UNIVERSITY OF GEORGIA
College of Engineering

Lockheed Martin Aircraft Wing Design Optimization Capstone Journal

By: Randal Richards

Table of Contents

Table of Contents 1

PRE-CAPSTONE REPORT ACTIVITIES 2

FIRST DAY ACTIVITIES: 3

Team Introductions: 3

Smith Family Case Study: 4

ACTIVE LEARNING ACTIVITIES: 5

Design Process 6

W2 Active Learning Activity: 6

Milestones Layout: 8

DEFINING STAKEHOLDERS ACTIVITY: 9

TEAM MEMBERS WEEK 3 ACTIVITY CONNECT WITH PROJET TEAM: 10

W3 Define your Project 13

MILESTONE DISCUSSION: 13

CAPSTONE: 16

First Client Meeting: 16

Microsoft Teams Call 09/15/2025 @ 6:00 PM EST : 17

Introductions 17

Definition 18

Sources to Study from Meeting 18

Fidelity Information 19

Some Project Objectives and Requirements 19

Develop Wing Design Optimization Process 19

Q&A Section of Meeting 20

Summary of Meeting 21

Team Meeting for M2 23

Summary: 24

Project Background 24

Industry Fundamentals 24

Industry Background 25

Project Purpose / Problem Definition 25
Projected Project Impact 25
Goal Statement 26
Meeting #2 w/ Dr. Banu (Supervisor) 26
Pre-Meeting 26
Meeting Notes 27
In Class in SLC: 27
In Class EELD Activity: 29
Summary: 29
Bi-Weekly Meeting 30
Pre-Meeting: 30
My Meeting Notes: 30
Summary: 31
Milestone #3 Stakeholder Mapping and Identification 31
Project Stakeholders and Needs: 32
Stakeholders 32
Mr. Seay (Primary) 32
Bob Clark (Primary) 32
Paul McClure (Primary) 33
Wind Tunnel Team (Secondary) 33
Madison Warner (Tertiary) 34
Previous Lockheed Capstone Group (Tertiary) 35
Stakeholder description: 35
Interview: 36
Interpretation: 36
Summary of Milestone: 38
Meeting with Paul McClure 39
Meeting Summary — Aerodynamic Workflow & Early-Phase Design Guidance: 39
Questions from Milestone #3 40
Meeting with Bob Clark 41
Questions from Milestone #3 42

Milestone #4 Project Understanding: 43
Contributions: 44
Wing Research 45
XQ-58A Wing Optimization — Working Notes (Low-Fidelity Phase) 45
Bi-Weekly Client Meeting Report 47
Client Meeting Summary — Lockheed Martin Sponsor 47
Research: Reymer's Equations and Tables 49
MATLAB WORK 52
Bi-Weekly Meeting 53
Research: MATLAB Code and Work 54
Old Code: 54
Supervisor Meeting 55
Ideation 55
Research: MATLAB Discrete Range Equation 60
New Code: 60
Bi-Weekly Client Meeting 63
SWOT ANALYSIS 65
Research: Troubleshooting 66
Bi-Weekly Supervisor Meeting: 69
Milestone #6: Project Presentation and Tables 70
Milestone #7: Presentation Work 71
Research for MATLAB: Finalization 71
Process & Research: 71
Next Steps: 71
Conclusion: 71
Capstone Meeting w/ Client 71
Capstone Meeting w/Client 71
MATLAB Code Work 71
Next Steps: 71
MATLAB Code Work 71
1. Project Overview 71

2. MATLAB Codebase — What Was Built and Fixed 71
2.1 Files Revised 71
2.2 Optimizer Fixes (MainDriver.m + objective_constraints.m) 71
2.3 runWingSimulation.m Fixes 71
2.4 missionRangeJet.m Fixes 71
2.5 loadVSPPolar.m and getCDfromCL.m 71
2.6 getIntegratedProps.m 71
2.7 saveGAHistory.m 71
2.8 validateBaseline.m (New File) 71
3. Edwin's MATLAB Files Built and Fixed 71
3.1 X58_Optimization_Driver.m 71
3.2 Master_Optimization_Suite.m 71
3.3 Plot_Wing_Overlays.m 71
4. Python AeroSandbox Notebook — Assessment 71
4.1 What the Python Code Does (That MATLAB Does Not) 71
4.2 What MATLAB Does That Python Does Not 71
4.3 Critical Issues Found in the Python Notebook 71
4.4 Team Decision: Consolidate to Python 71
5. Airfoil Optimization Results 71
5.1 Multi-Point Transonic Airfoil Optimizer Output 71
6. Why the Numbers Differ Between MATLAB and Python 71
7. Path Forward — Goals and Recommended Next Steps 71
7.1 Critical Fixes Before Next Run (Python) 71
7.2 Parameter Dashboard 71
7.3 Adding Control Surfaces 71
7.4 Multi-Aircraft Capability 71
7.5 Dropping Fuel Considerations 71
7.6 Recommended Implementation Order 71
8. Key Constants and Reference Values 71
8.1 XQ-58A Valkyrie Baseline (MATLAB model) 71
8.2 Python Notebook Configuration (current) 71

Capstone Showcase: 71
Team 1: SecureWick 71
Team 2: NOVA 71
Team 3: NutraSky Manufacturing 71
Part 3: Personal Reflection 71
AI DISCLAIMER: 71

PRE-CAPSTONE REPORT ACTIVITIES

FIRST DAY ACTIVITIES:

08/19/2025

Team Introductions:

GOAL – Get a group of 4 together and list all your EELD Shapes, Majors, Fun Facts, and goals after getting your degree.

Randal Richards

- Z
- Mechanical Engineering
- Hold two records for HS
- Wants to get government contract for Engineering.

Daniel Nelson

- Circle
- Mechanical Engineering
- One of 12 children
- Wants to be MEP.

Quinn Rodier

- Triangle
- Mechanical Engineering

- Rock climber for a few years
- Wants to go into Engineering Sales

Evan Sagatovski

- Square
- Mechanical Engineering
- In the US army
- Want to go work for a naval or aerospace company.

Smith Family Case Study:

GOAL – Provide an ample description of the problem that the Smith Family faces and begin to make a solution with listed steps.

MY APPROACH:

Start with the quickest, cheapest wins. Seal the home's shell and shore up attic insulation to cut drafts and keep conditioned air where it belongs. Next, install a smart thermostat, inefficient cycling and poor scheduling are quietly drive the utility bill through the roof, and automation keeps comfort steady without issues. For windows, you don't have to go straight to full replacements: if the frames are in good shape, add interior inserts or storms; reserve double-pane replacements for the worst offenders. Swap any remaining bulbs for LEDs, they use far less energy and last much longer. Finally, when the current system is nearing the end of its life, plan a right-sized, well-commissioned HVAC upgrade. That's when you lock in the biggest comfort gains and the most reliable savings during extreme weather.

TEAM APPROACH:

The Smith family should begin with the most cost-effective and accessible solutions. First, they should keep the thermostat at a constant temperature. This prevents the HVAC system from working harder to adjust to sudden changes, reducing unnecessary energy use and costs. Swapping older bulbs for LED lighting would also provide quick savings, as LEDs use less energy and last significantly longer. The family should also consider upgrading to smart thermostats and more efficient AC units. These systems would allow for better control and optimization of heating and cooling, ultimately lowering utility bills. Finally, the best but most expensive solution would be to install new insulation and seal the home properly. Improved insulation minimizes energy loss, reduces heating and cooling demands, and provides long-term savings. By prioritizing these steps, the Smith family can balance affordability with efficiency. Starting with smaller changes provides immediate savings, while long-term investments like insulation and upgraded HVAC systems maximize comfort and energy efficiency in the years to come.

Randal Richards

REFLECTION:

My approach led me to the first thing about the most cost effective and cheapest wins. To connect truly with a client I believe it's best to come at the problem in the simplest, yet, most effective plans for the home. I believe that speaking in layman's terms as well as adding context to the certain solutions was one of the best approaches to the problem.

DIFFERENCES IN INDIVIDUAL APPROACHES:

The different approaches were sometimes more orderly and word dense than others. I think these greatly highlight each team members strengths and uniqueness and add to the quality of a collaborative effort to solve the problem.

DECISION MAKING CONSENSUS:

The final approach was to start with the cheapest and most efficient solutions. Incorporating LED lights and replacing the AC system. Sealing the home and providing extra insulation was going to be the most costly and require the most effort. Using all the leadership strategies was a must and putting them at the bottom was something that I had to concede to make the solution viable.

CHALLENGES AND OPPORTUNITIES:

Although there arent many differences in the decision making for such a simple problem, there were issues between leadership styles on what the most effective method would be. The positive aspects were a plus though, granting new perspective on what should change and what would be the best for the projects.

TAKEAWAYS AND FUTURE APPLICATION:

I prefer a comprehensive solution, adding the pros and cons as I write and incorporating them with the stake of the home owners and their assets. Provided that there was no budget or timeline I still granted both in my own solution. I can leverage this style more in the real world as there are better applications with projected spending and timelines as these are highly valued skills in the engineering world.

Randal Richards

ACTIVE LEARNING ACTIVITIES:

08/26/2025

Design Process

- The design process is a systematic method for problem-solving.
  - Steps in the Design Process
    1. Define
    2. Formulate Solutions
    3. Develop Models and prototypes
    4. Present and Implement
- Defining is the most important step because if you have the wrong problem, it is impossible to create the right solution for the right problem.

CDIO- Conceive, Design, Implement, Operate

W2 Active Learning Activity:

GOAL – Covering the design process and discussing how Capstone Design projects track the design process. Afterwards, practice client interviews - an important part of the "Project Definition" step in the design process.

Round 1: Review the case study, “Managing Energy Efficiency in the Home,” and conduct client interviews (15 min):

- Acting as the Smiths:

Questions the design team asked acting as the SMITHS:

What kind of HVAC system is installed already?

A: Window AC, Natural Gas Furnace, and Fire Box.

Where have you noticed the windy drafts coming from?

A: Near doors and Windows

How often do you turn your lights off when you leave the room?

A: N/A

Do you have any lights that are currently LED?

A: N/A or 0

Is your thermostat changing throughout the day or on one temperature?

A: Static A/C ; no change

What is the highest utility bill that you've had?
A: Looking for 30% reduction in bills

What is the timeline?

A: Family must be able to reside in residence during the renovation period; No timeline given.

Have you had any inspections already?

A: N/A

What is the budget for the renovation?

A: $25,000Current Carbon Footprint?

A: N/A; Wants Lower

When was the latest renovation?
A: 1987.
Round 2: Define the project (5 min):

- Identify any questions/ requirements that the design team did not ask:
    - They did not ask how many occupants are in the home nor did they ask the style/square footage of the home. They did not ask the materials sed in the home and how many floors there were. They didn't ask the temperature or energy supplies in the home. They also didn't ask for the other amenities that the family would like in the home.

- Given the questions asked, will the design team succeed?
    - The team will succeed at a slower rate than desired. Most likely the family will have an issue with the design process as it does not include the specifics of the house or incorporate temperatures, climate, and location for their materials and equipment being used.

Randal Richards

- Define the project from your perspective.
  - The Smith family wants to renovate the home to meet clean energy standards, reduce utility cost, and provide comfort for the residents.

- Propose your own solution:
  - The Smith family will benefit from a comprehensive renovation focused on energy efficiency and comfort. This includes replacing the existing HVAC and heating system with high-efficiency models, installing underfloor heating in the bathroom, and adding a cold-climate heat pump. A multi-zone smart thermostat should also be incorporated to optimize temperature control throughout the home, along with an on-demand hot water system and a functional fireplace insert for both efficiency and comfort. Finally, sealing drafts and addressing insulation issues will be essential to ensure long-term energy savings and improved indoor climate control.

Round 3: Redefine the project and determine solution feasibility from AL #1 options (10 min). As a full team:

- Does the Design team's solution meet the Smiths' requirements?
  - The design team's solution meets the Smiths' requirements only partially. Overall, it does address the need to upgrade the system and make the home more energy efficient. However, some of the specific amenities requested, such as the cold-water pump for the hot water system, were not included.
- What info was missed during Round 1?
  - Smith's special amenities were missed during the questions.
  - Smith's specifications from the house were also missed.

Class Discussion (5 min):

- Solar panels wouldn't work because of the steepness of the roof. Also wouldn't work because its in Minneapolis and its in high altitude. Also have to think about the average temperature because of the long winters.

Milestones Layout:

- **Milestones in Capstone Design**

1. Edusourced files
2. Design Notebooks
3. Edusourced Milestones
   a. Updated Cover Sheet
   b. Presentation slide deck addition
   c. Capstone Final report additions

Randal Richards

4. Time Tracking
    a. Log Hours
        i. Use appropriate Time tracking
        ii. Include notes/descriptions
- **Example: Milestone 1: Client Meeting/Site Visit**

→Create Files

A. Fall 2025 Milestone Cover sheet
B. Client Meetings
C. Final Report
D. Final Presentation

Folder Names ALWAYS: FirstName_LastName_Fall2025_Notebook

DEFINING STAKEHOLDERS ACTIVITY:

09.02.2025

GOAL – Define the stakeholders for multiple projects and ideally continue to strengthen those skills for use later in the semester. Individuals are tasked with creating a list of components that identify the

- What is a stakeholder?
    - "Anyone contributing to the overall project" / "The person/s financing the project"
    - "Any group or individual who can affect or is affected by the achievement of the organization's objectives."
- In summary a stakeholder is:
    - Actively involved or their interests may be positively or negatively be affected by the project

- Meaningful Stakeholder Input's effectiveness tapers off after the beginning because the main problem is the project identification and the general planning issues. Identifying the *scope* of the project will make you more effective with your time and energy.
- These steps include:
    - Identify
    - Prioritize
    - Understand
    - Manage

→Completing 1-3 will help you *map* your stakeholders
→After mapping stakeholders you can better manage them through an engagement plan

- Example Problem to Identify Stakeholders

Project Focus: Spray Quality Monitor for Informed Pesticide Applications
Project Sponsor: UGA Tifton
Project Description: New Spray Moderator Control for Pesticides

- Identify Stakeholders:
    - Pesticide Groups, Independent Pest Research, Neighborhoods, GA Tifton, Farmers, Grocery Buyers, Shipment Facilities and Management, Pesticide Manufacturers, World Health Leaders, Water Filtration Companies, Engineers on Design Team. Stock Owners, News Outlet, Farmer Equipment Commercial Selling. Food Safety
- Some Easy Categories to list Stakeholders from:
    - Client, Funder, Supplier, Competitor, Regulator, Instructor, Community…etc

For Industrial Aircraft Wing Design Optimization:

- Identify Stakeholders:
    - **Primary :**
        - Sponsor (Lockheed Martin), Instructor, Design Group, Pilots
    - **Secondary:**
        - Plane Manufacturers & Engineers, Aerospace Engineers, Birds, Wing Design Optimization, Inspectors, Regulators
    - **Tertiary**:
        - Communities, Aircraft Terminal Operators, Airport Operators, Flight Command, Airforce Pilots, Airforce Commanders, Media, Industry Leaders,

Stakeholder Analysis:
Engage with stakeholders early to understand their expected involvement in, and outlook on, the project.

- More Personal the more helpful
- Reach out with Surveys and Emails
- Phone calls or Video meetings are the best way to make a connection
- Identify interpersonal preferences
- Start Building Relationships with Stakeholders
- Adaptability is really important as well.

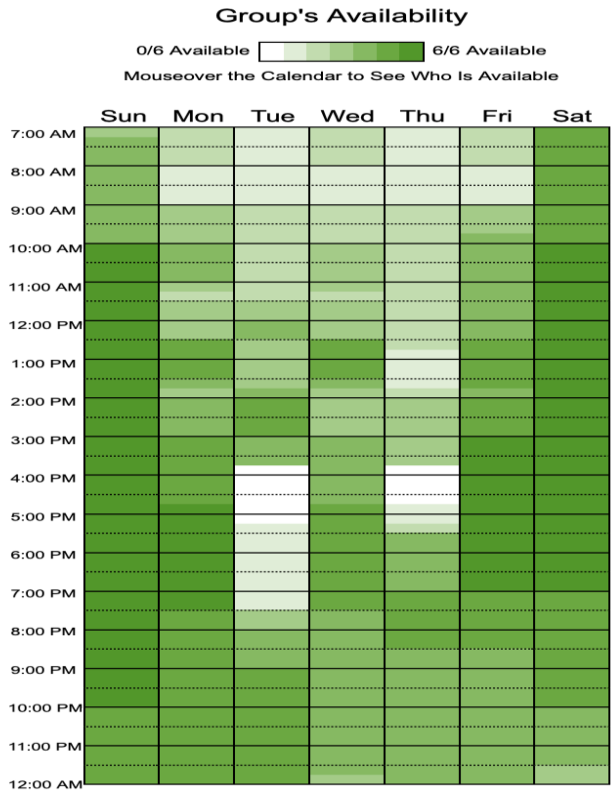
TEAM MEMBERS WEEK 3 ACTIVITY CONNECT WITH PROJET TEAM:

Capstone Group info:

- Cooper Mendlinger
  - cam03006@uga.edu
  - 678-517-8649
  - Square
  - Meche / Aero cert
  - Design, GD&T, Aircraft Structural Design, Optimization/pd techniques
  - Gulfstream Co-op 2 rotations
- Garrett Kuhn
  - gmk55337@uga.edu
  - 301-887-7331
  - Z
  - Mechanical Engineering / Aerospace Certificate
  - Relevant Coursework: Fluids, Thermodynamics, Circuits,
  - Relevant experience: Robotics, Some autoCAD
  - Post Graduation: Grad school for aerospace engineering
- Martin Vassilev
  - msv37717@uga.edu
  - 678-210-9977
  - Triangle
  - Mechanical Engineering / Aerospace Certificate
  - Strengths: Fluid Mechanics, FEA, Rocketry, Mechanical Design
  - Relevant Experience: Robotics Internship, Robotics REU, Level 2 Rocketry Certification
  - Post Graduation: Get my masters in Aerospace Engineering
- Griffin Cantrell
  - gtc21014@uga.edu
  - 770-670-9228
  - Mechanical Engineering

    - Circle
    - Private Pilot, Aviation Club, AIAA Rocketry
    - Normal course work
    - Ideally I'd like to work for an aerospace or defense company

