Aviary: An Open-Source Multidisciplinary Design, Analysis, and Optimization Tool for Modeling Aircraft with Analytic Gradients

Jennifer Gratz*, Jason Kirk†, Carl Recine§, John Jasa**, Eliot Aretskin-Hariton‡, Kenneth Moore**, Darrell Caldwell††, Kenneth Lyons§, Sydney Schnulo‡, Robert Falck‡, Herb Schilling‡, Erik Olson†, Dahlia Pham§, Xun Jiang††, Samara Murri††, Ben Margolis§, Eric Hendricks‡, Justin Gray‡, Bret Naylor**, Kiran Marfatia§§, Greg Wrenn††
National Aeronautics and Space Administration

Demands on aircraft design methods in recent years have begun to require increasingly higher amounts of coupling between disciplines and have also begun to require optimization in order to satisfy competing objectives involving large numbers of parameters that define unconventional configurations. These expanding requirements have amplified a need for new and improved aircraft design, analysis, and optimization codes that are capable of performing coupled design exploiting analytic gradients where possible. Aviary is a multidisciplinary design optimization and analysis framework which allows for tightly coupled simultaneous aircraft and subsystem design using analytic gradients. Aviary has employed the methods of two legacy aircraft analysis tools to provide native analytically differentiated calculations for five different disciplines, and it also has the ability to couple in external discipline analysis tools, whether or not those tools can provide analytic gradients. Preliminary examples and modeling efforts have shown Aviary's ability to effectively model novel concepts and explore large and non-intuitive design spaces. Finally, a multi-level user interface in Aviary creates an easy entry point for users with any level of multidisciplinary design, analysis, and optimization experience.

*NASA Glenn Research Center, Cleveland, OH, USA, jennifer.d.gratz@nasa.gov
†NASA Langley Research Center, Hampton, VA, USA
§NASA Ames Research Center, Mountain View, CA, USA
**NASA Glenn Research Center, Cleveland, OH, USA, Banner Quality Management Inc.
‡NASA Glenn Research Center, Cleveland, OH, USA
††NASA Langley Research Center, Hampton, VA, USA, Analytical Mechanics Associates Inc.
§§NASA Glenn Research Center, Cleveland, OH, USA, Intern

Nomenclature

Acronyms

2DOF	Two Degrees of Freedom
AATT	Advanced Air Transport Technology
AAVP	Advanced Air Vehicles Program
ANOPP2	Aircraft NOise Prediction Program 2
ARMD	Aeronautics Research Mission Directorate
EDS	Environmental Design Space
EPFD	Electric Powered Flight Demonstration
FLOPS	Flight Optimization System
GASP	General Aviation Synthesis Program
IASP	Integrated Aviation Systems Program
MDAO	Multidisciplinary Design Analysis and Optimization
NPSS	Numerical Propulsion System Simulation
OAS	OpenAeroStruct
PTIRS	Probabilistic Technology Investment Ranking System
TACP	Transformational Aeronautic Concepts Program
TACS	Toolkit for the Analysis of Composite Structures
TTBW	Transonic Truss-Braced Wing
TTT	Transformational Tools and Technologies
UAM	Urban Air Mobility

I. Introduction

The design and analysis of aircraft models is a field which has changed and grown significantly over time. Until somewhat recently, the state of the art in aircraft design and analysis was a method that is often colloquially called the over-the-wall method, where experts in different disciplines such as engine design, aerodynamic analysis, and performance analysis would independently design their portion or discipline of the aircraft, and then they would manually bring their designs together in an iterative process. This process required a human to drive the iteration loop between these different discipline experts, and at the time this was an effective and successful process. Eventually aircraft analysis tools began to emerge, and these tools improved this process. Flight Optimization System (FLOPS), an aircraft configuration optimization program [1], improved the process significantly by providing a framework where less human interaction was required in the design iterations, and where some optimization was possible. FLOPS was used to perform conceptual design, though it still required weights, aerodynamics, and propulsion analysis experts to design their subsystems separately and provide data if anything more than simple empirical relations was necessary. Another example code which took a step in the right direction is the General Aviation Synthesis Program (GASP), a preliminary aircraft design program that gives an aircraft analyst the ability to rapidly perform parametric studies [2]. This also improved the state of the art in aircraft design, although GASP, similar to FLOPS, still relied heavily on aerodynamicists and propulsion analysts building data tables that were manually provided to the aircraft sizing process. These "legacy" aircraft analysis tools had several benefits, some of which are still highly relevant today. They were relatively fast at performing their analyses (and are even faster now on modern hardware), they provided good default values that would produce reliable answers without the user specifying every single input, and their results were validated and trusted within the category and class of aircraft that they were modeled after. These benefits made the legacy tools valuable when they were built, and for some applications these legacy tools are still very useful today. However, there are some applications which are beginning to move beyond what the legacy tools were designed to handle.

