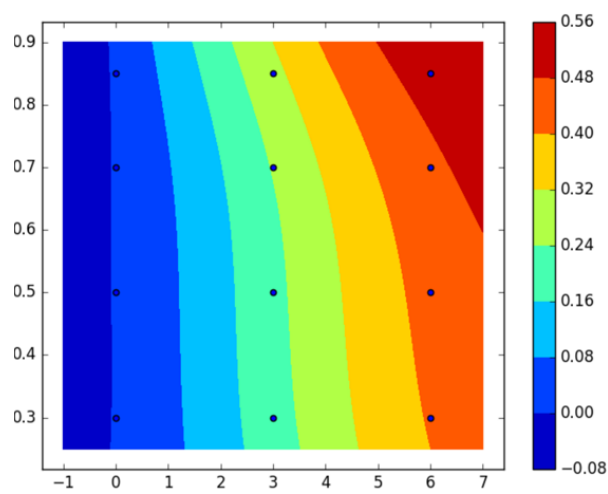


Figure 19: BWB Lift Coefficient Surrogate

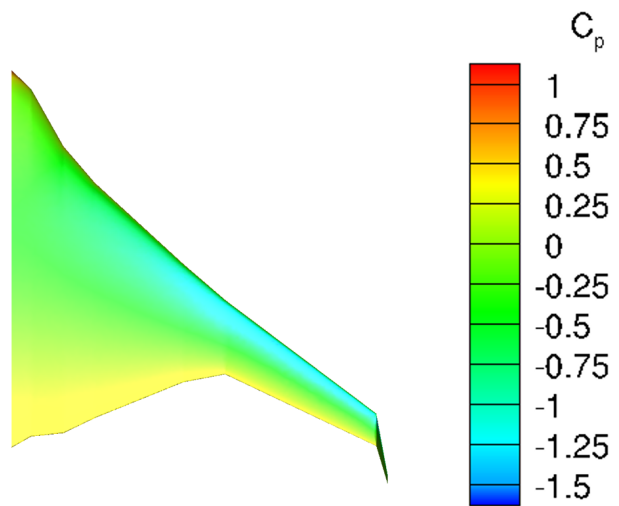


Figure 20: BWB Coefficient of Pressure

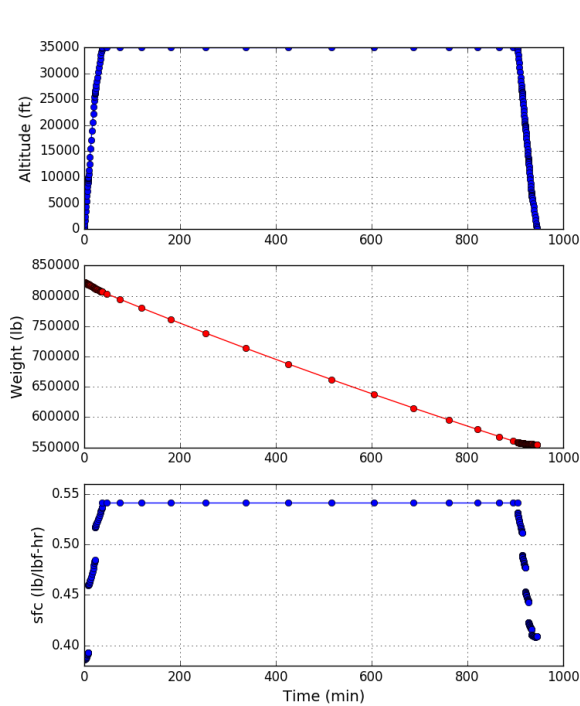


Figure 21: BWB Flight Profile

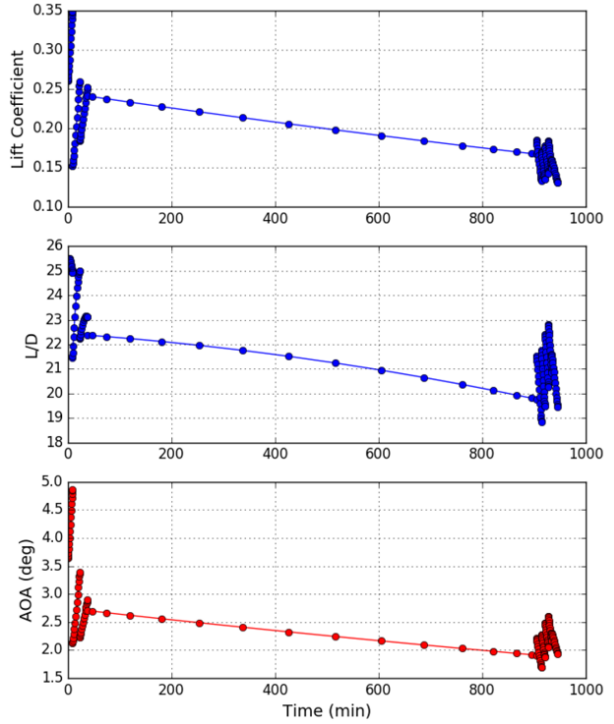


Figure 22: BWB Aerodynamic Data

## V. Summary

This work demonstrates how SUAVE is able to analyze unconventional configurations using a link to higher-fidelity tools. We established a tool-chain which can be used for a variety of aircraft types with Euler simulation using established open-source tools. This tool-chain is an improvement upon prior MDO work because of the level of automation, ease of setup, open-source nature, and the unified environment to drive the process. It can also be easily modified to permit surrogates of different values such as drag.

The use of OpenVSP enables SUAVE users to do things beyond CFD including the ability to read in accurate wetted areas not based on correlations and to visualize configurations. Gmsh can very quickly generate SU2 native volume meshes, and has widespread use already. SU2 has been chosen as the CFD solver as it has been extensively developed, is open source, is developed in part by the same research group, and can be easily queried from Python.