- Randal Richards
    - rjr23574@uga.edu
    - 404-984-7557
    - Mechanical Engineering / Z / Math Strengths, Leadership Strengths, Communication Strengths
    - Experience: Leadership Experience in St. Thomas for HKT Energy Group. Government work Instillation of High Voltage Cable in Residential and Commercial Areas. Solar panel and Transformer Design and Installation.
    - Post Graduation: Travel the world
    - Coursework: Normal Coursework
- Edwin Harper
    - jeh71407@uga.edu
    - 678-522-1018
    - Mechanical Engineering
    - Leadership Style: Circle
    - Relevant Coursework: Advanced Thermal Fluids, Energy Systems
    - Experience: Internship at Precision Manufacturing Plant, Leadership, Machining, Welding, Mechanical Assembly, CAD (Solid Works and Inventor)
- Group Availability:



W3 Define your Project

- Client: Lockheed Martin
- Instructor: David Krupp, Wayne Johnson
- Supervisor: Eliza Banu

For Industrial Aircraft Wing Design Optimization;

- Identify Stakeholders:
  - **Primary :** Sponsor (Lockheed Martin), Instructor, Design Group, Design Engineers at Lockheed Martin / Employees of Lockheed Martin, Client
  - **Secondary:** Plane Manufacturers & Engineers, Aerospace Engineers, Birds, Wing Design Optimization, Inspectors, Regulators
  - **Tertiary**: Communities, Aircraft Terminal Operators, Airport Operators, Flight Command, Airforce Pilots, Airforce Commanders, Media, Industry Leaders,
- Goals:
  - Optimize Aircraft Design for speed and efficient flight. The project introduces students to CAD Modeling and the rigorous process of wing testing. At the end, have a complete idea of workflow for aircraft design and CAD capabilities.
- Objectives & Deliverables:

Randal Richards

  - Group Members will use a range of engineering tools to carry out aerodynamic wing design and optimization. CAD software will be used to define wing geometries and generate parametric models. Lower-fidelity aerodynamic analysis tools will support early-stage airfoil and wing performance estimates, while higher-fidelity CFD will be used for detailed aerodynamic evaluation and refinement during optimization
- Deliverables will include a final report and presentation summarizing the workflow, optimized design, and lessons learned

MILESTONE DISCUSSION:

09/09/2025

Lecture Topics:

- Notebooks and Time Tracking
  - Stay away from regular emails
  - Keep communications in Edusourced!!!
- Design Notebooks:
  - Denotes Personal Contributions to work
  - Whats inside the notebook
    - Notes from the conversations meeting personal musings on the topics
    - Keep all entries sequential
    - strikethrough for mistakes;
    - Avoid blank gaps on pages
    - Append-only edits: never delete stuff just add a correction with initials + date/time
    - Timestamp and sign each entry
    - Witnessing: teammate comments

        - Table of contents: use headings or an index page linking entries
        - Back up digital file to secure storage
        - Sketch using stylus or scan in paper copy!
    - In Word, add a date by “Shift + Alt + D” and time by “Shift + Alt + T”
    - Progress checks every 2 weeks
    - Must compile into a single PDF File
    - Submit new notebook pages every two weeks
        - MMDDYY_Name_Notebook.pdf in Edusourced under Files/Notebooks
        - Notebooks not named with this will receive a big fat **ZERO!**

- Time Tracking:
    - Allows time tracking of individual and team contributions to the team.
    - Can help identify bottlenecks and inefficiencies
        - **Keeps teams honest**
        - **APART OF GRADE!!!**
    - There are multiple groups that time tracking may come into play
- Team Structure
    - Internal:
        - Project Team
        - Instructor
            - Meets weekly
            - Provides instruction
            - Administers content on the design process, guest speakers, all the lot
        - Supervisor/Mentor
            - Meets Biweekly
            - Connects to resources
            - Monitor Progress
    - External
        - Project Sponsor/Client
            - Informs needs
            - Approval/Disapproval
            - Funding
            - Engages with PS/Instructor
- Project Management
    - Initiation
    - Planning
    - Execution

  - Monitoring and Control Completion
- Scheduling Meetings
  - Lockheed Martin will take some time to have an in person meeting
  - Schedule a virtual or zoom meeting instead so we can initiate planning and continue working towards milestones
  - Travel expense to Lockheed Martin Mareitta:
  - ~76 miles and a reimbursement of $.67 a mile comes around $50.69 for one way trip.
- MCHE 2990 Throwback
  - Psychological Safety
  - Dependability
  - Structure and Clarity
  - Meaning
  - Impact

Active Learning-Team Breakout (15 min):

- Talked with team about meeting agenda and questions. Created questions for the meeting ahead of schedule and continued to draft agenda to clarify any points we may have missed before the meeting begins. Scheduled a Teams Call on Monday the 15th @ 6:00 PM EST to get all of the project details from sponsor Matthew Seay.

CAPSTONE:

First Client Meeting:

Microsoft Teams Call 09/15/2025 @ 6:00 PM EST :

Figure 1: Meeting with all group members for Capstone Design with Martin and Dr. Banu present.

In attendance:

Lockheed Martin – Matthew Seay

University of Georgia – Randal Richards, Dr. Eliza Banu, Martin Vassilev, Edwin Harper, Griffin Cantrell, Cooper Mendlinger, Garrett Kuhn.

Introductions

Senior Aeronautical Engineer

- F-22's and F-35s were manufactured in plant that Matt works at
- Marietta is where the C-130 and cargo planes are manufactured

Randal Richards

- Matt lived in Alabama and went to Auburn and started working for Lockheed Martin in 2020 (Roll tide!)
- He's a Senior Engineer working with skunk and also the Lockheed Martin Capstone team for the second time this year
- Really excited to talk to fellow engineers about the project and asses our levels of research and computation throughout the process.
- UGA Students gave introductions with Majors, specific qualifications, and interests.

Definition

- GOAL
    - Design and optimize an unmanned ACP aircraft wing using a CAD-driven, digitalized, pipeline with low and high-fidelity aerodynamics, delivering a repeatable optimization process and a final optimized wing through workflow analysis.

Sources to Study from Meeting

- We will be provided accepted models & workflow charts from Lockheed (if approved by legal team and shared)

If not provided with these workflow charts and accepted models from Lockheed, there are sources online that provide great documentation of the aircraft wing design process with open source code that we can use to amplify our understanding.

These will include:

- NASA:
    - Open-source Aeronautics catalog (software.nasa.gov, Aeronautics category). https://software.nasa.gov/software/category/aeronautics
    - OpenMDAO for MDAO workflows.
    - FUN3D (subject to availability & license/approval).
    - MDO/MDAO ecosystem (open/academic): https://mdolab.engin.umich.edu/software
    - MDOLab (UMich): MACH-Aero stack (pyGeo for geometry, pyOptSparse, MACH, OpenAeroStruct).
    - SU2 (finite-volume CFD, adjoints; open-source).
    - AVL / VSPAero (vortex lattice), XFOIL (2D airfoil), OpenVSP (parametric geometry).
    - OpenFOAM (CFD; if needed for custom physics/meshing).
- CAD & Pre/Peri/Post:

  - SolidWorks / CATIA / Fusion 360 / Onshape (select one), OpenVSP for rapid parametrics.
  - Pointwise/Gmsh/snappyHexMesh for meshing (depending on solver stack).
- Compute:
  - University Supercomputer access (with Dr. Banu's authorization) for high-fidelity CFD/adjoint runs.
- Resources from these sources can be productive for research and modeling

Fidelity Information

A lot of the design process is in two phases:

- Lower-Order (fast exploration):
  - Methods: Vortex Lattice (AVL/VSPAero)
  - Use: Large design-variable sweeps, planform and airfoil family screening, quick sensitivity.
- Higher-Order (refinement & validation):
  - Methods: RANS CFD (SU2 / FUN3D / OpenFOAM)
  - Use: Validate short list from lower-order results, refine with mesh-converged solutions.
- We could also use the University Super Computer with authorization from Dr. Banu

Some Project Objectives and Requirements

- Primary aero objective: Minimize drag coefficient (CD) at a given 1g cruise CL.
- Secondary: Reduce structural weight proxies while meeting lift/mission constraints (span/area limits, cruise Mach/Re, max thickness for structure).
- Aero-structural coupling:
  - Start aerodynamic-only; add a lightweight structural model (e.g., wingbox bending stiffness, thickness-to-chord constraints) for twist/buckling margins if time permits.

Develop Wing Design Optimization Process

- APPROACH

- Two-phase aero analysis (lower-order → higher-order) in a loop, with parametric geometry, automated meshing (no morphing of detailed high-order meshes when avoidable), and scalable compute.

Boundary: Control surfaces out of scope for this semester (possible follow-on). Focus on straight, cruise-dominant flight conditions

- First Semester Objectives
  - Survey:
    - Consolidate key concepts in wing design, aerodynamic fundamentals, and modern optimization methods.
  - Select and master the tools:
    - Identify readily available tools, down-select to a coherent stack, and work through representative example problems to ensure proficiency.
  - Secure compute and CFD access:
    - Define the path to CFD software (licenses, installs, validation cases) and line up computing resources and job workflows.
  - Deliver an initial process baseline:
    - Present a first-pass wing optimization workflow (inputs → geometry → analysis → optimization → reporting) with a demo case and documented assumptions.
- Second Semester Objectives
  - Refine and harden the pipeline using first-semester findings
    - Tighten meshing/CFD settings, validate surrogates, and lock in repeatable, end-to-end workflows.
  - Broaden scope to new configurations
    - Including transport-class planforms and flight conditions from low-speed/high-lift to transonic.
  - Tackle higher-complexity optimization cases
    - Applying the process to multi-disciplinary tradeoffs and documenting performance gains and limits.

Q&A Section of Meeting

- Q: What are some performance metrics that will be required?:
  - Structural designers will provide requirements; aerodynamics focus on maximizing L/D at 1g cruise, reduced drag at mid-mission cruise weight, and given angle of attack.
  - Thin airfoil theory is primary for fighters.
  - Camber and twist are critical parameters.
  - Example reference: 64A airfoils.
    - Question for Engineers: How do I reduce drag reduce weight and cruise at the 1g CL rate Optimize CD at a given CL

Randal Richards

- Q: When will be given key data on aircraft specs?:
    - Likely around mid semester check in after initial research and tool discovery phase
- Q: What are some references suggested?:
    - *Fundamentals of Aircraft and Airship Design* – Nikolai.
    - *Aircraft Design: A Conceptual Approach* – Ramer.
    - F-18 fuselage weight estimation equation mentioned for first-level weight assessments.
- Q: Best form of communication?:
    - Email is a good contact method.
    - Biweekly meetings suggested instead of monthly.
    - Lockheed makes competitive assessments —some references may be usable.
- Q: Site Visit status?:
    - Uncertain at this time, reaching out to contact at Marietta site.
- Q: Will we be making a physical design of the aircraft wing that we create?
    - Digital Design all the way. No way we can build anything other than at a DBF scale. We can make a 3D printed pin of the wing similar to the last time this capstone was available. Low speed tunnels could be used for optimization techniques and journaled for other use.
    - Previous Capstone chopped wing off in CAD and drew in their own wing with parametric modeling. Want you guys to do the same thing almost.
    - Can provide weight parameters and what the cruise numbers should be. Two textbooks that give the weights of aircrafts and how they are created and optimize what the 1g weight should be. The models you can pull from Grab CAD.
- Q: How do you want us to share our progress throughout the project?
    - Last time we did a monthly meeting, Biweekly is a fine cadence and we can also do months. Emails are great and can respond later in the day.
- Q: When would you share the parameters with us?
    - Not sure exactly what the progress for you guys is right now, but after we lock down a date we can piece together the workflow and mature it through problems and tasks. Midpoint meeting with a presentation and see the quality.
- Do we consider Control Surfaces?
    - No, just prioritize standard straight cruise lines. Maybe next semester with control surfaces on the loop.

Summary of Meeting

- We reviewed the plan to design and optimize a wing with CAD, low- and high-fidelity aero tools, and a clean digital workflow. If Lockheed shares models and charts, we'll use them; if not, we'll lean on NASA and open-source tools. Work runs in two phases: quick low-order screens, then higher-order CFD on the top picks, with cluster time if approved by Dr. Banu. The key goal is lower drag at a set 1g CL, with simple shape rules and no control surfaces this term. We'll pick a tool stack, lock down compute access, and show an initial process and demo case. Meetings are biweekly (monthly recap ok); site tour is not set; flight and weight inputs will be shared later.

9/16/2025

Lecture:

1. Milestone #2 Confirmation of Scope of Work 09/19/25
2. Milestone #3 Customer Discovery 10/03/25
    - Project Needs and Appendices are updated
        i. List or Table of Stakeholders with affiliation
3. Milestone #4 Project Understanding 10/14/25

-Identify who is the customer and who is the user. Your product specifications should not restrain the user or the customer in any way. Accommodate features after you establish them. We want to fully quantify and qualify the customer's needs and define what requirements you are meant to complete.

Design Specifications must be SMART:

- SMART
    i. Specific
    ii. Measurable
    iii. Achievable
    iv. Realistic
    v. Time Bound
- Concept Development Process
    i. Identify
    ii. Establish
    iii. Generate
    iv. Select Product Concept
    v. Test
    vi. Final Specifications
    vii. Downstream Development
- Product Requirements
    i. Customer Needs

ii. Product Specifications
iii. Establish Specifications

-Identify Codes, Standards, and Regulations. There may be relevant standards at how you measure specifications. So you need to find applicable codes and variations within those codes and adhere to them, desperately.

4. AL Exercise
    - Activity #1 Identify Stakeholders and Develop Engineering Requirements (10 min)
        i. Stakeholder List
            1. Primary Stakeholders
                a. Matt Seay
                b. Dr. Banu
                c. Dr. Johnson
                d. UGA
                e. Capstone Team
            2. Secondary Stakeholders
                a. Lockheed Martin Paralegals
                b. Lockheed Martin Engineers
                c. Inspectors
                d. Maintenance
                e. Software Engineers NASA
            3. Tertiary
                a. Birds
                b. Pilots
                c. Emission Groups
                d. Taxpayers
        ii. Stakeholder Interview (UGA)
            1. How does our optimization of aircraft wings affect your decisions when supporting carbon emissions groups?
            2. What are the project deliverables that you want us to output to continue our collaboration with Lockheed Martin

Team Meeting for M2

09/18/2025

Randal Richards

Summary:

- Team collaborated on drafting the Team Management Charter, Project Charter, and Final Report, Gantt Chart. After Edwin completed the Gantt Chart, the rest of the group, including myself finished what Griffin had started with the Team Management Charter. I took the lead for the Project Charter and made the Project Background, Details, Goals, and Impact section and took it to review with the team and they refined the document. Griffin also sent out emails to the instructors to get their sign off on the Project Charter for us to continue and get the sponsor's signature.
- Process for next couple of days is to finish the Project Charter document and fully update the final report and submit all of our submissions into the cover sheet for submission on M2.
- The Project Charter is included below for clear reference and as a reminder:

Project Background

This project introduces students to the challenges and workflow of aerodynamic wing design using a multi-fidelity modeling approach. In industry, engineers frequently blend low and high-fidelity models to balance computational efficiency with accuracy. Students will follow a similar approach, integrating computational tools and engineering judgment to develop, analyze, and refine optimized wing shapes for various aircraft configurations. The project draws from disciplines such as aerodynamics, structures, and optimization, giving students a systems-level view of aircraft wing design.

Industry Fundamentals

The aerospace industry in both military and civilian fields has remained one of the most competitive and innovative sectors globally. Aircraft design must meet rigorous safety, performance, efficiency, and cost requirements, making optimization one of the key foundations in aerospace research and design. Rapidly progressing computational power and ability to process input data through software simulations has revolutionized the aerodynamic optimizations.

This has allowed the industry to move away from physical and wind tunnel prototyping and in turn shortened and lowered cost for the design process. This leap forward has allowed for innovative optimization and concepts to be proposed and tested at much higher rates. Software that is able to compute fluid dynamics and finite element analysis and pushed forward the ability to explore variables and inputs at a continuously increasing rate.

Lockheed Martin has been one of the legacy aerospace innovators since the industry began and continues to be among the largest aerospace and defense company in the world. One of Lockheed Martin's driving principles in recent years has been investing in research and development for innovative workflows and computational software that allows them to pursue optimization and experimental concepts to accelerate development times and quality.

Randal Richards

Considering this, optimization remains a cornerstone that allows the company to continue in its dominant role in the aerospace industry.

Industry Background

The aerospace industry has long been driven by the pursuit of efficiency and furthering capability. This drive for advancement is what led the early innovations of the Wright Brothers to evolve into engineering marvels such as the Antonov An-225 Mriya and the Lockheed SR-71 Blackbird. Modern aircraft design now relies heavily on computational tools that push the limits of aerodynamics, structures, and optimization while reducing development costs and timelines.

A central theme of these advancements is the optimization of wing design to best suit the role of the aircraft. These adaptations can take the form of camber profile, chord length, winglet size, angle of incidence, material selection, and countless other factors. Such improvements enable aircraft to better overcome aerodynamic forces (lift, drag, weight, and thrust) resulting in configurations that are faster, more stable, lighter, more fuel-efficient, and quieter. To achieve this, engineers employ a wide spectrum of tools: low-fidelity methods, such as panel codes, provide rapid early-stage estimates, while high-fidelity simulations, such as Computational Fluid Dynamics (CFD), deliver detailed accuracy at higher computational cost. This multi-fidelity approach has become standard practice across the industry, enabling companies like Boeing, Airbus, and, of course, Lockheed Martin to refine designs efficiently while meeting strict performance goals.

Beyond corporations, universities and research groups are also key players and a primary driving force in advancing multidisciplinary design optimization (MDO) practices. Despite progress, challenges still remain. Notably, low-fidelity models often fail to capture complex aerodynamic phenomena, while high-fidelity simulations can be prohibitively expensive and require specialized hardware, such as supercomputers. Addressing these trade-offs continues to be a central focus within the industry, as the aerospace sector seeks to design faster, more efficient, and more sustainable aircraft for the future.

