

# Trajectory optimization of Transonic Truss Braced Wing using Medium-Fidelity Aerodynamics

**Tim Brooks**  
**NASA GRC [BQMI]**  
**Cleveland, OH**

**October 24, 2024**

# MBSA&E Phased Approach

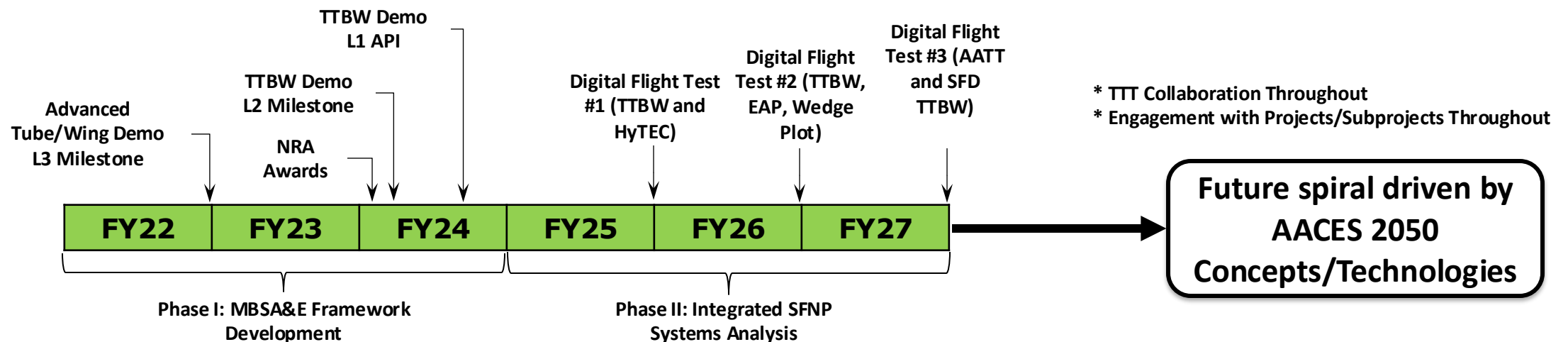


## Phase I: Develop common MBSA&E Framework/Ecosystem

- Create building blocks needed for key disciplinary analyses where lacking
- Integrate building blocks to form coupled MBSA&E framework – leveraging cross-project collaboration (e.g. TTT)
- Evaluate and test the MBSA&E framework with several use cases (conventional aircraft, TTBW/EAP)
- Explore and develop integration between MBSA/MDAO environment and MBSE tools

## Phase II: Cross-project integrated model development and coordination

- Development of open, common SFNP reference and vision vehicle concepts and models
- Regular, frequent tech interchange meetings across SFNP systems analysis teams, including external project partners
- Integrated systems analysis studies to incrementally ‘roll-up’ SFNP findings into a consolidated understanding of vision vehicle benefits and trades