Over time, the demands on aircraft modeling software increased with the increased demand for unconventional aircraft configurations, such as electric aircraft [3], hybrid-electric aircraft [4], and urban air mobility vehicles [5]. These unconventional configurations raised a new problem: the disciplines of these aircraft were more tightly coupled than the disciplines of traditional subsonic transport type aircraft, and the traditional methods of having subsystem experts build data tables which they manually iterated on with the aircraft modeling experts no longer were sufficient. The new vehicles required a new vehicle-analysis paradigm, one where discipline tools could be directly coupled to each other without a human in the loop. [5] [3] [6] A few examples of the need for and success of this coupled approach can be seen in Heath et. al. [7], Li et. al. [8], and Hendricks et. al. [5] Heath et. al. and Li et. al. demonstrate the need for coupled disciplines in the high speed regime of supersonics, and demonstrate that coupled aerodynamic and propulsion optimizations can and have been done. Both Heath et. al. and Li et. al. performed coupled aeropropulsive design of a supersonic aircraft in order to improve the sonic boom and design a low boom supersonic aircraft. [7] [8] This coupling was necessary in order to capture the engine installation effects on the sonic boom. [7] On the other end of the speed spectrum Hendricks et. al. demonstrated the need for coupled disciplines in an Urban Air Mobility (UAM) environment. They compared the design of a baseline tiltwing aircraft to a few different re-designs which contained increasing amounts of coupling, and they showed that the fully coupled design produced the best aircraft (increased overall efficiency and reduced power requirements). [5] These examples at the high-speed and low-speed ends of the aircraft speed regime demonstrate the importance of coupling between different disciplines in unconventional aircraft, and they also demonstrate that successful efforts have been made to find design improvements by exploiting these discipline couplings.

One significant desire that emerged out of the need for more tightly coupled disciplines was the enhancement of those couplings by analytic gradients. In particular, Multidisciplinary Design Analysis and Optimization (MDAO) emerged as a new and valuable discipline in and of itself for aircraft design; and analytic gradients quickly became an ingredient for computational speed. The reason for gradients playing a critical role was explained by Gray et. al.: "gradient-based optimization algorithms scale much better with the number of design variables than gradient-free methods. The computational efficiency of both Newton-type analysis methods and gradient-based optimization is, in large part, dependent on the cost and accuracy with which the necessary derivatives are computed." [9] This shows the computational efficiency that can be achieved in MDAO by using computationally cheap and accurate derivatives. Due to the fact that computing analytic derivatives requires fewer function calls than obtaining derivatives via finite-differences, and that analytic derivatives give precise numerical results instead of estimates, analytic derivatives present themselves as computationally superior to finite-difference derivatives, all else being equal. The effects of this reality were shown by Hendricks et. al. when they did a comparison of optimization computation time for engine cycle analysis codes with and without analytic gradients. [10] They found that for their study the derivative computation time was more than 300 times shorter when using an analytically differentiated code as opposed to a non-analytically differentiated code. [10] Thus, as aircraft concepts become more tightly coupled between disciplines, they call for codes with a few specific features: the ability to couple together discipline and analysis tools, object-oriented building blocks which can be used for this coupling of disciplinary tools, the ability to perform efficient computation across an extensive number of coupled parameters, and analytic gradients to speed up this computational process.