Project Purpose / Problem Definition

Lockheed Martin seeks a repeatable, CAD-centric wing design optimization pipeline to cut drag and other inefficiencies using realistic geometric and structural constraints. Engineers are going to use research gathered independently on current aircraft wings and design processes and translate it to a digital workflow that accelerates flight efficiency. Deliverables will include initial-to-optimized wing comparison, low & high-fidelity CFD results, documented start-to-finish process (programs, configurations, metrics), and final CAD models for unmanned drones, with performance deltas versus the baseline

Projected Project Impact

The project demonstrates a workflow to create optimized wing designs by integrating both low-fidelity and high-fidelity aerodynamic models into a unified workflow. The impact of this project will help reduce reliance on purely high-fidelity models by significantly decreasing computational costs and turnaround time. This will enable faster iteration cycles, which will help accelerate the preliminary wing design phases. The hybrid approach reduces cost, accelerates certification processes, and enables exploration of unique wing designs while being computationally affordable. This project will enable designers to follow a predictable workflow which will generate safe wing designs under a multitude of conditions.

Goal Statement

- What is the project objective? What does the customer want to achieve? What is the deliverable?

The goal of our project is to develop and document a complete aerodynamic wing optimization workflow. This workflow will integrate CAD, aerodynamic analysis (both low- and high-fidelity), structural analysis, and optimization methods for various aircraft configurations. Our capstone partner, Lockheed Martin hopes to provide students with a hands-on experience in real-world aerodynamic optimization. They hope to teach the students how to balance accuracy, efficiency, and design trade-offs using industry relevant tool and workflows. Our deliverable will be a final report and presentation summarizing the optimization workflow. This final report and presentation will also depict the students optimized wing design as well as their rationale behind key decisions, accompanied by documentation of trade-offs and lessons learned.

Meeting #2 w/ Dr. Banu (Supervisor)

09/22/2025 (30min)

Pre-Meeting

- Before meeting with Dr. Banu, our team reviewed the agenda and identified the key points we wanted to clarify with the instructor. We discussed the models and programs we plan to use, the potential challenges they might introduce, and the responsibilities each member will take on. While the conversation itself was not deeply detailed, it helped us continue shaping our understanding of the project scope.
- During the discussion, Cooper highlighted that the model and workflow we are developing may not fully align with the original objective he outlined—focusing on the unmanned drone. However, I am beginning to see more clearly how our work ties into the broader goal. In particular, we can parameterize the aircraft wing in CFD flow models, allowing us to vary its length, spacing, and material properties. This flexibility

will enable us to optimize wing performance, visualize the results, and plot them to guide design improvements for the aircraft.

Meeting Notes

(45min)

- Dr. Banu went on to state that we need to update our commitments to our roles and that our project charter was the best shes seen! Great news. Also, that we need to be able to research for the next two weeks. She made it very clear that shes readily available to reach out to the University for resources we may need for the capstone project.

In Class in SLC:

09/23/2025

The first part of our meeting was this:

- **8-9 Slides | 8-10 min**
- **Title**
- **Agenda**
- **Intro/summary/background**
- **Needs finding**
- **Value opportinities**
- **Competitors**
- **Engineers Specifications**
- **References**
- **Summary Q/A**
- **AL Exercise – Activity #1**

Activity 1 – Search for applicable codes, standards, and regulations

- **Round 1 – Split your team into two groups (10 min)**
    - Group A - search applicable codes, standards and regulations using google or other search engine
    - Record searches in your notebooks
- Group B - Search applicable codes standards and regulations using Co-pilot or other LLM AI
    - Record AI prompts in your notebook

- Q: What are the applicable engineering codes standards and regulations that we should be using?

Great question — since you're working on **aircraft wing optimization** for a **Lockheed Martin project**, you'll need to frame your work under a combination of **aerospace engineering codes, standards, and regulations**. These are typically broken into three categories:

1. Regulatory Framework (Governmental / Certification Authorities)

These are mandatory and govern how an aircraft (or its components) must be designed, analyzed, tested, and certified.

- **FAA (Federal Aviation Administration, U.S.)**
  - *14 CFR Part 23/25* – Airworthiness Standards for Normal, Utility, Acrobatic, and Transport Category Aircraft.
  - *14 CFR Part 21* – Certification procedures for products and parts.
  - *14 CFR Part 33* – For propulsion-related impacts (if loads transfer to engines).
- **EASA (European Union Aviation Safety Agency, EU)**
  - *CS-23 / CS-25* – Equivalent to FAA Part 23/25, for certification in Europe.
- **DoD / MIL-Standards** (if military context, likely since Lockheed is a defense contractor):
  - *MIL-STD-1530D* – Aircraft Structural Integrity Program (ASIP).
  - *MIL-HDBK-516C* – Airworthiness Certification Criteria.
  - *MIL-STD-810H* – Environmental Engineering Considerations (temperature, vibration, etc.).

2. Engineering Standards (Professional / Industry Codes)

These guide structural analysis, materials, and computational modeling:

- **SAE International (Society of Automotive Engineers)**
  - *SAE ARP* (Aerospace Recommended Practices) – Guidance on aero design and testing.
  - *SAE AMS* (Aerospace Material Specifications) – For alloys, composites, treatments.
  - *SAE AIR* (Aerospace Information Reports) – Reference practices.
- **AIAA (American Institute of Aeronautics and Astronautics)**
  - CFD and structural optimization best practices.
  - AIAA Standards for Verification & Validation (V&V) of CFD simulations.
- **ASTM International**
  - *ASTM F44 / F44.05* – General aviation aircraft standards.

    - *ASTM D3039 / D3479* – Tensile and fatigue testing of composites (if composite wings).
- **ANSI (American National Standards Institute)** – Governs how numerical methods and material testing align with U.S. standardization.

3. Modeling, Testing, and Safety Compliance Standards

- **CFD / Simulation Standards**
    - *AIAA Guide for CFD Verification and Validation* – ensures your wing optimization models meet credibility benchmarks.
    - *ISO 9001* – Quality management in modeling & documentation.
- **Structural / Fatigue Standards**
    - *FAA AC 25.571* – Damage-tolerance and fatigue evaluation.
    - *ASTM E647* – Fatigue crack growth rate.
    - *ASTM E466* – Axial fatigue tests.
- **Materials & Manufacturing**
    - *SAE AMS* series for aluminum, titanium, composites.
    - *Nadcap* (if manufacturing applies) – Aerospace supplier accreditation.

4. Lockheed Martin Internal Requirements

Even beyond public standards, Lockheed has **internal design guides, configuration control, and digital engineering practices** (often tied into **AS9100**, the aerospace quality management system).

- *AS9100D (Rev D)* – Aerospace Quality Management Standard (based on ISO 9001 but stricter).
- Proprietary Lockheed design criteria (you'll often see references like **LMS-xxxx** documents if you get access).

- **Round 2 – Convene Teams A and B (5 min)**
    - Compare lists and decide which C/S/R are legitimately applicable to your solution
    - Add the information to your specs table

In Class EELD Activity:

Summary:

Randal Richards

As we prepared for stakeholder identification and interviews, we discussed how different people may have different ideas of opinions of what should be done and how. The activity was a survival exercise where we were given a scenario and a list of supplies that we had to rank based on what we felt was most and least important. Once we filled the sheet out, we then had to compare with the group and come to a consensus. Once we compared, we were given the opinion of an expert and then had to find the difference between the rankings. Using this we then noted if we were survived better as an individual or group

Bi-Weekly Meeting

09/29/2025

In attendance:

Lockheed Martin – Matthew Seay

University of Georgia – Randal Richards, Dr. Eliza Banu, Martin Vassilev, Edwin Harper, Griffin Cantrell, Cooper Mendlinger, Garrett Kuhn

Pre-Meeting:

- Cooper suggested that we split the research into two different paths. Analyzing flight paths with supersonic and subsonic so that's a pretty easy one. There's a lot of different ways to describe an air foil as well so that's a question that we are going to get into.

My Meeting Notes:

- Mentioned Paul Mccord (Senior Fellow) expert and authoritative force on conceptual design aerodynamics. Optimizing mastermind. Mentioned Bob (air vehicle lead) and conceptual configurator. A mastermind behind aircraft sizing and their studies.
- Cooper asked how much we should focus on sub-sonic. Matt responds, "You want to handle both super and sub sonic performance and the 2D effects that you'll see on x-foil will not be the same as you draw oi with a 3D wing. He uses x-foil at the beginning when he's trying to come up with a subsonic performance parameter. It's been a multi-Mach test and they usually start with something that's been driven before. Having different airfoil sections would come into play and they used x-foil to determine how much more fuel they can sneak in with these sections. At the end most of the models get passed on through CFD. Wants a diversified Mach-regime and different flight patterns. As much as we want to think of optimization as independent of the initial point, the linear tools may be less sensitive to the initial design point. So, there's still a lot of merit that comes with predesign and the initial geometry with that."

Randal Richards

- "When Northrop was designing the B-21 they had one goal: maximize range at all costs. When they were doing wing designs, they had lateral and longitudinal stability considered. They use CFD to understand their control surfaces and then continue to optimize their wing. Depending on the aircraft you have, engineers will run cambered twist study and come up with the 90% answer and give it the thickness it would need. Run CFD tests. CFD sizing, run cambered, resizing with CFD tests. Cooper suggests a divide and conquer application with peer-reviewing.

Summary:

- We discussed two key experts: Paul McCord, a Senior Fellow in conceptual design aerodynamics and optimization, and Bob Clark, the air vehicle lead and conceptual configurator with strong aircraft sizing expertise. Cooper asked how much to focus on subsonic work, and Matt said we must cover both subsonic and supersonic performance. He explained that two dimensional XFOIL results will not match a full three dimensional wing, so XFOIL is best for early section work before moving to CFD. He wants a mix of Mach numbers and flight patterns, and he stressed that the starting geometry still matters because linear tools are less sensitive to the starting point but not immune to it. As an example, the B 21 program had one main goal to maximize range. Teams paired stability work with CFD to study control surfaces, then continued optimization by running camber and twist studies to get the ninety percent answer, assigning thickness, doing CFD sizing, and refining as needed.
- From meeting two we added that the F 35 used many competing performance points with active surfaces, which pushed heavier use of linear theory first and CFD for refinement, while a single dominant objective like the B 21 leans more on CFD. The rule of thumb is to use linear theory to reach about ninety percent of the answer and then refine with CFD, and fewer performance points means more CFD. The path is to turn requirements such as a twelve hundred nautical mile range into design targets, size the wing, pick airfoils and planform and twist, keep the number of variables manageable, run camber and twist studies, tailor spanwise thickness for performance, fuel volume, and weight, and then run CFD checkouts. If gaps remain, step up to higher fidelity and do local refinements. Coordination spans flight sciences and also thermal, design and configuration, and propulsion, with careful tracking of forces and moments and attention to aero propulsion coupling. For tools and validation we will use FUN3D with adjoint methods for sensitivities and the ONERA M6 wing as a standard validation case. The workflow should model the fuselage, start from the Valkyrie airframe, reoptimize the wing while constraining the root chord, and vary span, taper, camber, and twist, while keeping parametric weight equations based on standard texts. The next meeting goal is to present a complete optimization process, show initial test cases, and demonstrate solid familiarity with the toolchain.

Randal Richards

Milestone #3 Stakeholder Mapping and Identification

10/03/2025

Project Stakeholders and Needs:

Stakeholders

Mr. Seay (Primary)

Description:

Mr. Seay serves as our primary stakeholder and funding source for the Capstone Design project. In this role, he provides financial support, sets the strategic objectives, and ensures that the project aligns with Lockheed Martin's mission of advancing aerospace design and optimization. As such, he is the liaison between the capstone team and Lockheed Martin's engineers, reviewing deliverables at key milestones set throughout the project and offering feedback to keep the project consistent with industry standards. With high influence and interest in the capstone project, Mr. Seay's guidance and support are essential not only for the project's successful completion but also for fostering future collaboration between Lockheed Martin and the University of Georgia.

Needs:

Mr. Seay needs a credible, multi-fidelity wing optimization workflow with traceable assumptions and verification, aligned to Lockheed standards and the project goals he authorizes. He needs clear progress with biweekly updates, defined milestones, and simple performance metrics. Additionally, he requires professional rigor with version control, reproducible code and data, secure handling of sensitive material, and clean documentation. He needs useful deliverables like a chosen tool set, readable flowcharts and code, a validated unmanned aircraft wing case study, and a short final brief.

Bob Clark (Primary)

Description:

Mr. Bob Clark is a Conceptual Design Engineer at Lockheed Martin specializing in Conceptual Design Engineering and is an Aircraft sizing expert at Lockheed. With his knowledge and expertise, we hope to gain insight into how our research will affect his job and how his experience can optimize results.

Needs:

Randal Richards

Mr. Clark needs a clear workflow that delivers fast and dependable sizing inputs early in design so he can make confident calls. He wants simple controls for span, area, aspect ratio, camber, and twist that show cause and effect without extra noise. He needs results that tie back to plain assumptions and pass quick hand checks, with clean charts and tables he can drop into a concept study. On our stakeholder map and power–interest matrix, he sits in the Manage Closely group alongside our sponsor, so he also needs steady visibility, tidy documentation, and a repeatable process that runs on common tools and shares data smoothly with other teams.

Paul McClure (Primary)

Description:

Paul McClure is a Lockheed Martin Fellow with over 37 years of experience at the company. He spent three decades as a Senior Staff Engineer before receiving one of Lockheed's highest technical honors in 2018, Lockheed Martin Fellow.

He received his Master's degree in Aerospace, Aeronautical, and Astronautical Engineering from the University of Illinois in 1987 and is now an expert in aerodynamics and project management. Throughout his career he has focused on advancing aircraft performance, optimizing aerodynamic design, and guiding technical teams through complex engineering challenges.

With Mr. McClure's professional experience, we hope to gain insights into the industrial side of airplane wing optimization and design in addition to receiving advice on how we could better go about the issue Mr. Seay has provided us.

Needs:

Mr. McClure needs a workflow that turns early concepts into decision-ready results with minimal fuss. He wants inputs that are easy to set, assumptions that are stated up front, and outputs that line up with quick checks and past data. He looks for concise summaries, clear before-and-after plots, and a short list of actions. On our stakeholder map and power–interest matrix he is in the Manage Closely group, so he also expects regular updates, neat documentation, and a visible risk list. He needs a method that blends low and high fidelity tools, keeps full traceability, and is simple for other teams to reuse.

Wind Tunnel Team (Secondary)

Description:

In addition to our airplane wing optimization project, the university has also coordinated a Wind Tunnel Platform Design Team. Due to the strong overlap between design/optimization and testing we felt it logical to partner with this other capstone team. With this partnership we hope to gain insight into what fluid mechanic characteristics the other team is designing for, what experience they have had, and how we can better communicate/coordinate in the future.

Needs:

The Wind Tunnel Platform Design Team needs a clear handoff between our design outputs and their test plans. They need agreed test points, sensor lists, and data formats so results line up with our models. They need our assumptions, target conditions, and acceptance bands so they can size the rig and pick the right instrumentation. They need a shared calendar for model delivery, balance availability, and tunnel time. On our stakeholder map and power–interest matrix they sit in the Keep Informed group, so they need steady updates, quick responses to issues, and a simple place to share files and results.

Dr. Eliza Banu (Secondary)

Description:

As a Ph.D. in Mechanical Engineering and a professor of Fluid Mechanics, Dr. Banu is our acting Project Supervisor and well connected within the engineering field. Utilizing her experiences and expertise we hope to gain insights into specialized computational fluid dynamics that can permit access to key computational resources, research opportunities, and learning methods for approaching optimization.

Needs:

Dr. Banu needs a clear plan for how we will learn and apply advanced fluid methods. She needs defined milestones, simple deliverables, and clean documentation she can review and approve. She needs our assumptions, inputs, and results to be traceable and reproducible so they meet university standards. She also needs a short list of resources and access requests so she can open doors to labs, software, and collaborators. On our stakeholder map and power interest matrix she is in the Keep Informed group, so she needs steady updates and quick responses to questions.

Madison Warner (Tertiary)

Description:

Madison (Maddie) Warner is a May 2025 UGA alumna with a B.S. and A.B. in Geography. Currently, she is instructing at Air Athens and conducts private charters through Athens Pilot. She is a certified commercial pilot and flight instructor–instrument (CFII) with over 1,000 hours in the cockpit of various aircraft, including the Cessna 172, Piper Archer, Cessna Citation M2, and Beechcraft King Air 200. In addition, she holds a seaplane rating, adding to her diverse aviation background. Maddie hopes to one day join Delta Air Lines as a captain, where she can continue to apply her skills and passion for aviation in a commercial setting.

Randal Richards

Needs:

Maddie needs design outputs that connect directly to pilot tasks like takeoff distance, climb rate, stall margin, handling qualities, and fuel burn. She needs testable scenarios and data in pilot-friendly formats such as clear plots, short briefs, and simple checklists. She needs a fast feedback loop where her notes turn into concrete design changes and updated results. She needs early looks at simulation runs and planned test points so she can flag safety or usability concerns. On our stakeholder map and power interest matrix she is in the Manage Closely group, so she also needs regular updates, a clear schedule, and visibility into risks and mitigations.

Previous Lockheed Capstone Group (Tertiary)

Description:

The central objective of their research is to answer the question of how current deep learning technology can be applied to transonic wing design. Deep learning models are able to utilize historical and inputted data to produce a desirable result. This group produces a comprehensive set of high-fidelity simulations on transonic wings using Ansys Fluent as their main tool of simulations. They employ traditional computational fluid dynamics (CFD) methods with a parametric approach to produce designs and change aspects of their simulations.

Needs:

This group needs a clean data handshake with our team so their high-fidelity results can validate our models. They need agreed transonic test conditions, shared metrics, and simple data formats so comparisons are fair and fast. They need clear baselines, versioned cases, and short notes on model limit s and training data, so results are interpreted correctly. They also need a plan for joint experiments, including when we deliver geometry and when they deliver simulation outputs. On our stakeholder map and power interest matrix they sit in the Keep Informed group, so they need steady updates and quick responses without heavy meetings.

Contributions

- I went ahead and completed the stakeholder questions for their interview and also made sure that every single stakeholder in the final report had their needs clearly highlighted and documented for the group. I added these beforehand and the stakeholder interview questions below. I also documented the teams efforts below that as well with the two stakeholder maps considered.