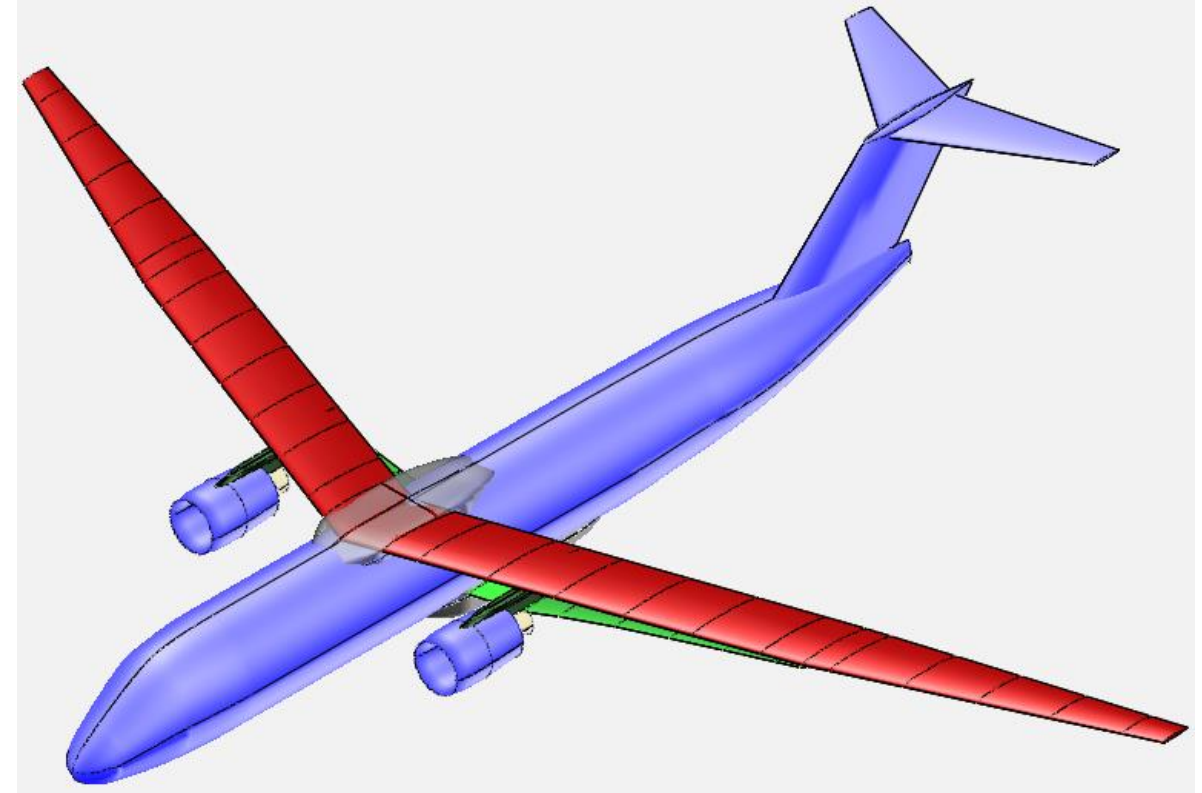
# Work based on SUGAR TTBW model



- The conventional aircraft design is based off the Boeing SUGAR Transonic Truss-Braced Wing (TTBW)
- This concept is a single-aisle transonic transport aircraft, similar to Boeing 737-400

## Design Specs

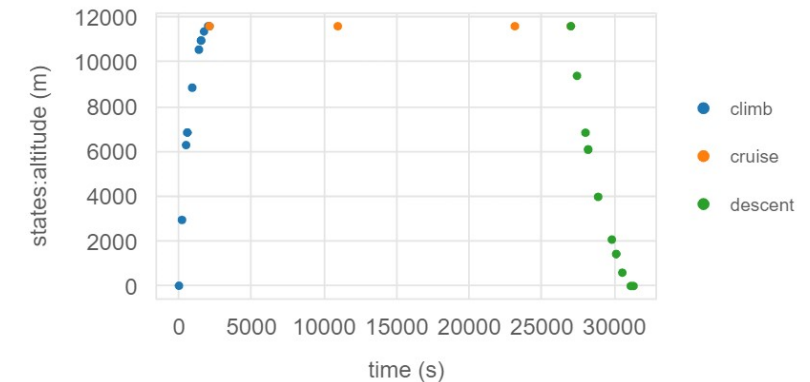
|                       |         |                 |
|-----------------------|---------|-----------------|
| <b>TOGW</b>           | 145 000 | lbs             |
| <b>OEW</b>            | 88 272  | lbs             |
| <b># of Pax</b>       | 154     |                 |
| <b>Max range</b>      | 3 205   | nm              |
| <b>Cruise Mach</b>    | 0.8     |                 |
| <b>Aspect Ratio</b>   | 19.7    |                 |
| <b>Reference Area</b> | 113.3   | ft <sup>2</sup> |
| <b>Span</b>           | 170     | ft              |
| <b>Sweep</b>          | 21.8    | deg             |



# Integration in Aviary: Sizing and Mission Analysis



- Initial development focused on a refactoring of the LEAPS code (Python FLOPS replacement) in OpenMDAO and Dymos to create a LEAPS2.0
  - Empirical weight correlation module
  - Low-fidelity aerodynamic module
  - Energy-height based equations of motion module
- Completed robust unit testing of LEAPS2.0 to FLOPS output for each module
- Integration with MBSA&E framework facilitated by common OpenMDAO base
- Recognized LEAPS2.0 was similar to the GASPy aircraft sizing and mission analysis tool under development at NASA (GASP replacement)



## Unified aircraft sizing and mission analysis tool

- Merges LEAPS2.0 and GASPy into a new tools with OpenMDAO and Dymos as the base
  - Includes empirical weight correlations and aerodynamic calculations from both original software tools
  - Includes unique equations of motion from each tool (energy-height and 2 DOF)
- Targeting open-source release for continued development and adoption