II. Background

Several efforts were made to adapt the existing aircraft design tools such as FLOPS and GASP to these increased demands. One framework that was developed, Environmental Design Space (EDS) [11], uses Numerical Propulsion System Simulation (NPSS) [12] as a driver for various disciplinary models. In addition, various studies such as a hybrid electric aircraft study out of the NASA Glenn Research Center built their own multidisciplinary drivers using FLOPS [4], and various updates were made to GASP to begin adapting to the new electric aircraft needs. However, across all of these efforts there were still pieces that needed to be added. Many of the independent frameworks built under FLOPS still required some sort of data table to be designed outside the framework, the updates to GASP accounted for electric aircraft somewhat effectively but did not couple in other disciplines, and none of the studies mentioned included analytic gradients. They were all stepping stones on the way to an aircraft design method that meets the needs of the emerging unconventional concepts, but there was still room for further development.

Outside of NASA, a multitude of conceptual-level aircraft design tools exist from both academia and industry. SUAVE [13] was created to analyze and optimize conventional and unconventional aircraft configurations. It is open-source and is also multi-fidelity in that it can combine different subsystem models at varying levels of fidelity. FAST-OAD [14] is an overall aircraft design tool that focuses on modularity and usability. It comes with empirical models for aerodynamics, propulsion, structures, and other disciplines based on commercial aircraft from the 90s and 00s. AeroSandbox [15] is a conceptual-level open-source aircraft design tool that focuses on being fully differentiable, which allows the use of efficient gradient-based optimization. It also allows for interfacing with user-defined external tools or additional disciplinary subsystems. There are many other tools too numerous to detail here and relatively limited information is available on tools internal to large aerospace design companies such as Boeing and Airbus. Although each of these aircraft design tools is useful, a single tool that combines analytic derivatives and coupled aircraft-mission design optimization in an open-source package did not exist prior to the work presented here.

The codes and tools mentioned so far lie on a spectrum of aircraft design frameworks which runs from a simple analysis tool with no analytic derivatives and requiring multiple data tables to be provided, all the way up to a fully integrated and completely differentiated tool which couples in every discipline and has analytic gradients for every discipline in a single framework. The ideal end of this spectrum has not yet been reached by the state of the art. OpenMDAO [9] [16] is a framework that makes the use of analytic gradients in a coupled optimization environment possible, but it only provides the bones for an MDAO aircraft tool, it does not actually provide any of the aircraft disciplines or an interface that aircraft analysts unfamiliar with an MDAO environment could use. Thus, OpenMDAO is not exactly a stepping stone on this spectrum, however, the capabilities of OpenMDAO can play a key role in advancing the state of the art in aircraft design tools by serving as a platform for those tools.

Aviary is a new open-source software built on top of OpenMDAO and written in python, and it takes another step along the spectrum towards a fully-coupled and fully analytically differentiated aircraft tool. The goal of Aviary is to provide an aircraft MDAO framework where disciplines can be tightly coupled together in a computationally efficient manner, where aircraft analysts can run designs, analyses, and optimizations of novel and unconventional aircraft concepts without having to modify the underlying code architecture, and where aircraft analysts can run the simpler end of these analyses without needing any expertise in MDAO. In pursuit of this goal, Aviary is a fully analytically differentiated aircraft design, analysis, and optimization tool that includes the low fidelity modeling techniques of both of NASA's existing legacy aircraft analysis tools FLOPS and GASP. In addition, Aviary includes a defined interface for coupling external disciplinary tools. It is important to notice that several of the external disciplinary tools which users may couple to Aviary do not have analytic gradients. In these cases Aviary takes advantage of the finite differencing and complex stepping capabilities of OpenMDAO to compute derivatives, but it simultaneously supports tools providing analytic derivatives if they are available. The support for analytic gradients in external tools will help Aviary to remain a tool of enduring relevance as the state of the art advances towards MDAO aircraft design.

This paper will show that Aviary is a multidisciplinary design optimization and analysis framework which allows for tightly coupled simultaneous aircraft and subsystem design using analytic gradients. Specifically this paper will discuss the legacy tool methods which are differentiated and re-used in Aviary, the evidence that the analytic derivatives of

Aviary's tools are fast compared to finite difference, the capability of external tool integration which Aviary has created, and the methods of user entry which Aviary has distinguished.