Paul McClure:

Stakeholder description:

Paul McClure is a Lockheed Martin Fellow with over 37 years of experience at the company. He spent three decades as a Senior Staff Engineer before receiving one of Lockheed's highest technical honors in 2018, Lockheed Martin Fellow.

He received his Master's degree in Aerospace, Aeronautical, and Astronautical Engineering from the University of Illinois in 1987 and is now an expert in aerodynamics and project management. Throughout his career he has focused on advancing aircraft performance, optimizing aerodynamic design, and guiding technical teams through complex engineering challenges.

With Mr. McClure's professional experience, we hope to gain insights into the industrial side of airplane wing optimization and design in addition to receiving advice on how we could better go about the issue Mr. Seay has provided us.

Interview:

Q) Question: At the conceptual design stage, which 3–5 performance metrics do you look at first for a transonic wing, and why those over others?

A) Answer

Q) Question: When you're moving from low-fidelity to high-fidelity analyses, what simple "gate checks" tell you it's worth spending CFD time on a candidate wing?

A) Answer

Q) Question: For student teams building a minimum-viable optimization workflow, what tool choices or modeling simplifications tend to give the best payoff early without harming later refinement?

A) Answer

Q) Question: What typical constraints or requirements (structural, aero, or certification-adjacent) most often force changes to an otherwise "good" aero shape at the conceptual phase?

A) Answer

Q) Question: If we handed you our workflow and an optimized wing today, what are the first things you would look at that would build your trust in our results?

A) Answer

Interpretation:

What did we learn? What was (un)expected? How can we use this?

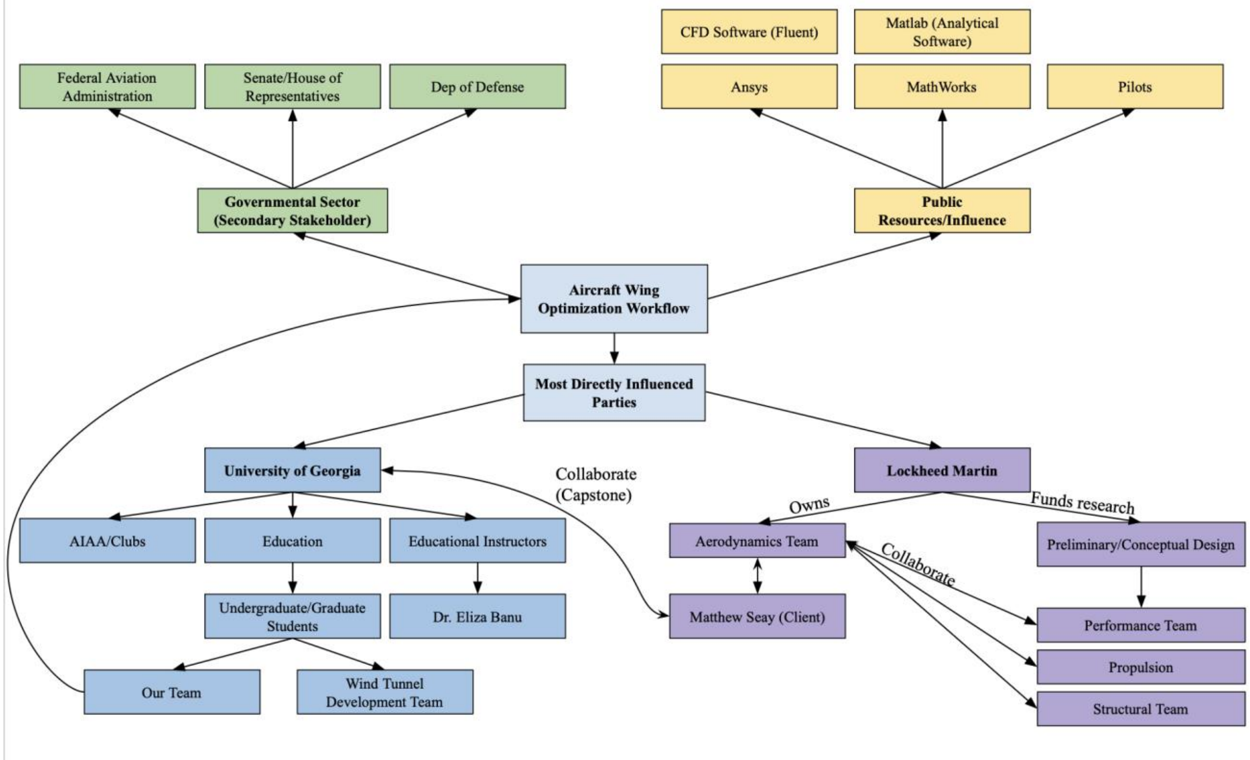


Figure 2.1: Stakeholder Mapping

- The diagram shows who exchanges what with whom and when. The arrows document the flow of requirements, data, reviews, and decisions among four clusters: Lockheed Martin, the University of Georgia, public resources, and government and standards. It also shows our team in the middle as the integrator. The map gives us the path for each handoff and the checkpoints where feedback returns to the team.

This is why the needs statements look the way they do. For Mr. Seay and the Lockheed engineers, the arrows from our team to Lockheed call for traceable results, short briefs, and regular reviews. For Dr. Banu and UGA, the arrows call for clear learning plans, reproducible runs, and specific access requests. For the wind tunnel partners and pilots, the arrows show test points, sensor lists, and pilot facing metrics moving both ways. For government and standards, the arrows point to compliance checks and reference methods that we must satisfy before a milestone is complete. For public resources such as software and datasets, the arrows remind us to manage licenses, versions, and validation cases.

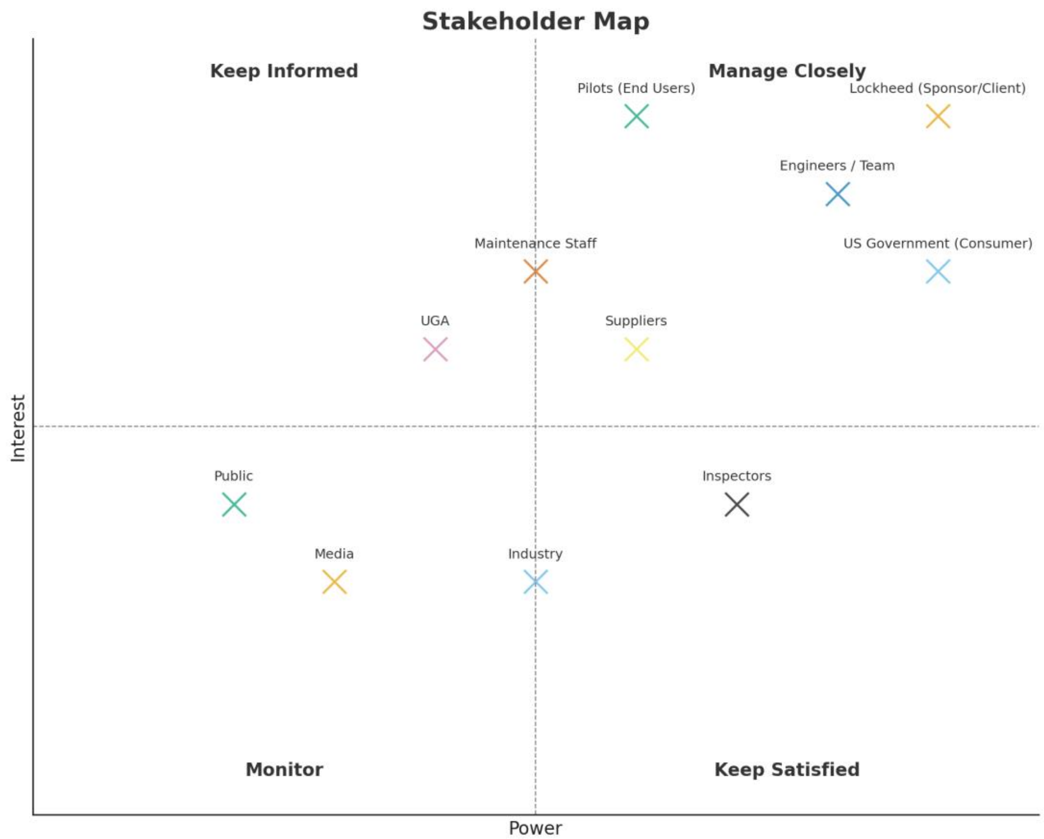


Figure 2.2 Stakeholder Map with Interest/Power

- The map documents who matters most, what they care about, and how often we should engage them. It sorts each group by power and by interest. That lets us set the right goals, messages, and deliverables for each audience. It also keeps us from over-serving low-interest groups and under-serving the ones who will grade our work.

For those in Manage Closely, like Mr. Seay, the pilots, and our core team, we wrote needs that ask for clear progress, regular updates, and decision-ready results. For those in Keep Informed, like Dr. Banu and the Wind Tunnel Platform team, we wrote needs that focus on steady updates, clean handoffs, and reproducible data. For those in Keep Satisfied, like inspectors and industry partners, we highlighted compliance, neat documentation, and simple summaries they can review quickly. For those in Monitor, like the public and media, we kept it minimal.

Summary of Milestone:

Here is what the stakeholder maps do for us right now and what it means for the milestone:

- First, they turn a vague goal into specific handoffs. The arrow map tells us exactly who gives what to whom and when. The power and interest map tells us how often to talk to each group and how much detail they need. Together they cut guesswork, reduce rework, and keep our effort aimed at what will be reviewed and approved.
- Second, they let us write simple interface contracts. For each arrow we define inputs, outputs, format, owner, and frequency. That becomes our checklist for data, models, and reviews. It also makes it easy to spot gaps early, like a missing test point or an unclear metric.
- Third, they clarify success. Each stakeholder need becomes a short acceptance statement for the milestone. If we meet those statements, the review goes smoothly. If not, we know what to fix next.

Meeting with Paul McClure

10/08/2025 6:00-7:00PM EST

Meeting Summary — Aerodynamic Workflow & Early-Phase Design Guidance:

Background & Design History

The stakeholder brings 38 years of experience, including F-22 work, flight testing on F-22 and F-35 prior to the major contract awards, and exposure to higher-TRL drag-reduction approaches (e.g., riblets). Their focus spans drag-reduction technologies and aerodynamic tool development, including rewriting tools used in primary conceptual design. They emphasized a preference not to jump straight from low-fidelity to high-fidelity CFD.

Program Requirements & Constraints

We still need formal specifications from Mr. Seay (target drag metrics, altitude, and Mach numbers). The guidance is to identify the most challenging requirement and optimize for maximum range while meeting the remaining specs. Constraints vary by basing (e.g., Navy “spot factor” limiting span) and by weight. A weight model must be incorporated and tied to planform span and thickness-to-chord (t/c), using actual thickness—Nikolai is the recommended reference for this integration.

Starting Point & Tooling Philosophy

Do not start with a highly detailed, “optimized” geometry, nor with CFD or simple panel codes. If available, use a code that produces decent predictions for parametric geometries to sweep wing sweep, span, taper ratio, etc., and build a complete aerodynamic database. For sustained-g concerns, use the database to locate polar breaks. The stakeholder endorsed commercially available integration/visualization tools such as JUMP and ModelCenter for rapid parametric exploration (sliders for independent variables and immediate performance feedback).

Randal Richards

Design-Space Exploration
Use carpet plots to visualize sensitivities: two or more independent variables versus one or more dependent metrics, with planes interpolated across the chart. Because there's no single x-axis, multiple overlays are natural and helpful. When running gradient-based CFD optimizations, expect local/global extrema; broad parametric sweeps and visualization are essential to avoid being trapped.

Orderly Progression (Low- to Higher-Fidelity, but Not Prematurely)
Confirm that the design space is well explored before escalating fidelity. Perform wide sweeps (e.g., 0–60° sweep, not just "intuitive" 40–60°) and spans/tapers, then shape clean starting geometry and appropriate airfoils. Use NACA camber lines/thickness distributions as structured starting points. Determine required camber magnitude relative to mission points. Low-order tools can reveal if, say, cruise lift coefficient implies a wing that's larger than necessary—reducing required camber and complexity. (Our current configuration excludes high-lift devices; note this limits control-surface-based solutions.)

Analytic References & Subsonic Checks
For subsonic regimes, consult state-of-the-art charts in Nikolai: e.g., L/D on the y-axis versus span divided by wetted area on the x-axis. With parabolic polars, lines for skin-friction coefficients and span efficiency produce families of straight-line trends; our configurations should fall within reasonable bounds on these charts. These provide early-phase sanity checks before any CFD.

Cost Considerations
Cost is not a primary driver at this stage. The advice is to establish weight credibly first, then map to cost—useful for later trades, but not a gating factor now.

Questions from Milestone #3

Q1. *What is the primary performance metric we should optimize (e.g., range), and how should we incorporate secondary mission metrics such as loiter, sustained-g, takeoff/landing, and basic flying qualities (S&C)?*
Answer: One-page metric hierarchy (objective vs. constraints) with numeric targets and pass/fail checks.

Q2. *What specific criteria indicate we've explored the low-order design space enough to justify escalating to higher fidelity (e.g., CFD)?*
Answer: Readiness checklist (coverage thresholds for sweep/span/taper/camber, stability of trends, sensitivity thresholds).

Q3. *For subsonic early-phase validation, which public/low-cost tools, charts, or references (e.g., Nikolai's L/D vs. span/wetted-area with parabolic polars) should we use, and how should we benchmark our concepts against them?*
Answer: Short methods note citing the charts/tools, plus a benchmark figure placing our candidates within expected bands.

Q4. *Should we run a broad parametric sweep (e.g., sweep 0–60°, multiple spans and taper ratios) to map the design space, rather than narrowing by intuition upfront?*
Answer: Sweep plan (factor ranges and step sizes) and a design-of-experiments table for the aerodynamic database.

Q5. *Before starting CFD, what evidence and artifacts do you expect to see—e.g., carpet plots showing sensitivities, identification of polar breaks, boundary-condition definitions, and integrated loads across all wetted surfaces?*

Answer: Mini-packet with carpet plots, candidate short-list, BC/spec sheet, and integration checks annotated.

Meeting with Bob Clark

10/10/2025 12:30-1:30PM EST

Meeting Summary — Mr. Bob Clark (Conceptual & Structural Design Guidance)

Background & Expertise
Mr. Clark graduated from Georgia Tech ~40 years ago and began his career at General Dynamics (same site today). He spent ~11 years in structural engineering, worked at Northrop Grumman, then returned to advanced design—"drawing airplanes people will never see." He led the original structural arrangement on the F-35, collaborates closely with Matt Seay (whom he regards as an excellent engineer), and routinely blends optimization with concept development. He is from Macon, GA.

Mindset & Early-Career Advice
His core lesson: seek mentorship and humility. Do not posture as all-knowing; you're learning the "language" and the work cadence first. Liaison roles teach how design intent meets the factory floor. He emphasizes mining prior art—internal arrangements and historical derivatives (e.g., Agile Falcon/F-16 and Mitsubishi F-2 parallels). Assume earlier engineers solved much of what you face; stand on their shoulders.

MDAO Framing & Inputs
He likens MDAO to a "Magic 8-Ball"—the usefulness depends on the question you ask. Typical inputs: propulsion, geometry, aerodynamics, analysis (much of our present effort), and weights. Two strong themes: (1) think the problem through before rushing tools; (2) weights drive everything. Younger teams often under-appreciate weight's leverage—study Raymer, Nicolai/Nikolai, and similar references, especially the weight chapters and equations, until the parameters that matter to weight engineers are second nature.

Sizing & Early Methods
For first cuts on wing area and thickness (t/c), use the classic charts and rule-of-thumb ranges by regime to establish credible starting points. Discipline yourself to finish the current step before escalating fidelity—define a completion gate for each phase so the process marches forward cleanly.

Design of Experiments (DOE) & Run Matrices
Use a structured DOE run matrix to map the space rather than narrowing by intuition. Broad coverage prevents local-optima traps and gives clearer trade visibility for down-selection.

Collaboration & Communication Patterns
Coordination with aero typically happens via scheduled meetings (often camera-off for security). Real collaboration depends on the team; strong facilitation helps. Bring digestible summaries, clear model handoffs, decision logs, and a little humor—earn trust by being prepared and "semi-intelligent" rather than loud.

Adoption Criteria for a Usable Workflow
For his team to actually use a student-built workflow: it must be easy, transparent about assumptions and limits, produce results they trust, and fit within IT/security constraints. GUI frictions should be minimal; speed and clarity matter.

Starting from the Valkyrie Configuration
Begin with open sources: collect and document every public parameter you can (e.g., Wikipedia and references). Build a reconciled baseline spec table. From there, center early optimization on L/D, separating wing drag from trim drag. For camber and twist, set up clean parametric studies; note our no-control-surface constraint and plan trim explicitly.

Questions from Milestone #3

Q1. *How did you get your background in aerospace engineering?*
Answer: Mentorship and liaison roles; learn the language, study prior arrangements, build on existing solutions.

Randal Richards

Q2. *What methods or rules of thumb reliably estimate wing area and thickness before higher-fidelity analysis, and when do we move to the next step?*
Answer: Use vetted charts; complete each sizing step and sanity-check before escalation.

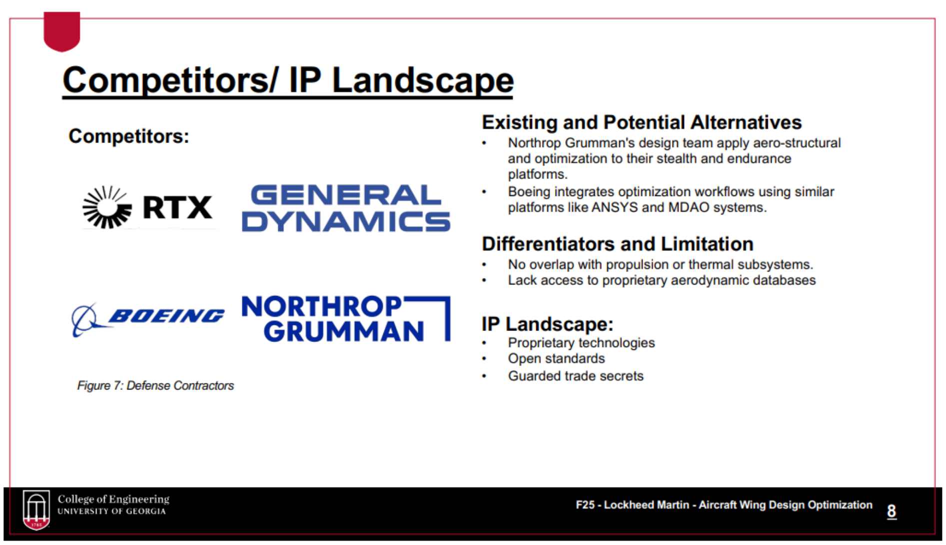
Q3. *Which early parameters have the greatest leverage (area/wing loading, AR, taper, sweep, t/c)?*
Answer: Area dominates early performance; AR governs induced drag (don't neglect other drags). t/c is regime-dependent; sweep often dominates for signature alignment in military roles.