# This work uses Mphys: An OpenMDAO library for high-fidelity multi-physics



## Aero Solvers:

FUN3D  
CART3D  
ADFlow  
DAFoam  
VLM  
SU2

OpenAeroStruct

## Structural Solvers:

TACS  
Modal Solver  
NASTRAN

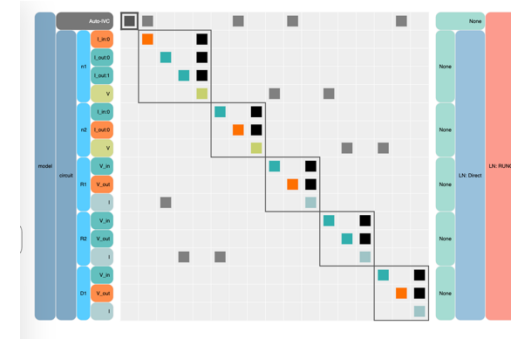
Load & Displacement Transfer:  
MELD  
Rigid Link Transfer

## Propulsion:

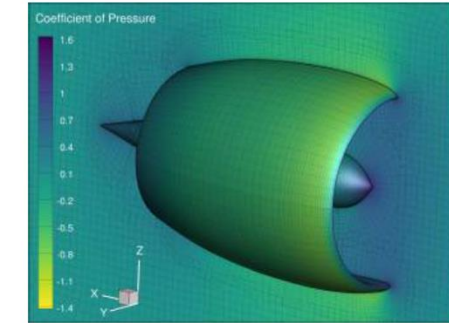
pyCycle

## Geometry:

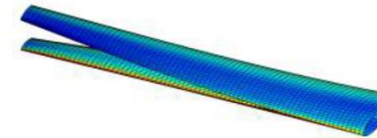
pyGeo (wraps OpenVSP, ESP, FFD)



OpenMDAO (NASA GRC)



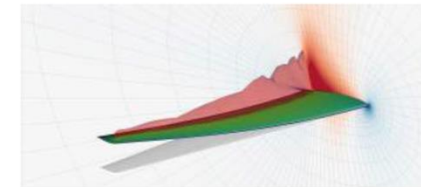
ADflow+pyCycle (Michigan)



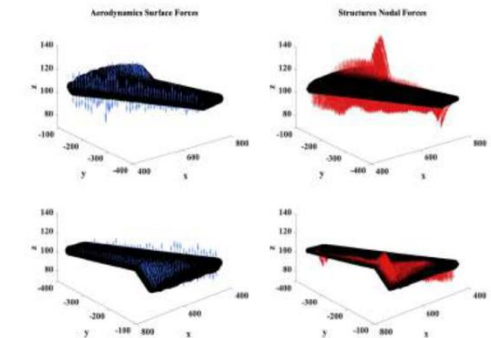
VLM+MELD+TACS



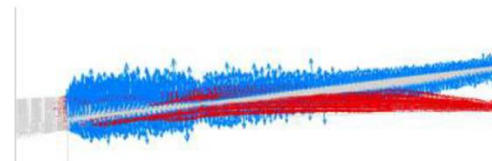
FUN3D+MELD+TACS



ADflow+RLT+TACS (Michigan)



SU2+MELD+NASTRAN (Stanford)