III. Analytically Differentiated Disciplines in Aviary

Aviary employs analytic gradients on legacy aircraft design methods for sizing, aerodynamic analysis, mission analysis, propulsion analysis, and geometry analysis. Specifically, Aviary has taken the legacy aircraft design methods from each of FLOPS and GASP in the disciplines listed in above, and has implemented those methods along with hand-calculated analytic gradients in the OpenMDAO framework. Below is a brief discussion of the originating methods for each discipline, and how those methods were adapted and combined together in the Aviary framework.

A. Sizing Analysis and Weights

For sizing analysis Aviary provides some built-in weight estimating relationships that have been fully differentiated. There are two different options for built-in weight estimating relationships in Aviary, one which has been ported over from the equations used in GASP that are documented by Hague et. al. [17], and another which has been ported over from the equations used in FLOPS that are documented by Wells et. al. [18] The uniqueness of the weight and sizing analysis in Aviary is not the origin of the equations, which are based on empirical relationships developed in the late 20th century. The uniqueness of the weight and sizing analysis comes from the combination of these two methods into one tool, and even more so from the hand differentiation of all the equations in both methods. Because of this hand differentiation, Aviary has analytic gradients across the entire weight and sizing routine.

This section will discuss where those methods came from and the details of how they were combined into one tool.

B. Aerodynamic Analysis

For aerodynamic analysis Aviary uses built-in empirical relationships and also provides the ability to read a data table that was generated by an outside aerodynamic analysis code. As mentioned in Section I, Aviary provides an improvement in the capability to fully couple together disciplinary analysis codes, a capability that Section I demonstrated the need for. Specifically, the aerodynamic portion of Aviary has three different options: 1) user-input of tabular aerodynamic data, 2) user-connection of another aerodynamic analysis tool, and 3) built-in empirical aerodynamic equations. Option 1 is present because some types of modeling will perform aerodynamic design separate from aircraft design, and Aviary needs the ability to handle these situations. Option 2 is the primary mode which allows Aviary users to couple aerodynamic disciplinary design to aircraft design. Option 3 is the continuation of the legacy empirical aerodynamic estimating relationships from GASP and FLOPS. Just like in the weight and sizing analysis, the aerodynamic equations from GASP [19] and FLOPS [1] have been combined into one tool within Aviary, and they have been fully analytically differentiated to provide analytic gradients across the entire aerodynamics routine.

The rest of this section will discuss where the empirical methods came from and the details of how we combined two empirical methods and a table-loading capability into one tool.

C. Mission Analysis

For mission analysis Aviary retains the dynamic equations from the legacy aircraft analysis tools, but employs different numerical integration schemes on those equations in order to provide optimal control capabilities as well as analytic gradients. There are two options for equations of motion in Aviary, one based on a height-energy approach [20] used in FLOPS and one based on a Two Degrees of Freedom (2DOF) approach used in GASP. The equations of motion listed in Francisco et. al. [20] and Hague et. al, [21] are implemented in Aviary, but the integration approaches mentioned in those works were not employed. Instead, Aviary uses two different integration schemes which the user can select between and apply to either set of dynamic equations. The first integration scheme Aviary offers is a collocation scheme built on top of OpenMDAO that takes advantage of the capability for analytic gradients and is implemented in a software called dymos. [22] The second integration scheme Aviary offers is a shooting with gradients method that also takes advantage of the capability for analytic gradients, and is implemented in a software called simupy [23]. There are tradeoffs between both types of dynamic equations as well as between both types of numerical integration.

The rest of this section will discuss where the equations came from and how Aviary employs different numerical integration techniques. Specifically it will discuss the pros and cons of the different approaches and when each approach is useful.

D. Propulsion Analysis

For propulsion analysis Aviary again provides the capability to read in a data table from an outside analysis tool. This capability is present because despite the value often added by capturing the coupled nature of engine design with other disciplines, there are times when it is not necessary or desirable to couple full engine cycle design with the aircraft analysis. However, one of the originating uses of Aviary is to couple to external engine design tools in order to perform coupled aeropropulsive optimizations. An example of using Aviary to couple to an external engine cycle analysis tool is a Transonic Truss-Braced Wing (TTBW) study performed in Aviary by researchers at NASA. [24]

The rest of this section will discuss where the data table interpolants came from, and the details of how we combined the interpolation techniques of both FLOPS and GASP into one tool. We will reference [1] and [25].