Four cases were evaluated here. First these methods were verified using the Onera M6. Next, a conventional case was shown with the narrow-body transport. Then we moved to less conventional configurations with a supersonic transport and a BWB. This serves as a demonstration of the flexibility of this process as the same code base can create these vastly differing geometries.

The underlying code used to generate these geometries has been released to the community to allow those interested to continue to use and develop these tools further. These efforts are all hosted on Github for the curious user.<sup>16</sup>

This work is the start of future developments for including increasingly higher-fidelity design strategies into conceptual design. The key here is to make the inclusion of these tools simple, reliable, and flexible enough to handle new, unthought of aerospace vehicles. Further integration with SU2 will occur to tightly couple and ease the connectivity for users. Future work will also include structural based modeling and weight estimation, RANS CFD simulations, as well as frameworks for handling these differing levels of fidelity within analysis and optimization.

## Acknowledgments

Timothy MacDonald and Emilio Botero would like to acknowledge the support of the Department of Defense (DoD) through the National Defense Science & Engineering Graduate Fellowship (NDSEG) Program.

Michael Vegh would like to acknowledge the support of the DoD through the Science Mathematics and Research for Transformation (SMART) Scholarship Program.

Anil Variyar would like to acknowledge the support of the Stanford Graduate Fellowship.

## References

- <sup>1</sup>Lukaczyk, T., Wendorff, A. D., Botero, E., MacDonald, T., Momose, T., Variyar, A., Vegh, J. M., Colonno, M., Economon, T. D., Alonso, J. J., Orra, T. H., and Ilario da Silva, C., “SUAVE: An Open-Source Environment for Multi-Fidelity Conceptual Vehicle Design,” *AIAA Aviation*, Dallas, TX, June 2015.
- <sup>2</sup>Botero, E., Wendorff, A. D., MacDonald, T., Variyar, A., Vegh, J. M., Alonso, J. J., Orra, T. H., and Ilario da Silva, C., “SUAVE: An Open-Source Environment for Conceptual Vehicle Design and Optimization,” *AIAA Scitech*, San Diego, CA, January 2016.
- <sup>3</sup>Variyar, A., Economon, T. D., and Alonso, J. J., “Multifidelity Conceptual Design and Optimization of Strut-Braced Wing Aircraft using Physics Based Methods,” *54th AIAA Aerospace Sciences Meeting*, 2016, p. 2000.
- <sup>4</sup>Hwang, J. T. and Martins, J. R. R. A., “GeoMACH: Geometry-Centric MDAO of Aircraft Configurations with High Fidelity,” *Proceedings of the 14th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference*, Indianapolis, IN, September 2012, pp. AIAA 2012-56051.
- <sup>5</sup>Böhnke, D., Nagel, B., Zhang, M., and Rizzi, A., “Towards a collaborative and integrated set of open tools for aircraft design,” *51st AIAA aerospace sciences meeting including the new horizons forum and aerospace exposition, grapevine (Dallas/Ft. Worth Region), Texas, USA*, 2013, pp. 7–10.
- <sup>6</sup>David, L. R. and Sturdza, P., “A rapid geometry engine for preliminary aircraft design,” Tech. rep., AIAA-2006-0929, 2006.
- <sup>7</sup>Hahn, A. S., “Vehicle sketch pad: a parametric geometry modeler for conceptual aircraft design,” *48th AIAA Aerospace Sciences Meeting and Exhibit*, 2010.
- <sup>8</sup>Geuzaine, C. and Remacle, J.-F., “Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities,” *International Journal for Numerical Methods in Engineering*, Vol. 79(11), 2009, pp. 1309–1331.
- <sup>9</sup>Palacios, F., Colonno, M. R., Aranake, A. C., Campos, A., Copeland, S. R., Economon, T. D., Lonka, A. K., Lukaczyk, T. W., Taylor, T. W. R., and Alonso, J. J., “Stanford University Unstructured (SU2): An open-source integrated computational environment for multi-physics simulation and design,” *51st AIAA Aerospace Sciences Meeting and Exhibit*, Grapevine, TX, January 2013.
- <sup>10</sup>Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E., “Scikit-learn: Machine Learning in Python,” *Journal of Machine Learning Research*, Vol. 12, 2011, pp. 2825–2830.
- <sup>11</sup>Schmitt, V. and Charpin, F., “Pressure distributions on the ONERA-M6-wing at transonic Mach numbers,” *Experimental data base for computer program assessment*, Vol. 4, 1979.
- <sup>12</sup>Straathof, M. H. and van Tooren, M. J., *Adjoint optimization of a wing using the CSRT method*, American Institute of Aeronautics and Astronautics (AIAA), 2011.
- <sup>13</sup>Viviand, H., “Numerical Solutions of Two-Dimensional Reference Test Cases,” *Test Cases for Inviscid Flow Field Methods, AGARD-AR-211*, 1985.
- <sup>14</sup>Darcy L Allison, Craig C Morris, J. A. S. R. K. K. and Watson, L. T., “Reevaluating conceptual design fidelity: An efficient supersonic air vehicle design case,” *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, July 2015.
- <sup>15</sup>Liebeck, R. H., “Design of the blended wing body subsonic transport,” *Journal of aircraft*, Vol. 41, No. 1, 2004, pp. 10–25.
- <sup>16</sup>Botero, E., Lukaczyk, T., timdmacdo, jmvegh, da Silva, C. R. I., tmomose, tarikorra, awendorff, Kruger, M., anilvar, and fcapristan, “suavecode/SUAVE: SUAVE Release 0.9.0,” Jan. 2017.