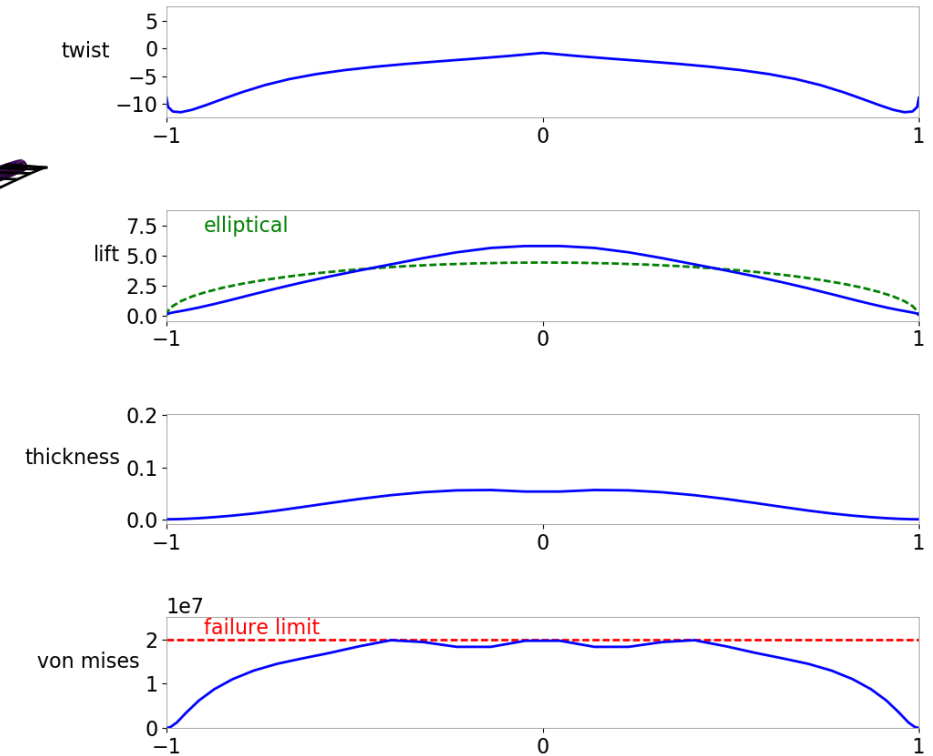
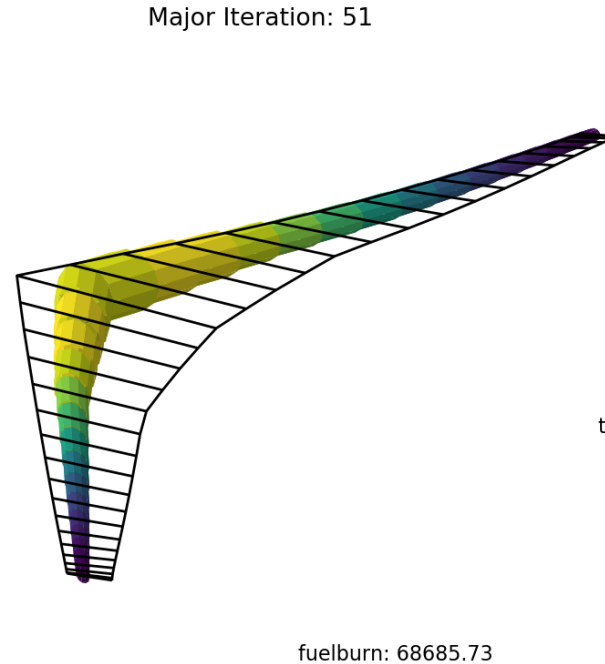
SU2+MELD+TACS (Stanford)

A. Yildirim, K. Jacobson, J. Anibal, B. Stanford, J. Gray, C. Mader, J. Martins, G. Kennedy. "MPhys: A Modular Multiphysics Library for Coupled Simulation and Adjoint Derivative Computation," Structural and Multidisciplinary Optimization, 2024.

# This work uses OpenAeroStruct for aerodynamic analysis



- OpenAeroStruct is a lightweight tool that performs aerostructural optimization using OpenMDAO.
- It couples a vortex-lattice method (VLM) and a 6 degrees of freedom 3-dimensional spatial beam model to simulate aerodynamic and structural analyses using lifting surfaces.
- These simulations are wrapped with an optimizer using NASA's OpenMDAO framework.
- For this work we utilize only the aerodynamic VLM solver (no aeroelasticity)