E. Geometry Analysis

For geometry analysis Aviary provides some built-in geometric estimating relationships that have been fully differentiated. There are two different options for built-in geometric analysis in Aviary, one which has been ported over from the equations used in GASP that are documented by Hague et. al. [26], and another which has been ported over from the equations used in FLOPS that are documented by McCullers et. al. [1] The uniqueness of the geometric analysis in Aviary is not the origin of the equations, which are based on empirical relationships developed in the late 20th century. The uniqueness of the geometric analysis comes from the combination of these two methods into one tool, and even more so from the hand differentiation of all the equations in both methods. Because of this hand differentiation, Aviary has analytic gradients across the entire geometry routine.

This section will discuss where those methods came from and the details of how they were combined into one tool.

IV. Speed of Derivatives

We will show that the derivatives of Aviary's modules are fast (as compared to finite difference) and accurate. For this demonstration we will perform timing studies on the derivatives from the model used in a TTBW study from reference [24]. This section will detail the execution, performance and results of those timing studies. For the studies we will extract derivative timing data from the Aviary model at three different points during the TTBW optimization: one point at the beginning of the optimization, one point in the middle of the optimization, and one point at the end of the optimization. We will run these exact points using finite-difference derivatives as well, and will compare the timing results of the original Aviary derivative timing (analytic gradients) to the finite difference derivative timing at all three points in the optimization. We will also compare the relative accuracy of these derivatives.

V. Integration of External Tools

In addition to providing calculations and analytic gradients for the disciplines listed above, Aviary enables the integration of external disciplinary tools into the framework, and it smoothly handles any analytic gradients that the tool provides. This allows for tight coupling to detailed disciplinary models. The rest of this section will detail how we perform this external tool coupling, and give some examples of this coupling. The external codes that we have integrated and may provide examples for are OpenAeroStruct (OAS) [27], VSPaero [28], OpenVSP [29], pyCycle [10], Toolkit for the Analysis of Composite Structures (TACS) [30], Aircraft NOise Prediction Program 2 (ANOPP2) [31], and Probabilistic Technology Investment Ranking System (PTIRS) [32]. These examples will demonstrate the basic computational efficiency of our approach, and we will show that Aviary's flexible integration capabilities and its ability to make use of analytic gradients allowed us to use the analytic gradients of the example tools when they were available.

VI. User Experience and User Entry Points

Aviary distinguishes different methods of user entry in order to enable flexible use of the tool. As mentioned in Section I, the goal of Aviary is to provide an aircraft MDAO framework where disciplines can be tightly coupled together with analytic gradients, where aircraft analysts can run designs, analyses, and optimizations of novel and unconventional aircraft concepts without having to modify the underlying code architecture, and where aircraft analysts can run the simpler end of these analyses without needing any expertise in MDAO. A significant part of that goal is providing different levels at which an Aviary used can access the code based on their level of expertise with MDAO and the various underlying softwares. This section will discuss how Aviary has broken the user-interface up into three different levels which are each targeted at a different user skill set and which each offer an increasing amount of code capability.

VII. Conclusion

This paper will have shown that Aviary is a multidisciplinary design optimization and analysis framework which allows for tightly coupled simultaneous aircraft and subsystem design using analytic gradients. This conclusion will re-address the legacy tool methods which are differentiated and re-used in Aviary, the evidence that the analytic derivatives of Aviary's methods are fast compared to finite difference, the capability of external tool integration which Aviary has created, and the methods of user entry which Aviary has distinguished.

In addition, this conclusion will note that Aviary has demonstrated its ability to achieve design insight on a non-traditional aircraft [24], and will discuss the future work for Aviary.

Acknowledgements

The work presented in this paper and the Aviary framework was developed with support from NASA's Transformational Tools and Technologies (TTT), Advanced Air Transport Technology (AATT), and Electric Powered Flight Demonstration (EPFD) Projects. These projects are funded by Transformational Aeronautic Concepts Program (TACP), Advanced Air Vehicles Program (AAVP), and Integrated Aviation Systems Program (IASP) respectively. Funding for those Programs is provided by the Aeronautics Research Mission Directorate (ARMD). In addition, this work is indebted to Ben Phillips from NASA Langley and Joseph Garcia from NASA Ames for their advisory role in the Aviary framework.