Q4. *How do you and your team coordinate with aerodynamic analysts—models, summaries, iterative loops?*
Answer: Meetings on Teams (no cameras), variable collaboration norms; clarity, brevity, and documented decisions help.

Q5. *If we aimed to make our workflow valuable enough for your team to use, what would make it part of your regular design process?*
Answer: Ease of use, explicit limits/assumptions, trustworthy outputs, low GUI friction, and alignment with IT/security constraints.

Q6. *What should we do when starting with the Valkyrie configuration?*
Answer: Gather public imagery/specs, build a reconciled baseline parameter table, then plan L/D-focused camber/twist trades with trim drag included.

Milestone #4 Project Understanding:



Engineering Specifications Table

Category	Stakeholder Needs	Design Requirements	Acceptance Criteria	Qualifications	Regulatory Requirements
Safety	Data Integrity	Ensure workflow with secure computational environment	Workflow runs safely without risk of corrupted simulations or data loss	Requirement	STD-8719.13C (Software Safety)
Quality Control	Accurate aerodynamic results	Workflow produces traceable and precise results	CFD results and low-fidelity models validated against known standards	Requirement	AIAA R-101-2013 (CFD Verification & Validation)
Cost	Affordable and efficient use of resources	Efficient resource usage	Optimized computational workflow	Requirement	N/A
Repeatability	Must be reusable for given wing configurations	Modular Workflow with input parameters	Re-run workflow with on various wing configurations	Requirement	N/A

College of Engineering
UNIVERSITY OF GEORGIA
F25 - Lockheed Martin - Aircraft Wing Design Optimization 9

Contributions:

Competitors & IP Landscape

- Led a market/IP scan to identify current and emerging competitors and any IP that could constrain our approach.
- Distilled findings into "Existing vs. Potential Alternatives," with clear differentiators and known limitations to guide our design space and de-risk overlap.

Engineering Specifications

- Built the initial Engineering Specifications table and standardized the format for quick read-through.
- Resolved the "how to quantify" debate by defining outputs as clear binary I/O where appropriate, removing low-value filler and keeping only decision-critical specs.
- Ensured each spec ties directly to a verifiable outcome, improving traceability from requirement → test.

Impact

- Provided a focused view of the external landscape to avoid redundant work and IP conflicts.
- Streamlined specs into an actionable checklist, speeding future design reviews and go/no-go decisions.

Ready for the slide (one-liner)

Randal Richards

- “Owned competitor/IP review and authored a streamlined, binary-output Engineering Specs table—removing non-essential details and aligning each spec to a testable outcome to accelerate design decisions.”

Wing Research

10/13/2025

XQ-58A Wing Optimization — Working Notes (Low-Fidelity Phase)

Scope

- Build a multi-fidelity wing optimization workflow that starts cheap and fast (parametric geometry + low-order aero + basic weights), then graduates to higher-order CFD once a good design region is found
- Use the XQ-58A (Valkyrie) as the first reference configuration: keep fuselage/root constraints and optimize wing parameters (span, taper, sweep, twist, airfoil selection) for mission-level metrics (range/endurance/L/D at specified conditions)
- Ground all early weight/performance estimates in Raymer to keep the math traceable and defendable to stakeholders (Lockheed Martin sponsors + UGA advisors)

XQ-58A

- The XQ-58A baseline we were given is a compact, high-subsonic unmanned air vehicle with a slender fuselage, modest internal volume, and a moderately swept trapezoidal wing sized for efficient cruise and reasonable stowage; our model fixes the fuselage and root constraints and lets us work the planform and twist.
- The wing shows medium aspect ratio with limited thickness for volume, so structural weight and stall margin become active constraints as span and taper move.
- Aerodynamically, the geometry is set up for clean lifting-line behavior with predictable induced drag trends.
- From a Raymer perspective, the configuration is sensitive to aspect ratio, sweep, and t/c in the wing weight correlation, which tempers any “more span” instinct and pushes us toward smarter twist and sectional airfoil choices to grow L/D without a weight cliff.

- Overall, this airframe is a good candidate. Twist and airfoil tailoring buy early performance, while planform changes must be balanced against structural penalties and potential transonic effects before we hand the best candidates to CFD.

Reference and numbers

- Low-cost, high-subsonic unmanned air vehicle focused on efficient cruise and long range.
- Baseline used for our low-fidelity optimization with fuselage and wing-root constraints held fixed.
- Working analysis mass model used in our studies: takeoff weight 4,540 lb, dry weight 2,500 lb, fuel 1,440 lb; TSFC held at 0.55 1/hr for early range scoring.
- Reference cruise condition set in high-subsonic regime; trim enforced so lift equals weight at the mission point.

Wing Geometry and Characteristics

- Planform: moderately swept trapezoidal main wing; V-tail on the airframe.
- Span: approximately 27 ft class.
- Sweep: moderate leading-edge sweep; quarter-chord sweep used in Raymer inputs and for drag checks in our notes.
- Taper: moderate taper from root to tip; rectangular-to-trapezoid transition consistent with a compact UAV planform.
- Dihedral: small positive dihedral for lateral stability.
- Twist: washout from root to tip to delay tip stall and balance induced loading for higher L/D.
- Thickness distribution: higher t/c at the root for structure and volume, thinning toward the tip for lower drag and better stall behavior.
- Mean aerodynamic chord and reference area taken from the OpenVSP export for consistency across MATLAB and Python tools.
- Structural notes for modeling: composite skins with internal spars and ribs assumed in weight proxies; wing weight estimated with Raymer's correlation using area, aspect ratio, sweep, t/c, design gross weight, dynamic pressure, and taper.
- Aerodynamic modeling: section polars from XFOIL or curated tables; lifting-line or VLM for 3D lift and induced drag; parasite drag from wetted-area and Cf; simple wave-drag gate applied only when sweep and Mach approach transonic.

Computations and Readable

- Taper ratio = c_tip / c_root measured directly from planform.
- Chord distribution c(y) from trapezoid geometry for spanwise loading setup.
- Quarter-chord and leading-edge sweep angles taken from planform; consistent definition recorded for all runs.
- Twist schedule θ(y) extracted at stations; total washout = θ_tip − θ_root.
- Thickness ratios t/c at root, mid, and tip captured from airfoil sections.
- MAC and its spanwise location computed from taper; used to align aerodynamic and structural references.

Bi-Weekly Client Meeting Report

Client Meeting Summary — Lockheed Martin Sponsor

10/13/2025

Project: XQ-58A Wing Optimization Workflow
Location: Virtual
Duration: 60 minutes

Attendees

- Lockheed Martin: Matthew Seay sponsor, contacts to Fellows shared
- UGA Team: Griffin Cantrell, Edwin Harper, Garrett Kuhn, Cooper Mendlinger, Randal Richards, Martin Vassilev
- Faculty: Dr. Eliza Banu mentor

Meeting Purpose

- Align on low-fidelity optimization plan and how it feeds higher-fidelity CFD
- Confirm geometry constraints for the Valkyrie baseline
- Capture expectations for deliverables and review cadence

Sponsor Highlights

- Start with linear tools and section methods for the ninety percent answer
- Focus on camber and twist for the first gains
- Keep fuselage and wing-root fixed and reshape only the wing
- Validate with a standard case like ONERA M6 before heavy refinement
- Document all assumptions and unit choices in the notebook

Project Scope Confirmed

Randal Richards

- Build a multi-fidelity pipeline: CAD to low order aero to basic structures to CFD spot checks
- Use OpenVSP for parametric geometry and export consistent references
- Use XFOIL for section polars and VLM or lifting-line for 3D loads and induced drag
- Use Raymer for wing weight and simple system stand-ins for zero fuel weight
- Score designs with Breguet range at a single cruise point using constant TSFC for now

Geometry and Constraints

- Valkyrie baseline as provided
- Wing span, taper, sweep, twist are open design variables
- Wing-root station and fuselage geometry are locked
- Minimum thickness and sane tip chord enforced
- Moderate sweep preserved to avoid risky transonic behavior at cruise

Data and Tools

- MATLAB as driver for optimization and plotting
- Python bridge to edit OpenVSP and return geometry metrics
- GA and CMA-ES for search in rough design spaces
- ANSYS Fluent reserved for a small set of down-selected wings
- Mechanical used for quick structural sanity checks with simple spar load models

Performance Point

- High subsonic cruise at a single mission weight for early trades
- Trim enforced so lift equals weight
- TSFC held constant to stabilize ranking

What Success Looks Like to Sponsor

- Clear, traceable workflow diagrams and repeatable scripts
- A short list of candidate wings with documented gains in L over D and estimated range
- Evidence that constraints were respected and that geometry is buildable
- A brief validation step against a known case and at least one CFD check

Risks Noted

- Low-fidelity blind spots for shock and interference
- XFOIL fragility at tough Reynolds and angles
- Over-emphasis on aspect ratio without weight penalties
- Inconsistent sweep definitions across tools

Mitigation Approach

Randal Richards

- Use penalties and safety cushions near transonic edges
- Precompute and smooth polar sets where needed
- Log units, sweep definition, and bounds on every run
- Cross-check a few designs in CFD to anchor the model

Deliverables and Cadence

- Biweekly progress notes to sponsor by email
- Mid-semester: draft workflow and tool down-select
- End of fall: optimized UAS wing candidate set with rationale
- Spring: refinement and expansion to additional regimes

Open Items from Meeting

- Confirm which sweep definition the sponsor prefers in weight calculations
- Confirm load factor to carry in Raymer wing weight during this phase
- Share contact info for Fellows available for quick design reviews

Research: Reymer's Equations and Tables

10/15/2025

Thrust and "specific" measures
Specific thrust (per unit inlet mass flow)
F_spec = T / m_dot_air
Thrust loading (what we actually size to a lot)
T_W = T / W
Wing loading
W_S = W / S
Why this matters

- Raymer treats T/W and W/S as first-cut sizing knobs. We use W/S to set CL at cruise and T/W to see if our drag budget is flyable.
- In our loop we test whether T/W at the selected mission point covers drag plus any small climb or maneuver margin.

Basic aero and drag build-up
Lift and drag at a mission point
L = q S C_L
D = q S C_D
q = 0.5 rho V^2

Induced drag (lifting-line form)
$C_{Di} = C_L^2 / (\pi e AR)$
Total drag
$C_D = C_{D0} + C_{Di}$ (+ C_{Dwave} when needed)
Why this matters

- These are the numbers our optimizer tries to improve via span, taper, sweep, and twist.
- Lower C_D at the trimmed C_L raises L/D and directly increases range for the same fuel.

Fuel flow and "fuel usage drives the path"
Jet fuel flow using TSFC
$dW_{fuel}/dt = c_T * T$
Here c_T is TSFC in 1/s if T is in force units and time is seconds.
Sign convention: fuel weight decreases, so $dW/dt = -c_T * T$.
Cruise simplification (T equals D)
$dW/dt = -c_T * D$
Fuel burn per distance (useful along the ground track)
$dW/ds = (dW/dt) / (ds/dt) = -(c_T / V) * D$
Integrated form along a cruise leg
$W_f = W_i * \exp(- (c_T / V) * \int_s (D / W) \, ds)$
When D/W is small and nearly constant you recover the Breguet form below.
Why this matters

- This is the clean way to tie fuel usage to the *path*. If we change altitude or speed, D and V change, and so does the local burn rate.
- In our notebook we use the constant-point Breguet first, then keep this differential form in mind for step-climb or multi-segment missions.

Breguet range and endurance (early ranking)
Breguet range for jets (constant TSFC, constant L/D)
$R = (V / c_T) * (L/D) * \ln(W_i / W_f)$
Endurance proxy for jets at fixed point
$E = (1 / c_T) * (L/D) * \ln(W_i / W_f)$

Why this matters

- Raymer uses these for conceptual trades. We set c_T constant on purpose in the early loop to stabilize comparisons.
- Maximizing L/D at the mission point gives the biggest first-order push to R and E for the same fuel fraction.

Weight accounting we track in the loop

Zero fuel weight and gross weight
$W_{ZFW} = W_{empty} + W_{payload}$
$W_{gross} = W_{ZFW} + W_{fuel}$

Segment fuel fraction idea

W_f / W_i = product of segment fractions

Example: warmup * takeoff * climb * cruise * reserve

We often collapse to a single cruise-leg fraction for early trades.

Why this matters

- Our MATLAB code keeps W_{ZFW} and W_{fuel} explicit so we can see the impact of wing weight changes from Raymer on the fuel available for range.

Raymer wing weight correlation (imperial form often used)

$$W_w = 0.036 \cdot S^{0.758} \cdot W_{fw}^{0.0035} \cdot \left(AR / \cos^2(\Lambda_{25}) \right)^{0.6} \cdot q^{0.006} \cdot W_{dg}^{0.49} \cdot (1 + \lambda)^{0.1}$$

Where

S = wing area, AR = aspect ratio, Λ_{25} = quarter-chord sweep,

q = dynamic pressure at design point, W_{dg} = design gross, W_{fw} = fuel in wing,

λ = taper ratio.

Why this matters

- This captures how pushing AR or sweep has a structural price. Our optimizer sees the penalty through W_w which raises W_{ZFW} and reduces W_i/W_f leverage in Breguet.

Wing geometry relations we read or compute from drawings

Taper ratio and chord distribution

$$\lambda = c_{tip} / c_{root}$$

$$c(y)/c_{root} = 1 - (1 - \lambda) \cdot (2y / b)$$

Mean aerodynamic chord and location

$$MAC = (2/3) \cdot c_{root} \cdot (1 + \lambda + \lambda^2) / (1 + \lambda)$$

$$y_{MAC} / b = (1/6) \cdot (1 + 2\lambda) / (1 + \lambda)$$

Quarter-chord sweep from leading-edge sweep

$$\tan(\Lambda_{25}) = \tan(\Lambda_{LE}) - \left[(1 - \lambda)/(1 + \lambda) \right] \cdot (1/AR)$$

Why this matters

- These let us tie the notebook’s drawing measurements to the aero and weight equations without needing the absolute scale.

Thrust and drag balance checks (sanity at the mission point)

Trim and balance

$$C_L = W / (q S)$$

$$T \text{ required at cruise} = D = q S C_D$$

Thrust loading check
T_W_required = D / W = C_D / C_L

Why this matters

- A quick T/W check tells us whether a candidate wing is even flyable at the chosen condition with our assumed engine thrust margin.

Load factor, stall margin, and twist logic
Stall margin at each station
C_L_station <= C_L_max_station
C_L_station from lifting-line using c(y) and twist theta(y)
Induced loading target
Elliptic-like loading improved by washout at tip
Twist delta_theta = theta_tip − theta_root
Why this matters

- Raymer and classical lifting-line both push us toward tip washout to delay tip stall and trim induced drag. We give the optimizer twist freedom but cap extremes for manufacturability.

Putting it together for our project

- We score candidates with Breguet range using constant TSFC and the trimmed L/D from low-order aero.
- Drag build-up gives D and therefore the fuel burn rate through dW/dt = −c_T * D and dW/ds = −(c_T/V)*D, which is the path-aware view behind Breguet.
- Raymer's wing weight feeds W_ZFW, reducing available fuel or increasing gross weight, so aggressive AR or sweep only "wins" if L/D gains beat the structural penalty.
- Geometry relations from the drawings give lambda, sweeps, MAC, and twist entries for the optimizer without needing the classified or unavailable full dimensions.

MATLAB WORK

10/16/2025

Randal Richards

```
function [state, options, optchanged] = saveGAHistory(options, state, flag)
% saveGAHistory - An output function to log the entire history of a GA run.

persistent all_designs all_scores

optchanged = false;

switch flag
    case 'init'
        % Pre-allocate space for efficiency
        total_individuals = options.PopulationSize * options.MaxGenerations;
        all_designs = zeros(total_individuals, length(state.Population(1,:)));
        all_scores  = zeros(total_individuals, 1);

    case 'iter'
        % Get current generation's data
        current_pop = state.Population;
        current_scores = state.Score;

        % Calculate start and end indices for this generation
        start_idx = (state.Generation - 1) * options.PopulationSize + 1;
        end_idx = state.Generation * options.PopulationSize;

        % Store the data
        all_designs(start_idx:end_idx, :) = current_pop;
        all_scores(start_idx:end_idx) = current_scores;

    case 'done'
        % Truncate any unused pre-allocated space
        % This correctly handles early stopping from MaxStallGenerations
        last_idx = state.Generation * options.PopulationSize;
        all_designs = all_designs(1:last_idx, :);
        all_scores  = all_scores(1:last_idx, :);

        % Assign the final history to the base workspace for analysis
        assignin('base', 'ga_history_designs', all_designs);
        assignin('base', 'ga_history_scores', all_scores);

        % Clean up persistent variables
        clear all_designs all_scores
end
end
```

Overview:

- MATLAB work is going great; Cooper is leading the charge in terms of the full optimization plan for the MATLAB to work on. As an inexperienced user, I have taken the backseat so far, but I will be making the Discrete Range Equation for the MATLAB code to use so that it has proper steps for fuel usage. So far, I will just be reading the code and analyzing it to see improvements. With this, I will also be highlighting key places where I can input my own code so that we can have the Discrete Range Equation with no errors.