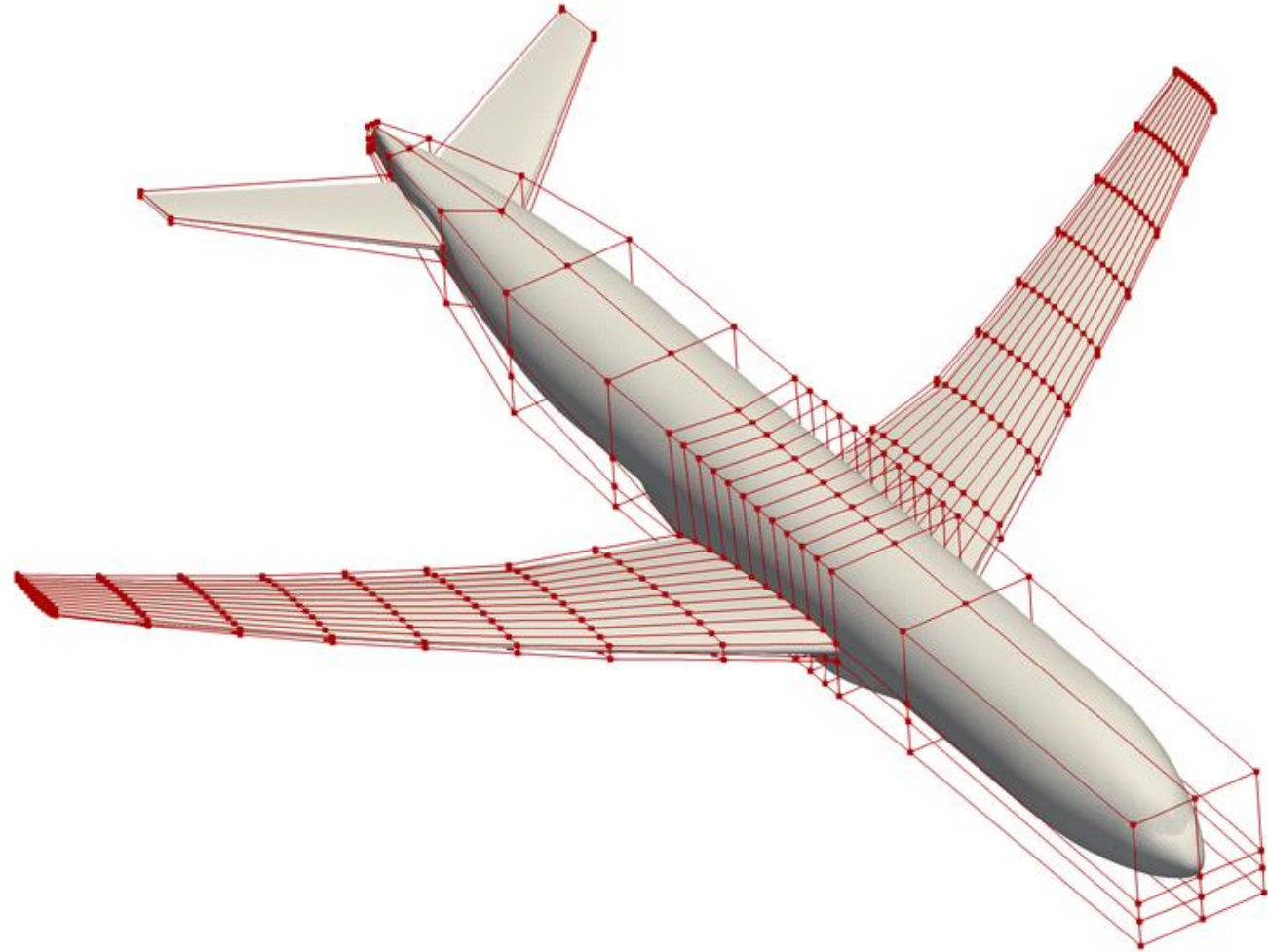


John P. Jasa, John T. Hwang, and Joaquim R. R. A. Martins. "Open-source coupled aerostructural optimization using Python." *Structural and Multidisciplinary Optimization* 57.4 (2018): 1815-1827. DOI: 10.1007/s00158-018-1912-8

# We use pyGeo as a geometry engine



- The pyGeo library is used to perform geometric manipulation on the aerodynamic mesh
- We utilize pyGeo's OpenVSP wrapper for this work
- Maps OpenVSP geometry variables (twist, span, sweep, etc.) to updated nodes on the VLM mesh
- Process is fully-differentiated allowing for geometric optimization
- The pyGeo library is used to wrap OpenVSP into Mphys/OpenMDAO