References

[1] McCullers, L. A., "Aircraft Configuration Optimization Including Optimized Flight Profiles," *Recent Experiences in Multidisciplinary Analysis and Optimization*, 1984, pp. 395–412. URL `https://ntrs.nasa.gov/citations/19870002310`.

[2] Hague, D., "Main Program," *GASP - General Aviation Synthesis Program*, Vol. 1, 1978. URL `https://ntrs.nasa.gov/api/citations/19810010562/downloads/19810010562.pdf`.

[3] Falck, R., Chin, J., Schnulo, S., Burt, J., and Gray, J., "Trajectory Optimization of Electric Aircraft Subject to Subsystem Thermal Constraints," Denver, Colorado, 2017. URL `https://arc.aiaa.org/doi/pdf/10.2514/6.2017-4002`.

[4] Antcliff, K. R., Guynn, M. D., Marien, T., Wells, D. P., Schneider, S. J., and Tong, M. J., *Mission Analysis and Aircraft Sizing of a Hybrid-Electric Regional Aircraft*, 2016. https://doi.org/10.2514/6.2016-1028.

[5] Hendricks, E., Falk, R., Gray, J., Aretskin-Hariton, E., Ingraham, D., Chapman, J., Schnulo, S., Chin, J., Jasa, J., and Bergeson, J., "Multidisciplinary Optimization of a Turboelectric Tiltwing Urban Air Mobility Aircraft," *American Institute of Aeronautics and Astronautics*, 2019. https://doi.org/https://doi.org/10.2514/6.2019-3551.

[6] Schnulo, S., "Overview of NASA GRC Electrified Aircraft Propulsion Systems Analysis Methods," URL: https://ntrs.nasa.gov/api/citations/20180001332/downloads/20180001332.pdf, 10 2017. EnergyTech 2017 Presentation.

[7] Heath, C., Seidel, J., and Rallabhandi, S. K., *Viscous Aerodynamic Shape Optimization with Installed Propulsion Effects*, 2017. https://doi.org/10.2514/6.2017-3046.

[8] Li, W., and Geiselhart, K., *Propulsion-Airframe Integration for Conceptual Redesign of a Low-Boom Supersonic Transport*, 2023. https://doi.org/10.2514/6.2023-1511.

[9] Gray, J. S., Hwang, J. T., Martins, J. R. R. A., Moore, K. T., and Naylor, B. A., "OpenMDAO: An open-source framework for multidisciplinary design, analysis, and optimization," *Structural and Multidisciplinary Optimization*, Vol. 59, No. 4, 2019, pp. 1075–1104. https://doi.org/10.1007/s00158-019-02211-z.

[10] Hendricks, E. S., and Gray, J. S., "pyCycle: A Tool for Efficient Optimization of Gas Turbine Engine Cycles," *Aerospace*, Vol. 6, No. 8, 2019. https://doi.org/10.3390/aerospace6080087.

[11] Nunez, L. S., Tai, J. C., and Mavris, D. N., *The Environmental Design Space: Modeling and Performance Updates*, 2021. https://doi.org/10.2514/6.2021-1422.

[12] Lytle, J. K., "The Numerical Propulsion System Simulation: A Multidisciplinary Design System for Aerospace Vehicles," 1999. URL https://ntrs.nasa.gov/api/citations/19990062672/downloads/19990062672.pdf.

[13] Lukaczyk, T. W., Wendorff, A. D., Colonno, M., Economon, T. D., Alonso, J. J., Orra, T. H., and Ilario, C., "SUAVE: an open-source environment for multi-fidelity conceptual vehicle design," *16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 2015, p. 3087.

[14] David, C., Delbecq, S., Defoort, S., Schmollgruber, P., Benard, E., and Pommier-Budinger, V., "From FAST to FAST-OAD: An open source framework for rapid Overall Aircraft Design," *IOP Conference Series: Materials Science and Engineering*, Vol. 1024, IOP Publishing, 2021, p. 012062.

[15] Sharpe, P. D., "AeroSandbox: A Differentiable Framework for Aircraft Design Optimization," Master's thesis, Massachusetts Institute of Technology, 2021.