Bi-Weekly Meeting

10/27/2025

Overview:

- We had the nice change today to see Mr. Seay's face for the first time in live action. Great surprise here. Meeting was discussing the range equation while adding in the steps for the fuel usage at a select altitude and Mach speed. The constraints we have are the weights of the aircraft, such as the gross, dry, and empty weights. These are important for the later Reymer equations.

Research: MATLAB Code and Work

10/15-10/30/2025

Old Code:

Overview:

The old code evaluates range with the classic constant-C Breguet shortcut. It calls runWingSimulation(x) to get each candidate's aerodynamics and weights, notably L/D_new and WZFW_new), then back-calculates a single lumped factor $C=V/c$ from a fixed baseline range. With that fixed C, the objective computes range increase vs. baseline using a single log of the endpoint weights, i.e., $R_{new}=C\,(L/D)_{new}(\ln(Wi/Wf))$, and returns negative range increase so GA can minimize it. In this mode the cruise speed and TSFC are implicitly constant across all designs, all nominal fuel is burned, and tank capacity/reserve aren't enforced inside the objective.

Because it's closed-form, the old path is fast and great for early trades. You can sweep geometry, see how *L/D* and ZFW shifts move the log term, and quickly rank designs by marginal range gain. The GA infrastructure around it—WingOptimizer (driver), saveGAHistory (complete population logging), Excel export of Top-100 and diverse k-means picks, and the carpet/parallel-coordinates plots—already gives you robust design-space visibility. That makes it a solid sanity-check harness: run big populations, confirm trend directions, and identify promising regions to tighten with higher-fidelity assumptions.

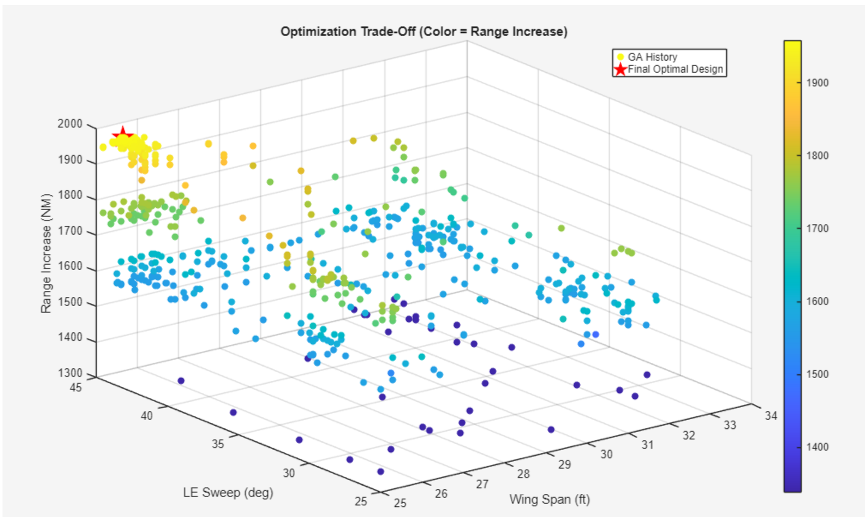
Going forward, we are keeping the old mode as a benchmark and regression test and will use it alongside the stepwise loop.

- We will use it to set upper-bound expectations for what a geometry could deliver if cruise policies were idealized
- Do parameter calibration
    - Tweak C until the old model matches a known mission point, then compare the stepwise result with capacity/reserve on
- Generate initial seeds for refined runs. This two-track workflow preserves the speed and clarity of the old code for exploration while the new stepwise objective handles realism and final selection.

Discussion:

Randal Richards

- The old code was very good. It ran multiple generations with a back-calculated C value that made the Breguet Range equation very simplistic and basically constant. With this C, the speed and fuel specific consumption is captured inside of the variable. The back-calculation is made because we weren't explicitly modeling V and c. Instead we were using it from the baseline that we had made.
- After optimizing the changes to the twist, span, sweep, and other variables it gives a code for maximum range increase. This is very important because it gives us a baseline for the best geometries to maximize the total range of the 3000 NM flight that we were given.



- The plot above shows the Range Increase (NM) and has values ranging from ~1300-2000. Basically, we have J = - (RangeNew – RangeBase) This is great for us because not only did it show us that the optimizations for each generation worked, but it also showed us that we can get variation in the span, sweep, L/D, fuel capacity, AoA, and Wing weight which is exactly what we were looking for.

Supervisor Meeting

11/03/2025

Overview:

- We showed Dr. Banu our progress as a group and our OpenVSP code so that we can directly link the high-fidelity model to our optimization process for maximizing range. We began talking about the next steps and each members contributions to the project.

Ideation

Randal Richards

Concept #: 1

Concept Name: OpenVSP Parametric Wing Generator + Batch Evaluator

DESCRIBE THE CONCEPT (provide a thorough description of the concept that includes design features/elements, capabilities, etc.). You may provide a sketch of the concept on an additional page.

A Python-driven pipeline that uses the OpenVSP API to procedurally generate parametric wing geometries (span, taper, sweep, dihedral, twist distribution, and airfoil families). For each geometry, the tool auto-meshes and evaluates aero performance via low-fidelity solvers (XFOIL for section polars, VLM for 3D lift/induced drag, empirical wave drag at transonic Mach). The user can lock certain parameters (e.g., span or L/D target) and choose an objective (maximize range, maximize endurance, minimize drag at a mission point). The pipeline outputs CSVs, plots, and native VSP3 files ready for handoff to higher-fidelity CFD or structural tools.

STAKEHOLDER NEEDS (What are the stakeholder needs that this concept addresses?)

Addresses the need for a fast, parametric concept generator that ties directly to a geometry standard used across LM/academia for downstream analyses. Enables designers to scan planform and twist spaces quickly before committing compute budget to CFD. Supports low-overhead iteration during early trades and provides consistent geometry artifacts for multi-discipline teams.

ORGANIZATIONAL AND STRATEGIC VALUE/OPPORTUNITIES (What are the potential achievements/ reward/improvements, return on investment, revenue/profit, and innovation value of the concept?)

Accelerates early-phase convergence and reduces costly CFD iteration loops. Improves organizational knowledge re: planform/twist sensitivities across missions. Reusable scripts/templates form a library of configurations and results, increasing ROI with each program. Enables rapid A/B testing for sponsor questions and enhances proposal agility.

POTENTIAL COMPETITION/RISKS (What are the risks to the client and other stakeholders with this concept? What are the technical design/production risks (cost, IP, practicality) of this concept?

Randal Richards

Risk of over-reliance on low-fidelity predictions (e.g., shock drag, interference) and optimizer bias to mesh/parameterization. OpenVSP or XFOIL/VLM limitations at certain Reynolds/Mach regimes. IP risk minimal but requires proper model governance for export-controlled data. Practical risks include maintaining software versions and ensuring mesh/solver robustness across wide geometry ranges.

ADVANTAGES/DISADVANTAGES OF CONCEPT (What are the Adv/Disadv for the client if they were to select this concept?)

Advantages: direct CAD tie-in, reproducible batch studies, fast iteration, clean handoff to CFD. Disadvantages: limited fidelity for complex flow features and structural loads; requires validation and guardrails to avoid misleading optima.

BROADER IMPACT (What are the global, economic, environmental and societal implications of this concept?)

Potential to steer early wing concepts toward lower fuel burn and emissions by filtering poor performers before expensive CFD/wind-tunnel steps. Globally, faster iteration can shorten development cycles and promote energy-efficient designs, with positive economic and environmental effects.

Concept #: 2

Concept Name: Airfoil + Twist Optimization Toolkit (XFOIL/VLM + GA/CMA-ES)

DESCRIBE THE CONCEPT (provide a thorough description of the concept that includes design features/elements, capabilities, etc.). You may provide a sketch of the concept on an additional page.

A focused optimizer that treats spanwise airfoil selection and twist distribution as the primary design variables. Section polars are computed via XFOIL (or a precomputed database), then assembled into a lifting-line/VLM model for 3D performance. Metaheuristics (GA or CMA-ES) handle discontinuous design spaces and constraints (e.g., tip twist bounds, root thickness, stall margins). Objectives include maximizing L/D at specified α/M/weight, or maximizing Breguet range/endurance under fixed fuel or takeoff weight. Users can pin certain stations, enforce manufacturability (limited airfoil families), and export optimal station data for CAD/CFD.

Randal Richards

STAKEHOLDER NEEDS (What are the stakeholder needs that this concept addresses?)

Directly addresses early aero-performance shaping for sponsors focused on camber/twist sensitivity. Provides a systematic, repeatable way to create station-by-station prescriptions that downstream disciplines can accept without major rework.

ORGANIZATIONAL AND STRATEGIC VALUE/OPPORTUNITIES (What are the potential achievements/ reward/improvements, return on investment, revenue/profit, and innovation value of the concept?)

Improves turnaround on conceptual trades, enabling quick what-ifs for mission changes. Increases reuse across programs because the station framework generalizes to many wings. Potential revenue impact via faster proposals and more competitive performance estimates.

POTENTIAL COMPETITION/RISKS (What are the risks to the client and other stakeholders with this concept? What are the technical design/production risks (cost, IP, practicality) of this concept?

Susceptible to local minima if initialization is narrow; XFOIL limits at high-Re adverse pressure gradients; VLM assumptions miss viscous/3D separation. Requires careful constraint/specification to avoid non-buildable twist/airfoil combinations. Tooling risks: dependency on operating-system builds of XFOIL and Python packages.

ADVANTAGES/DISADVANTAGES OF CONCEPT (What are the Adv/Disadv for the client if they were to select this concept?)

Advantages: targeted to the highest-leverage variables (airfoil & twist); robust metaheuristics; easily coupled to structural weight proxies. Disadvantages: still low-fidelity and must be validated; not a full aero-structures co-optimization out of the box.

BROADER IMPACT (What are the global, economic, environmental and societal implications of this concept?)

Better early twist/airfoil tailoring can yield lower drag and fuel burn, reducing operating costs and emissions. Toolkit can be shared across academic/industry collaborations, broadening access to efficient design practices.

Randal Richards

Concept #: 3

Concept Name: Surrogate-Assisted Wing Optimizer (GP/RF/NN on VLM/XFOIL data)

DESCRIBE THE CONCEPT (provide a thorough description of the concept that includes design features/elements, capabilities, etc.). You may provide a sketch of the concept on an additional page.

A two-stage approach: (1) Design of Experiments (Latin Hypercube) samples the wing design space (span, taper, sweep, twist, airfoil family). Each sample is evaluated with XFOIL+VLM to collect metrics (CL, CD, L/D, α for trim, estimated range). (2) Train a surrogate model (Gaussian Process / Random Forest / lightweight NN) to predict objectives/constraints cheaply. The optimizer then runs on the surrogate with periodic truth-model checks to avoid drift. Supports multi-objective Pareto front generation and exports candidate sets for CFD.

STAKEHOLDER NEEDS (What are the stakeholder needs that this concept addresses?)

Addresses stakeholder need to explore very large design spaces quickly while preserving a link to underlying physics. Useful when turn-time is critical (design reviews, proposal deadlines) but a physics backbone is still desired.

ORGANIZATIONAL AND STRATEGIC VALUE/OPPORTUNITIES (What are the potential achievements/ reward/improvements, return on investment, revenue/profit, and innovation value of the concept?)

High leverage: once a surrogate is trained, exploration and sensitivity mapping are near-instant, enabling interactive design sessions. Strategically, it builds institutional datasets and models that grow in value across programs. Potential ROI from reduced compute cost and faster down-select to CFD-worthy concepts.

POTENTIAL COMPETITION/RISKS (What are the risks to the client and other stakeholders with this concept? What are the technical design/production risks (cost, IP, practicality) of this concept?

Model bias if DOE coverage is poor; surrogates can overfit outside their training domain. Requires governance for data versioning and validation (MLOps practices). Practical risks include maintaining reproducibility and documenting limits of applicability for reviewers.

ADVANTAGES/DISADVANTAGES OF CONCEPT (What are the Adv/Disadv for the client if they were to select this concept?)

Advantages: orders-of-magnitude faster inner-loop searches; supports Pareto trade studies and uncertainty sweeps. Disadvantages: extra upfront DOE/training step; requires careful guardrails to prevent extrapolation errors.

BROADER IMPACT (What are the global, economic, environmental and societal implications of this concept?)

Broader impact includes democratizing rapid optimization, enabling smaller teams to participate in efficient aircraft design. Economically, shorter cycles reduce risk; environmentally, better early screening elevates efficient concepts sooner.

Research: MATLAB Discrete Range Equation

11/02-11/10/2025

New Code:

Overview:

- The new range evaluation replaces the single-formula Breguet shortcut with a stepwise Breguet loop that burns fuel in small increments and sums the distance for each increment. Two tiny utilities make this work: breguetStepJet.m computes the range gained for one burn step ΔW, and breguetCruiseLoopJet.m iterates those steps from the initial weight down to a fuel-reserve stop, accumulating range along the way. A third helper, schedules_constant.m, supplies constant schedules for speed V, TSFC c, and L/D (and can later be swapped for variable schedules).
- Inside the optimizer, flightScoreObjective.m now (1) calls runWingSimulation(x) to get the candidate's new L/D, zero-fuel weight, and wing fuel capacity; (2) sets the usable fuel to min(nominal fuel, capacity); (3) enforces a reserve fraction; and (4) calls the stepwise loop to produce absolute range R for that design. The objective returned to GA is simply

-R (GA minimizes), and the nonlinear constraint remains the capacity check: required wing fuel minus capacity ≤ 0.

- On the GA side, nothing else is heavy: we kept the existing driver and logging. The only interface changes were wrappers—fitnessWrapper returns the scalar fitness, and nonlconWrapper returns [c,ceq]—so GA + HybridFcn (fmincon) can call the objective cleanly. The post-processing scripts now plot and export absolute range (NM) from the GA history, not "range increase."

Improvements:

- The new loop plugs into the same simulation backbone we already had. runWingSimulation.m still computes geometry, weights, and aerodynamics; notably it uses getIntegratedProps.m to integrate spanwise thickness and airfoil cℓ, then assembles parasite, induced, and wave drag to form L/D_new . Those outputs flow unchanged into flightScoreObjective.m. We did not disturb the GA scaffolding (WingOptimizer.m), history logger (saveGAHistory.m), or the Excel/plot exports—only the way the range is evaluated.
- What improves is physical fidelity and policy control. The old "constant-C" method implicitly assumed fixed V and c, burned all fuel with no reserve, and effectively ignored tank capacity—so it credited range that may not be flyable. The stepwise loop now honors (a) capacity-limited usable fuel, (b) reserve, and (c) the candidate's own L/D. It also gives you a clean seam to introduce variable schedules later (e.g., V(W), c(M,h), L/D(W)), which the constant-C formula can't represent.

Differences:

- The old results reported very large range increases because they used a back-calculated constant C=V/c, assumed full nominal fuel burn with no reserve, and did not cap fuel by wing capacity; the log term ln(Wi /Wf) was therefore generous, and any optimistic L/D or implicit V/c inflated the answer. The new results report absolute range from a capacity- and reserve-aware step loop, so the log term is smaller (you stop at reserve, and usable fuel may be below nominal), and the range sits near the hundreds of NM with our current V/c, L/D, and capacity. Same aircraft family—different accounting rules—hence the gap.

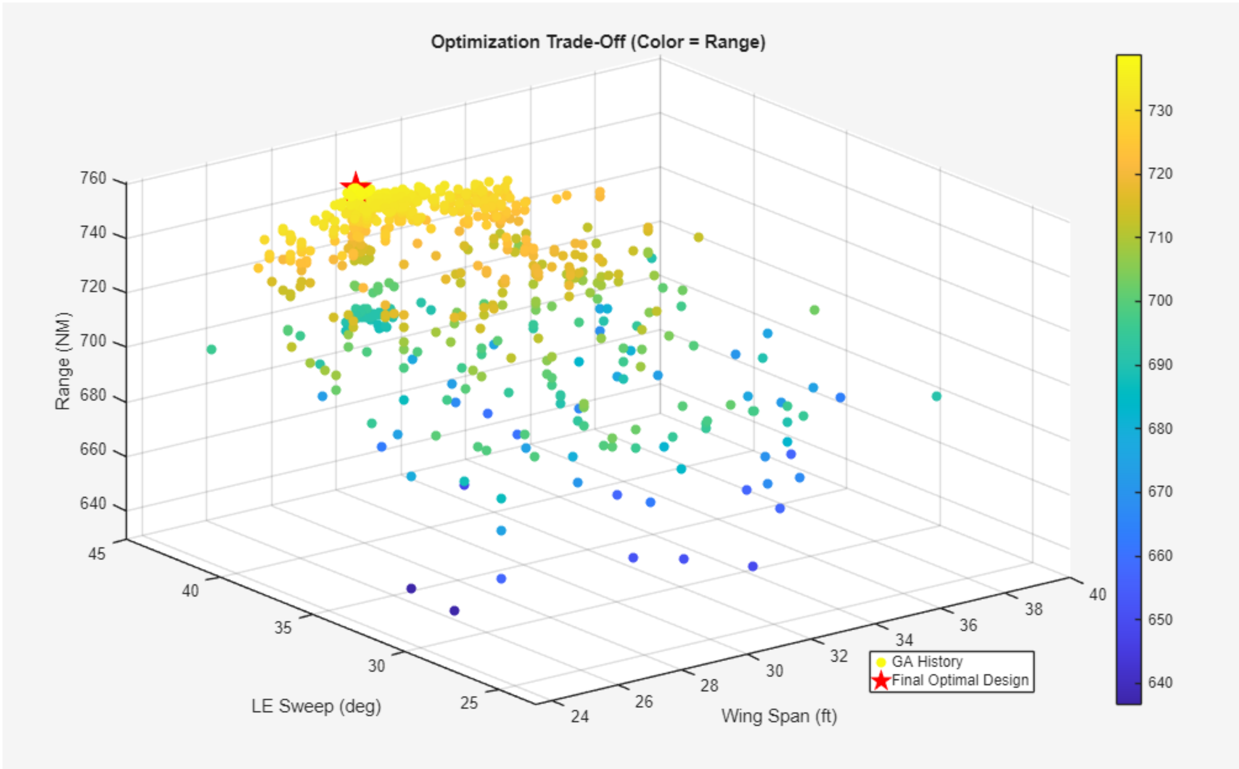
Going forward (pushing towards 3000 NM)

- We'll raise the Breguet product in three coordinated ways:
  - Improve L/D by tightening the drag build-up in runWingSimulation.m
    - Refine Cf , *CD0,other*, wave-drag trigger Mdd, and the off-design "creep" coefficient; also widen the GA bounds for span/taper/twist after VSP validation.