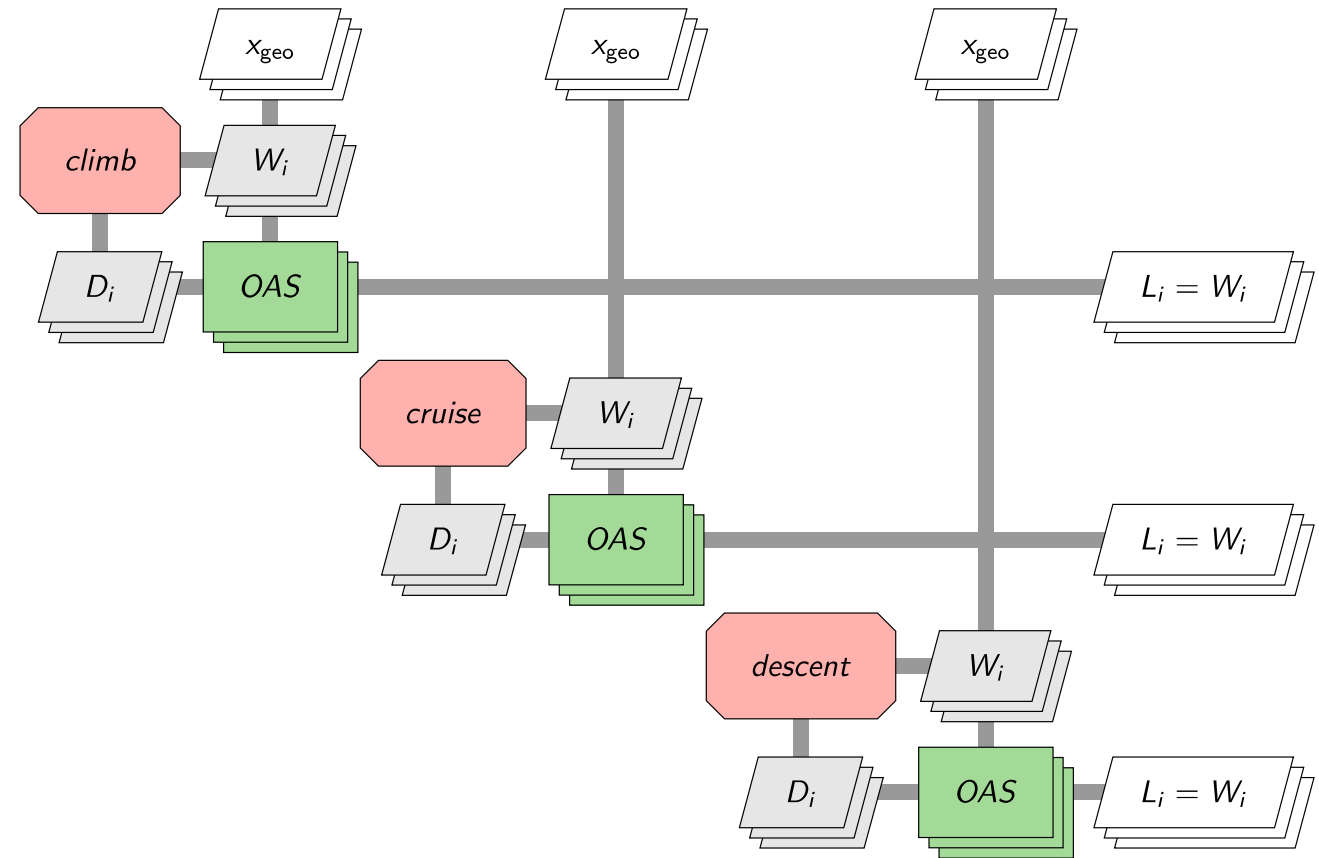


H. Hajdik, A. Yildirim, E. Wu, B. Brelje, S. Seraj, M. Mangano, J. Anibal, E. Jonsson, E. Adler, C. A. Mader, G. Kenway, and J. R. R. A. Martins. pyGeo: A geometry package for multidisciplinary design optimization. *Journal of Open Source Software*, 8(87), 5319, July 2023.

# Aviary – OpenAeroStruct Integration



- Multiple OpenAeroStruct instances are connected to each Aviary mission phase
- Each instance is responsible for computing the instantaneous drag for a given weight at each time integration point
- The OpenAeroStruct instances are coordinated using an OpenMDAO `ParallelGroup`, allowing for the aerodynamic calculation to be effectively parallelized



# Optimization Definition



Find:

$$\min_D \mathcal{F} = W_{\text{fuel}}$$

Where:

$$\mathbf{D} = \{W_{\text{gross}}, \bar{X}_{\text{traj}}, \alpha_{\text{traj}}, \theta_{\text{wing}}\}$$

Subject to:

$$\mathcal{R}(W_{\text{gross}}) = W_{\text{empty}} + W_{\text{fuel}} + W_{\text{payload}} - W_{\text{ramp}} = 0$$