[16] Heath, C., and Gray, J., "OpenMDAO: Framework for Flexible Multidisciplinary Design, Analysis and Optimization Methods," 2012. https://doi.org/10.2514/6.2012-1673.

[17] Hague, D., "Weights," *GASP - General Aviation Synthesis Program*, Vol. 5, 1978. URL https://ntrs.nasa.gov/api/citations/19810010566/downloads/19810010566.pdf.

[18] Wells, D. P., Horvath, B. L., and Mccullers, L. A., "The Flight Optimization System Weights Estimation Method," 2017. URL https://ntrs.nasa.gov/api/citations/20170005851/downloads/20170005851.pdf.

[19] Hague, D., "Aerodynamics," *GASP - General Aviation Synthesis Program*, Vol. 3, 1978. URL https://ntrs.nasa.gov/api/citations/19810010564/downloads/19810010564.pdf.

[20] Capristan, F. M., and Welstead, J. R., *An Energy-Based Low-Order Approach for Mission Analysis of Air Vehicles in LEAPS*, 2018. https://doi.org/10.2514/6.2018-1755.

[21] Hague, D., “Performance,” *GASP - General Aviation Synthesis Program*, Vol. 6, 1978. URL https://ntrs.nasa.gov/api/citations/19810010567/downloads/19810010567.pdf.

[22] Falck, R., Gray, J. S., Ponnapalli, K., and Wright, T., “dymos: A Python package for optimal control of multidisciplinary systems,” *Journal of Open Source Software*, Vol. 6, No. 59, 2021, p. 2809. https://doi.org/10.21105/joss.02809.

[23] l. Margolis, B. W., “SimuPy: A Python framework for modeling and simulating dynamical systems,” *Journal of Open Source Software*, Vol. 2, No. 17, 2017, p. 396. https://doi.org/10.21105/joss.00396.

[24] Aretskin-Hariton, E., Gratz, J., Jasa, J., Moore, K., Falck, R., Kuhnle, C., Hendricks, E., Kirk, J., Olson, E., Caldwell, D., Recine, C., and Lyons, K., *Multidisciplinary Optimization ofa Transonic Truss-Braced Wing Aircraft using the Aviary Framework*, to appear in SciTech 2024 forum.

[25] Hague, D., “Propulsion, Part I - Theoretical Development,” *GASP - General Aviation Synthesis Program*, Vol. 4, 1978. URL https://ntrs.nasa.gov/api/citations/19810010564/downloads/19810010564.pdf.

[26] Hague, D., “Geometry,” *GASP - General Aviation Synthesis Program*, Vol. 2, 1978. URL https://ntrs.nasa.gov/api/citations/19810010563/downloads/19810010563.pdf.

[27] Jasa, J. P., Hwang, J. T., and Martins, J. R. R. A., “Open-source coupled aerostructural optimization using Python,” *Structural and Multidisciplinary Optimization*, 2018. https://doi.org/10.1007/s00158-018-1912-8.

[28] NASA, “Software Package, Version 3.31.1,” URL: http://openvsp.org/, 2022. NASA.

[29] McDonald, R. A., and Gloudemans, J. R., *Open Vehicle Sketch Pad: An Open Source Parametric Geometry and Analysis Tool for Conceptual Aircraft Design*, 2021. https://doi.org/10.2514/6.2022-0004.

[30] Kennedy, G. J., and Martins, J. R., “A parallel finite-element framework for large-scale gradient-based design optimization of high-performance structures,” *Finite Elements in Analysis and Design*, Vol. 87, 2014, pp. 56–73. https://doi.org//10.1016/j.finel.2014.04.011.

[31] Lopes, L., and Burley, C., “Design of the Next Generation Aircraft Noise Prediction Program: ANOPP2,” *American Institute of Aeronautics and Astronautics*, AIAA, 2011.

[32] Frederic, P., Bezos-O’Connor, G. M., and Nickol, C., *Cost Analysis Approach in the Development of Advanced Technologies for Green Aviation Aircraft*, John Wiley and Sons, Ltd, 2016, pp. 1–13. https://doi.org//10.1002/9780470686652.eae1078.