- Increase V/c by adopting realistic cruise schedules (e.g., V↑ at altitude, c↓ with a better TSFC map)
  - Done first via schedules_constant parameters, then via V(W) and c(M,h) tables.
- Grow the log term by boosting usable tank volume
  - Replacing the simple fuel_vol_factor with a VSP-derived wing-box volume and usability factor and right-sizing reserve per mission rules.
  - Concretely: we'll import from OpenVSP the spanwise tank cross-section A-tank (y) → capacity, the measured Swet and MAC/AR → better drag and CL trim, and VSPAERO CL/CD vs *α,* M → a table-driven L/D(W).
  - Then the GA re-optimizes against the stepwise objective, targeting the combined product needed to reach ~3000 NM.

- Altogether, we have a discrete *stepwise* Bregeut loop that uses the fuel burn. It doesn't change the schedules *inside* the step it only keeps them as constant. It's still a lot more realistic than the straight shot constant that we were using for range because it enforces usable fuel, reserve, and updates **weight** each step, which is similar to what our client wanted. Going forward, we are going to try these steps to maximize the output for the range and then go ahead and add a dynamic range equation that can read and adapts to the aircrafts changing state and environment. This will let the optimizer tackle trajectory and geometry together, as opposed to just geometry. This will produce tighter, more credible ranges for a given mission rule set. For this, we will need tables and curves from the VSP for V, c, and L/D.

Results for first Test Run:



Graph:

- Each dot is one design the GA evaluated.
- Axes: X = Wing span (ft), Y = LE sweep (deg), Z = Absolute range (NM) computed with the stepwise Breguet loop.
- Color bar = range (NM) too (warmer = farther).
- Red star = the final optimal design returned by GA (highest range among the population history).

Results:

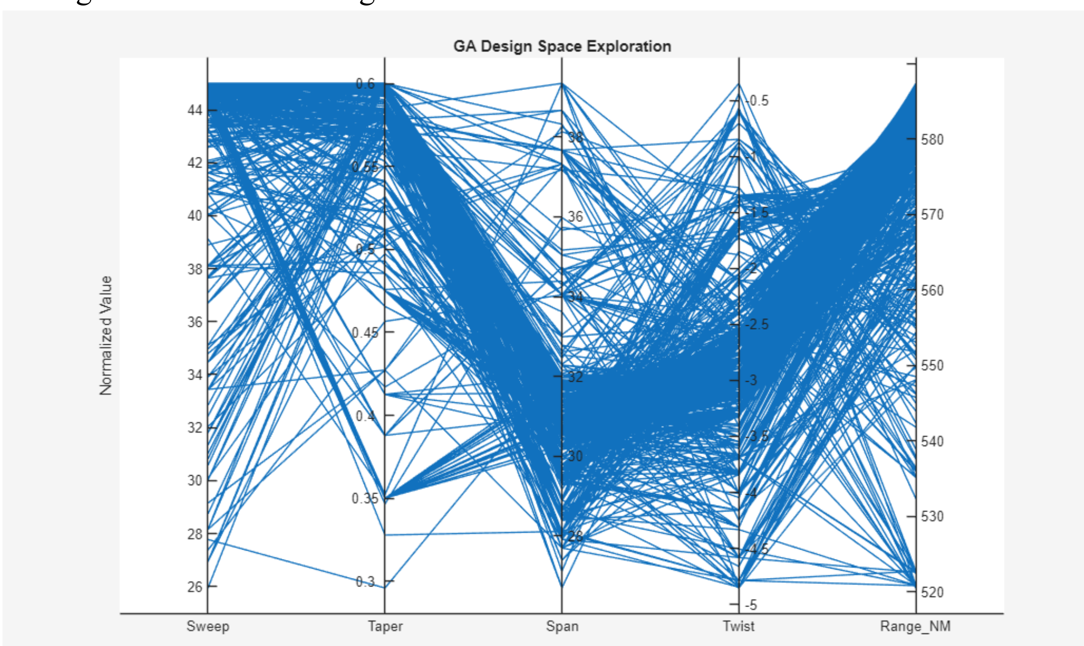
- Best region: ~28–32 ft span and ~36–40° sweep produce the highest ranges (~740–755 NM) in our current model.
- Plateau behavior: Near the top you see a flat "cap"—gains are small there, suggesting we're limited by fuel capacity and/or L/D in our current assumptions (TSFC, reserve policy, etc.).
- Worse regions: Very short spans (<26 ft) and very long spans (>36–38 ft) trend to lower range

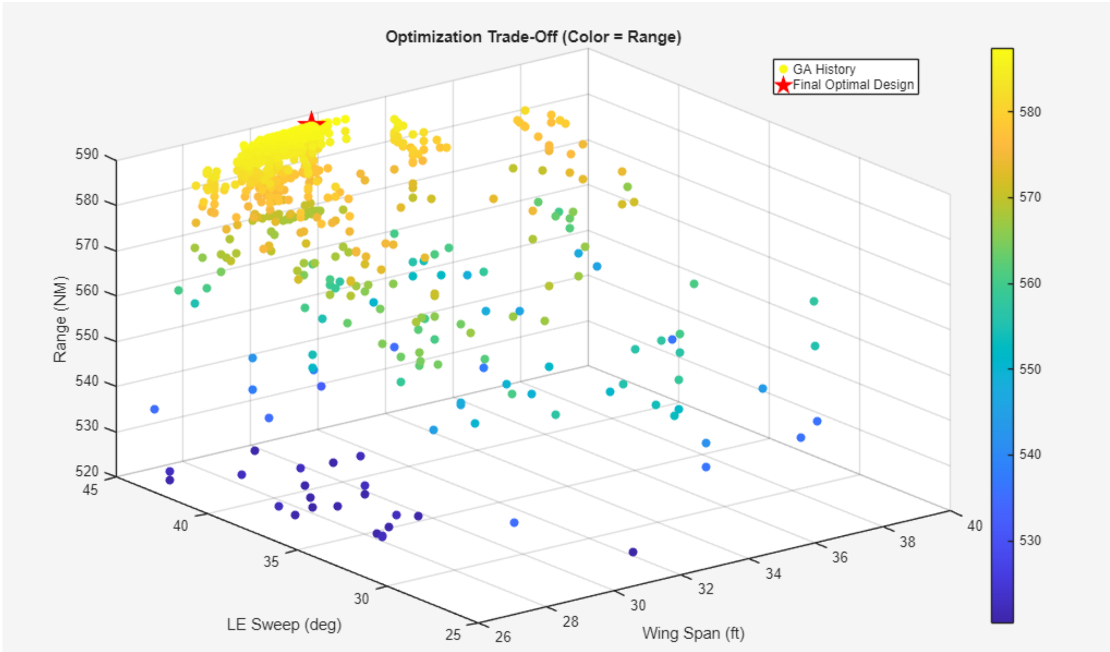
Bi-Weekly Client Meeting

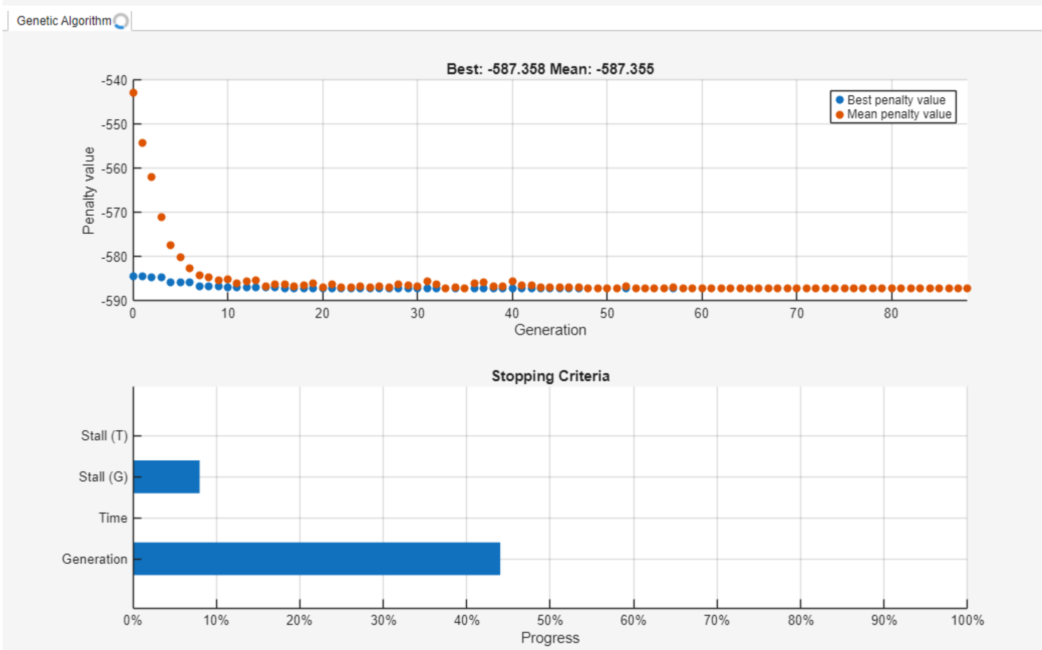
11/10/2025

Overview: Went over everyone's contributions to the project so far.

- I shared the preliminary Discrete Range Equation that I've been working on. I didn't have the full model to test until shortly after the meeting because I had to download 5 different MATLAB extensions but Mr. Seay had enjoyed the dedication and work done on the Range equation that he brought up in the previous meeting.
- Garret went over the Python -> MATLAB -> OpenVSP port that he was working on as well. Utilizing the old range equation from our previous code and showing that the best models were being used in the higher fidelity models and their analysis.
- Martin shared the new OpenVSP file with three different plots that were for C/D and other geometries on the wing.




Optimization Trade-Off (Color = Range)
GA History
Final Optimal Design
Range (NM)
LE Sweep (deg)
Wing Span (ft)


Genetic Algorithm
Best: -587.358 Mean: -587.355
Best penalty value
Mean penalty value
Penalty value
Generation
Stopping Criteria
Stall (T)
Stall (G)
Time
Generation
Progress

SWOT ANALYSIS

11/11/2025

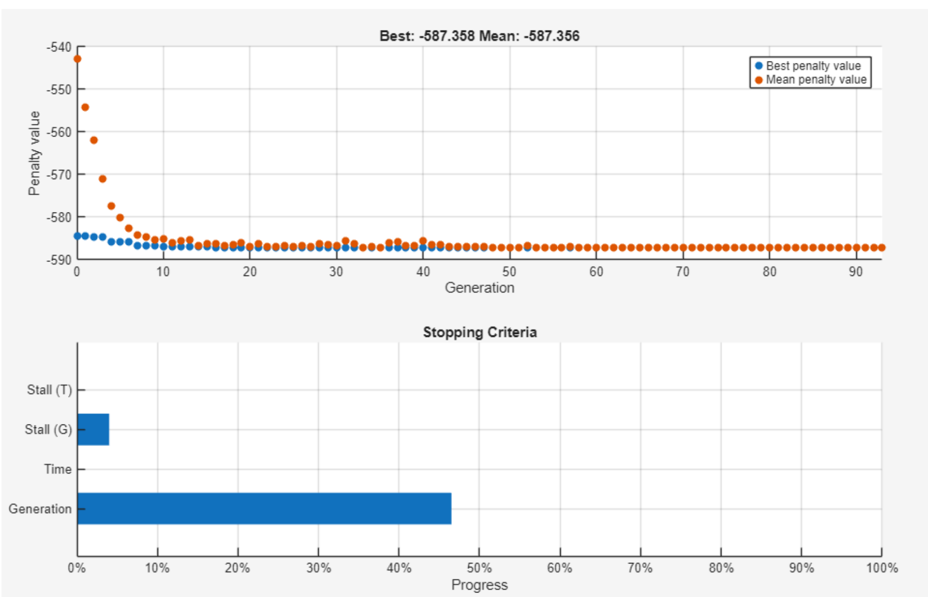
	Strengths	Weaknesses	Opportunities	Threats
1	MATLAB Experience	Weekly meetings	Few optimization workflows	Schoolwork
2	CFD	Ansys Student License	Integration with HPC lab	Modeling inaccuracy
3	Good communication	Proprietary databases	Automation	Hardware limitations
4	Supercomputer access	Lack of accurate measurements	Potential reusability	Holidays
5	Dr. Banu	Broadness of work	Environmental impact reduction	Team absences

Research: Troubleshooting

11/15/2025

Overview: During some of the runs for the new GA optimizer I ran into some problems with the Analysis stopping short. As well as some issues understanding the use of the TSFC tables, Mach, Altitude, and C/D profiles given from OpenVSP.

Analysis stopping short of 100%



- *What happened* – Several GA runs were ending at ~50% of the "Max Generations" bar even though we expected them to run longer. MATLAB reported: "ga stopped because the average change in the penalty function value is less than options.FunctionTolerance and the constraint violation is less than options.ConstraintTolerance."

- *How we figured it out* – Looking at the best/mean penalty history showed that after ~80–100 generations, the curves flattened out and the top ~100 designs were almost identical (design variables differing by ~1e-4). That meant the search had converged in a very narrow feasible region rather than "quitting early by mistake."
- *Fix implemented* – We adjusted the GA settings to encourage more exploration before convergence: increased `PopulationSize`, increased `MaxGenerations` and `MaxStallGenerations`, and reduced the elitism fraction while slightly tightening `FunctionTolerance` and `ConstraintTolerance` to avoid premature termination due to numerical noise. We also sanity-checked that all four design variables (sweep, taper, span, twist) actually affect the outputs in `runWingSimulation`.
- *Future improvements* – Do a quick sensitivity sweep around the current best design every time we change the physics/model, to verify that each variable still has a meaningful impact on range and penalties.
- Log and plot raw range, L/D, and constraint values vs. generation (using the GA history arrays) to see which terms are driving convergence.
- If the landscape genuinely is flat, expand the design space (e.g., allow larger changes in span/area or add shape variables) instead of only tweaking GA knobs.
-
```
ga stopped because the average change in the penalty function value is less than options.FunctionTolerance and
the constraint violation is less than options.ConstraintTolerance.
--- Phase 1 Optimization Complete ---
Best design found (fmincon polished):
  Sweep:   45.00 deg
  Taper:   0.60
  Span:    31.81 ft
  Twist:   -3.00 deg
```
- As for the fix, I had understood that the objective is sensitive and the GA clearly explores the full boundaries of these limits earlier on, so the best range we can do at this time (~580NM) is still clustered in that very small region with modest range improvements.

TSFC Tables

- *What happened* – When we first introduced tabulated TSFC instead of a single constant value, the computed range collapsed from the 2000–3000 nm target down to a few hundred nm. The GA optimized correctly, but it was optimizing a much harsher mission model than we intended.
- *How we figured it out* – Comparing old vs. new runs showed that the only major change was TSFC, and inspecting the table values revealed that we were either:
- pulling the wrong entries (e.g., wrong Mach/altitude index or off-by-one in the table), or
- using TSFC units or scaling inconsistent with the Breguet equation (e.g., lb/lbf-hr vs 1/hr).

Randal Richards

Running a simple "single-point" range calculation with a hand-picked TSFC from the table made the problem obvious: range dropped exactly in proportion to our mistaken TSFC value.

- *Fix implemented* – Standardized TSFC units to match the Breguet implementation and documented that in the code.
- Wrote a clear lookup function that takes Mach and altitude and returns TSFC, with explicit indexing and comments so we don't mix up rows/columns.
- Verified a few sample TSFC values against the original performance charts before letting GA use them.
- *Future improvements* – Add assertions or checks in the TSFC function (e.g., error if TSFC is outside a reasonable range).
- Implement interpolation across Mach and altitude so the GA isn't limited to discrete TSFC grid points.
- Keep a "diagnostic mode" where we fix TSFC to a constant again; if range suddenly changes by an order of magnitude, we know the table hookup is broken.

Mach and Altitude Handling

- *What happened* – Incorporating Mach and altitude from the mission profile into the performance code introduced confusion in how they were used: sometimes they were treated as fixed cruise values, and other times they were passed into the TSFC and aerodynamic models inconsistently. This led to mismatches between the assumed flight condition and the data pulled from the tables.
- *How we figured it out* – We compared the mission step loop with the TSFC and aero lookups and noticed cases where, for example, Mach was updated but altitude was not (or vice versa), or we were using sea-level properties with high-altitude Mach numbers. Running a few time-step prints (Mach, h, TSFC, CL, CD) showed the inconsistency directly.
- *Fix implemented* –
    - Defined a single "mission state" structure for each step that includes Mach, altitude, weight, and other needed quantities, and passed that consistently into the TSFC and C/D lookup functions.
    - Locked in a clear definition of the design point (cruise Mach and altitude) and ensured GA optimizes around that point while still allowing the mission script to reference it correctly.
    - Verified that atmospheric properties (density, speed of sound) are computed at the same altitude that we use for Mach and TSFC.
- *Future improvements* –
    - Add plotting of Mach and altitude vs. mission step alongside TSFC, CL, and CD to quickly spot inconsistencies.

   - If we later allow GA to vary Mach or altitude, keep those as explicit design variables with tight bounds and document how they feed into both TSFC and aero models.

C/D Profiles from OpenVSP / VSPAero

- *What happened* – After importing CL/CD (or C/D) profiles from OpenVSP/VSPAero, the optimized ranges dropped significantly and the GA solutions clustered tightly. The issue wasn't in GA itself; it was the way we were using and scaling the polar data and how we tied it back into the mission and weight model.
- *How we figured it out* – We compared:
   - the analytical/approximate drag model vs. the interpolated VSPAero polar at the same Mach, altitude, and CL,
   - and saw that the VSPAero-based CD (or C/D) was much higher than the simplified model at cruise.
     We also checked how the tables were indexed: in a few early runs, CL or angle of attack indices were mismatched. When we manually queried the tables at the best GA design, the resulting CL/CD matched the low range we were seeing.
- *Fix implemented* –
   - Cleaned up the polar data structure and wrote a dedicated interpolation function `getAeroCoeffs(Mach, alpha or CL)` so the indexing logic is in one place, well-commented, and tested.
   - Verified a few points against VSPAero's own report to ensure we're pulling the correct CL and CD.
   - Checked that the reference area, reference span, and units used in VSPAero match those in our MATLAB performance model so CD isn't artificially inflated.
- *Future improvements* –
   - Build a small validation script that sweeps AoA or CL at a single Mach/altitude and plots our interpolated polar vs. the original VSPAero output. Run this whenever we regenerate or modify the polars.
   - Consider using different polars for different wing configurations or adding correction factors if we know VSPAero's setup (mesh, turbulence model, etc.) is conservative.
   - Eventually connect the GA to re-generate or at least re-query the VSPAero polar for promising design points rather than assuming a fixed polar for all geometries.