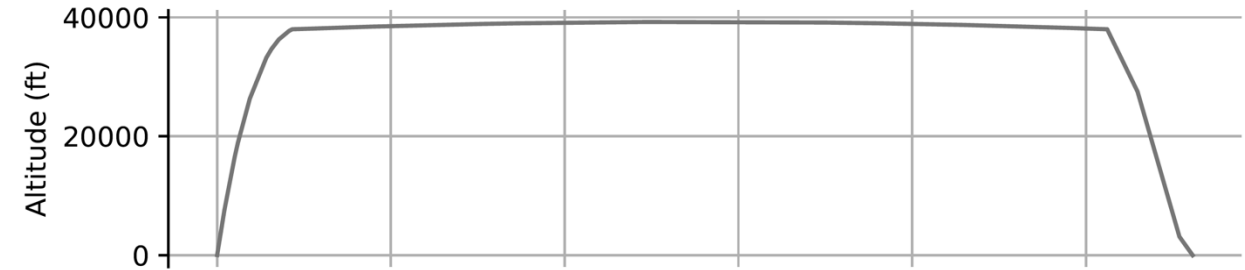
$$\bar{g}_{\text{min}} \leq \bar{g}_{\text{traj}} \leq \bar{g}_{\text{max}}$$

$$\bar{h}_{\text{traj}} = \bar{h}_{\text{eq}}$$

$$L_{\text{traj}} = W_{\text{traj}}$$

- Optimized with SNOPT
- 1e-5 feasibility constraint tolerance
- 1e-5 optimality tolerance
- 103 design variables
- 117 nonlinear constraints

**Trajectory design variables:**



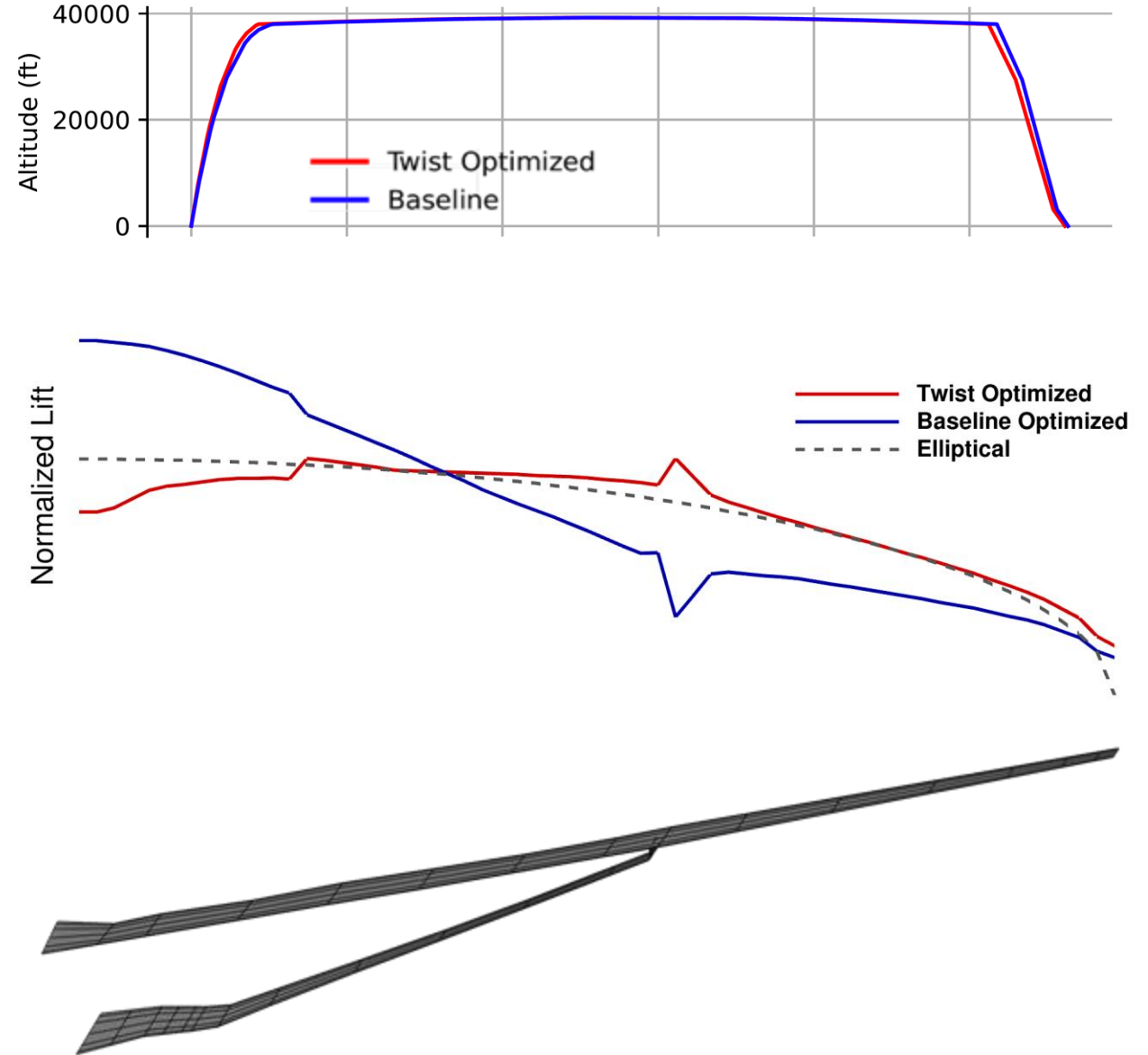
**Twist design variables:**



# Results



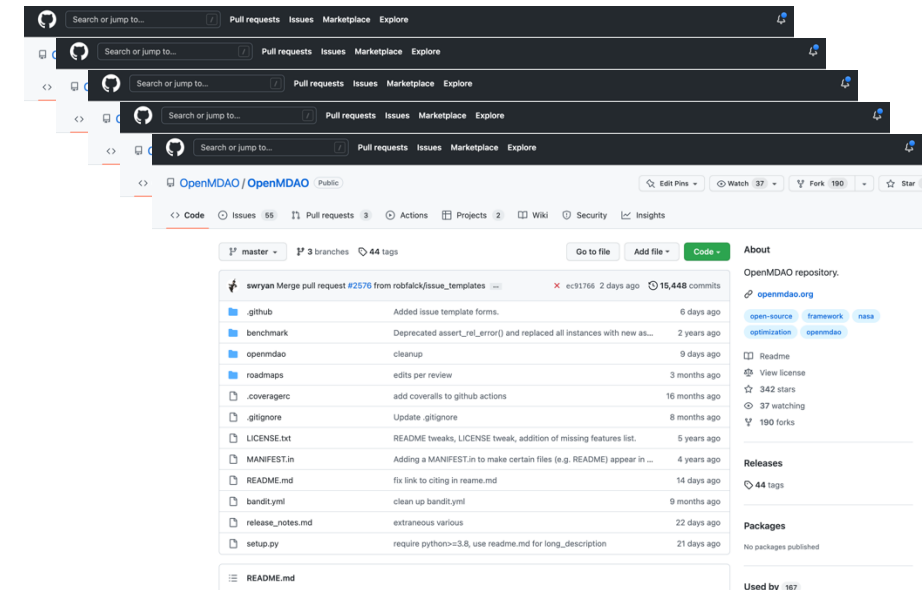
- Baseline design is an Aviary-optimized fixed geometry model (no twist variables)
- Twist optimized design achieves a more elliptical lift distribution to reduce induced drag and improve fuel performance
- Final wing geometry is mostly smooth with exception of small kink at wing root due to twist not being optimized there
- Final fuel burn savings of 8% relative to unoptimized baseline geometry



# Future work & Takeaways



- Development avenues for building on this work:
  - Control surfaces
  - Aerodynamic trim constraint
  - Aeroelastic flexibility considerations
  - Speed up OpenVSP geometry processing
- All the tools mentioned today are public and open-source, check them out!
  - Aviary: <https://github.com/OpenMDAO/Aviary>
  - Mphys: <https://github.com/OpenMDAO/mphys>
  - OpenAerostruct: <https://github.com/MDOLab/OpenAeroStruct>
  - OpenMDAO: <https://github.com/OpenMDAO/OpenMDAO>
  - OpenVSP: <https://github.com/OpenVSP/OpenVSP>
  - pyGeo: <https://github.com/MDOLab/pygeo>



# Thank you

---



This work was completed with support from the Systems Analysis and Integration group under the Advanced Air Transport Technology (AATT) project in the Advanced Air Vehicles Program.