Bi-Weekly Supervisor Meeting:

11/18/2025

Overview:

Randal Richards

Shared CFD Results / Next Steps

- CFD check-in with Dr. Banu
  - She said the results look good as a first pass, but we should align our setup with Lockheed's exact parameters before using them as final.
- Email to Mr. Seay + re-run CFD
  - Email Mr. Seay asking for the full CFD/mission parameters (Mach, altitude, Re, ref area/chord, BCs, turbulence model).
  - Update our case with those values and re-run CFD to create a Lockheed-consistent "anchor" case for our workflow.
- More OpenVSP wing designs
  - Start generating a small family of parametric wing variants in OpenVSP based on our GA variables (sweep, taper, span, twist).
  - For each, run VSPAero and pull updated polars so the MATLAB optimizer is using aero data that matches our evolving geometries.

Milestone #6: Project Presentation and Tables

- Evidence of the development of QFD, FMEA, decision matrices Below

Decision Matrix (5 = Strongest, 1 = Weakest)

Criteria	Hierarchical Open-Source Pipeline	Comparative Optimization Model Analysis	Interactive Design Space Explorer
Fidelity Range (30%)	5→1.5	4→1.2	3→0.9
Computational Efficiency (25%)	4→1	3→0.75	2→0.5
Discoverability of Optimum (25%)	4→1	5→1.25	2→0.5
Usability (10%)	3→0.3	3→0.3	5→0.5
Scalability (10%)	4→0.4	4→0.4	2→0.4
Total	4.2	3.9	2.8

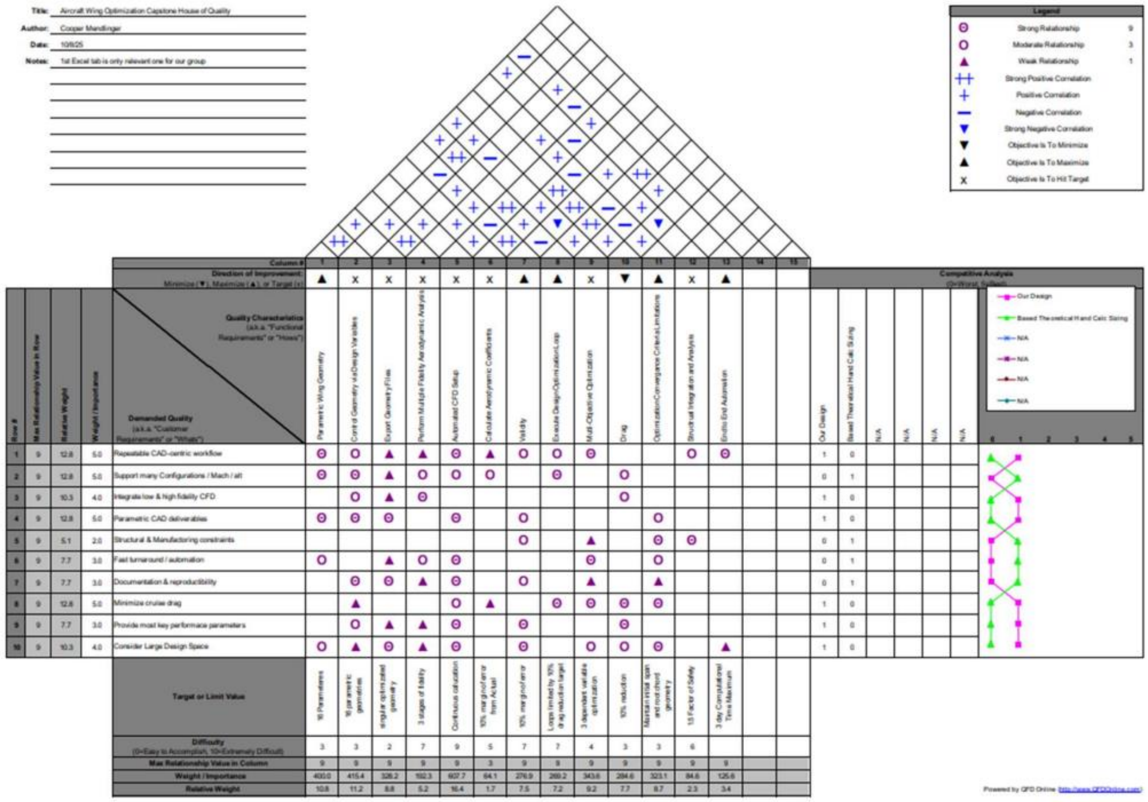
College of Engineering
UNIVERSITY OF GEORGIA

F25 - Lockheed Martin - Aircraft Wing Design Optimization 15

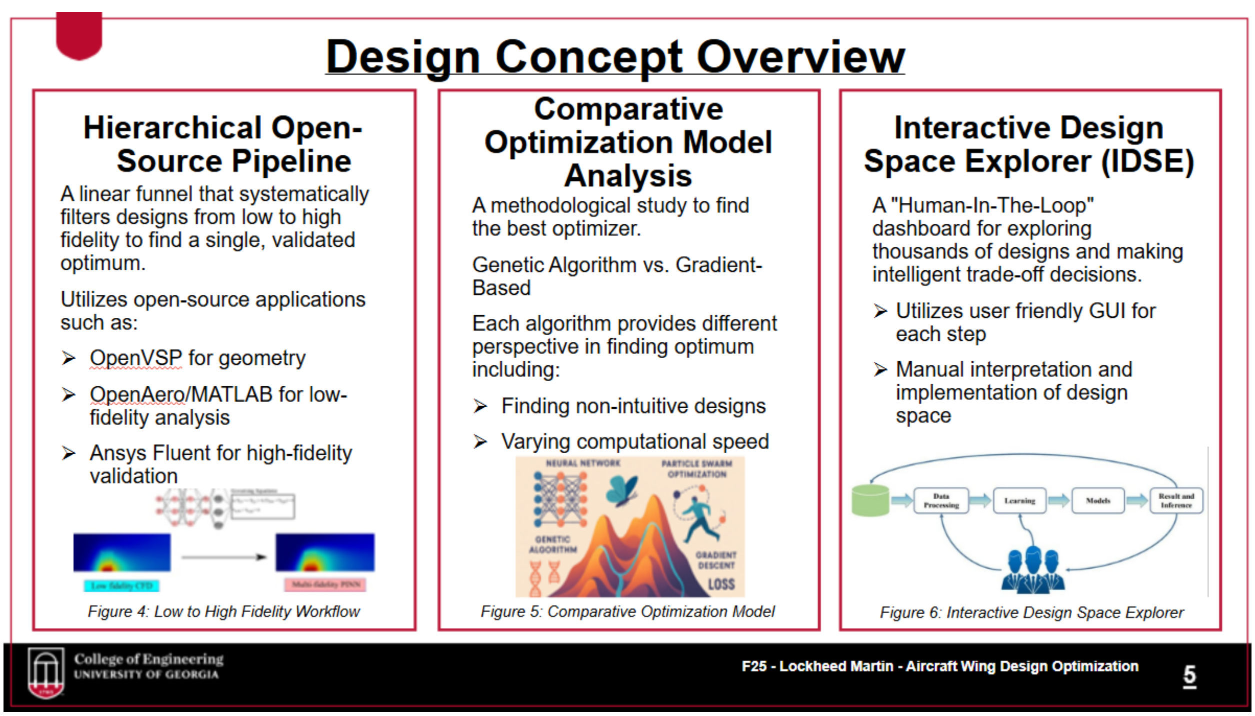
FMEA Matrix

Failure Mode and Effects Analysis (FMEA)

System / Item / Process Step	Potential Failure Mode	Potential Failure Effects	Severity	Causes	Occurrence	Current Controls	Detection	RPN	Recommended Actions
Name, ID number, etc.	How could the system / item / process potentially fail?	Consequential impact on other systems, departments, etc.		All contributing factors		Prevention / Detection		Risk Priority Number	Steps required to reduce severity, occurrence, and detection
MATLAB – optimizer configuration	Poor optimizer settings (step size, tolerance, max iterations)	Premature convergence or non-convergence; sub-optimal wing passed forward as "best" design	6	Default settings used blindly, insufficient iteration budget, non-convex design landscape	5	Visual inspection of convergence history; limited sensitivity study	5	150	Document default vs tuned settings, perform structured sensitivity study on key parameters, encode recommended settings in a config file
OpenVSP geometry generation	OpenVSP creates geometries with self-intersections or gaps	CFD meshing fails or produces highly skewed elements, results unreliable or simulations crash	7	Extreme parameter combinations, inadequate surface tolerances, poor trimming	5	Occasional manual geometry cleanup	4	140	Add geometry quality script (check intersections, sharp angles), define safe parameter limits, reject designs that violate quality metrics
CFD Case setup	Mis-specified boundary conditions (freestream, symmetry, far-field, wall or reference area)	Mis-predicted loads and performance metrics; direct impact on mission fuel burn/range estimation	9	Copy-paste of case files; inconsistent reference area vs optimizer, wrong Mach/Re input	3	Template case files; spot check against handbook coefficients	6	162	Create a setup checklist, script all boundary condition and reference values from a single source of truth



Evidence of concept developments, use of engineering tools to evaluate ideas



Feedback and reflection from the previous presentation to the supervisor/client:

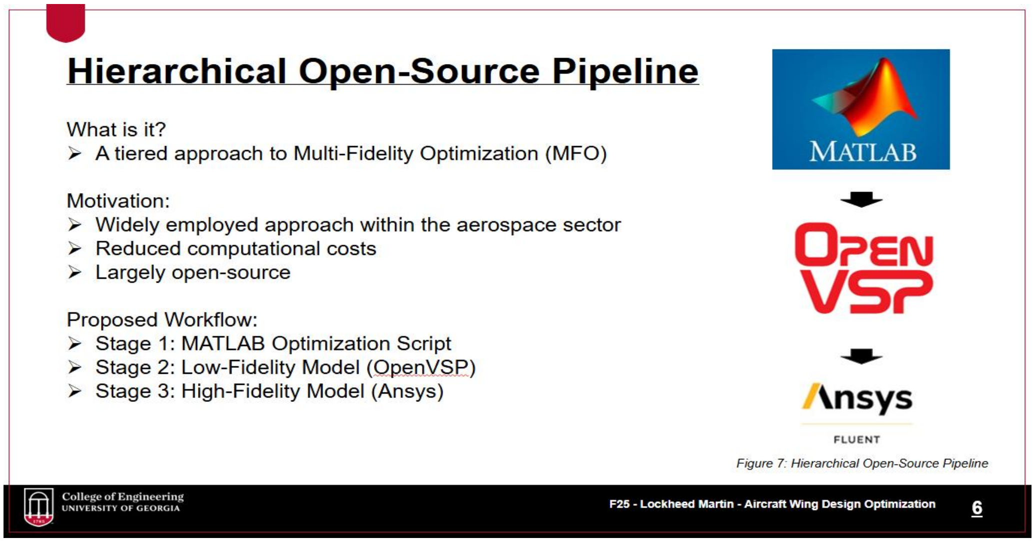
Randal Richards

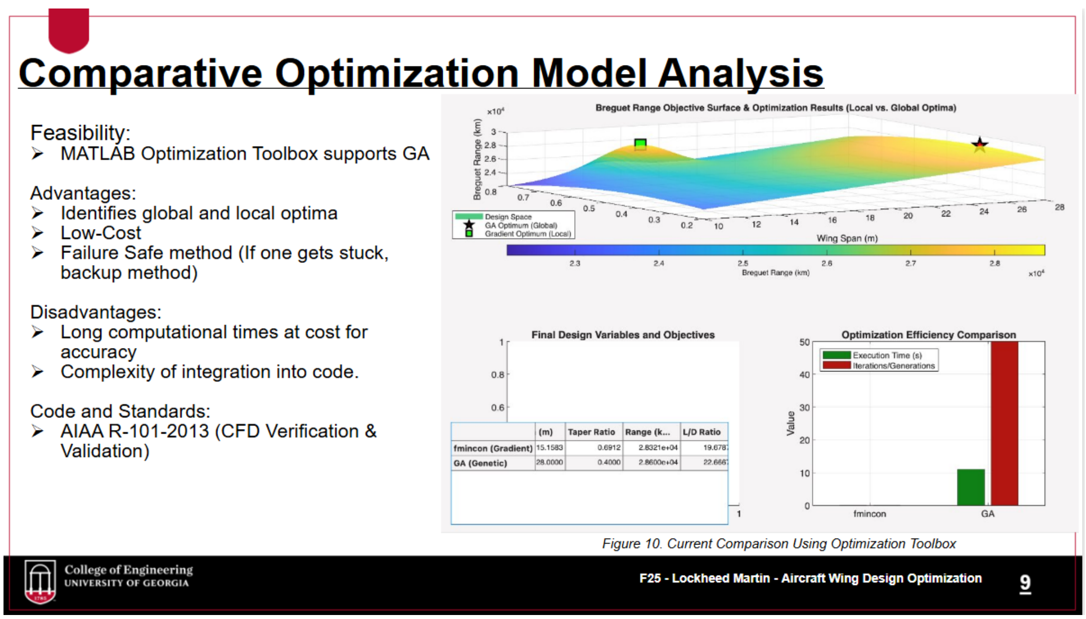
- Added more quantifiable criteria in this presentation so our concepts are supported by clearer performance metrics instead of only qualitative descriptions.
- Carefully checked formatting and typos to make the slides more professional and easier for the sponsor and supervisor to follow.
- Refined and improved monetary estimates so the cost side of our concepts is more realistic and defensible.
- Added the presentation date on the introduction slide for better documentation and version tracking.
- Made a point to repeat questions during the presentation before answering them, to ensure everyone heard the question, to buy a second to think, and to give more focused, relevant responses.

Milestone #7: Presentation Work

Overview: Presentation Contributions

- Contributed significantly to the overall presentation by helping structure the slide deck, refining technical content for clarity, and making sure our key objectives, methods, and results were communicated in a way that both faculty and industry could follow.
- Carefully reviewed peer evaluations from earlier presentations, using that feedback to fix pacing, improve slide readability, and add more concrete numbers/criteria so the story of our project felt more polished and professional.
- Helped deliver a strong presentation to our Lockheed Martin sponsors, responding clearly to their questions, highlighting the real-world relevance of our workflow, and contributing to a positive overall impression that the sponsors said aligned well with their expectations.





Multi-Fidelity Workflow Clarified

- We defined a structured path from CAD in OpenVSP through low-fidelity aerodynamics and into higher-fidelity CFD, so each change to wing geometry has a clear place in the analysis chain.

Dynamic Range Equation Centralized

- We recognized that our range equation must act as the main performance metric, linking TSFC, lift-to-drag ratio, and weight changes so every design update is evaluated on its impact on range.

Higher-Fidelity Data Integration

Randal Richards

- We saw that coarse aerodynamic estimates were suppressing range, so we are now feeding VSPAERO CL–CD polars and drag components directly into MATLAB to drive a more realistic range calculation.

GA Design Space Control

- We observed that early genetic algorithm runs tended to drift toward unrealistic designs, so we are tightening variable bounds and adding penalty structures to keep solutions within feasible geometric and structural limits.

Optimizer Role Definition

- We concluded that GA is best for broad exploration while gradient-style methods are better for local refinement, and we plan to log convergence behaviour to demonstrate how these approaches complement each other.

Mid-Semester Action Plan

- We are prioritizing a stable automated loop between MATLAB and OpenVSP, locking in a reliable GA configuration tuned to our parameters, and producing mid-semester results that show both an improved wing and a reusable optimization workflow.

Research for MATLAB: Finalization

12/1/2025

Process & Research:

Range equation process I developed

- I started from the classical Breguet range equation and rewrote it into a MATLAB function that works directly with my XQ-58A baseline. I defined a clear state vector with takeoff weight, fuel weight, zero fuel weight, and aerodynamic performance so that each evaluation of the range is consistent with the aircraft geometry and mission assumptions.
- I then embedded that range model inside my existing MATLAB optimization framework. The genetic algorithm calls a wrapper function that sends a design vector (sweep, taper, span, twist) into my analysis script, which computes L/D, wing weight, updated zero fuel weight, and finally the range. The optimizer only sees the negative of the range increase, so all of the aircraft physics stay inside my analysis code.
- After seeing the range collapse from around 3700 NM (with very optimistic assumptions) to about 600 NM once I used more realistic aerodynamics, I began building a “dynamic” range equation. In this version I step through the mission with small fuel burn increments, update the weight each step, and use aerodynamic and TSFC data from tables instead of fixed constants. This is where the OpenVSP polar files and the new MATLAB functions I wrote become essential.

Independent research and forward progress toward 3000 NM

- I researched how to move from a single L/D value in the Breguet equation to using full CL–CD polars from OpenVSP's VSPAERO module. I learned how to generate .polar files, how the lift and drag coefficients vary with angle of attack, and how to export those results. I then wrote a MATLAB reader that loads the .polar file, parses the columns, and interpolates CL and CD for the specific lift coefficient my mission needs. This lets my range code use a realistic L/D that changes with angle of attack instead of a fixed "best" L/D, which was one of the main sources of over-optimism in my initial range estimates.
- I dug into how TSFC and weight evolution affect range predictions when I use a stepwise method instead of the closed-form Breguet solution. Earlier I was holding TSFC and L/D constant and simply applying a single weight ratio, which made the optimizer "think" it had much more freedom than it actually does. I then explored using a constant TSFC that represents a realistic cruise value and coupling it with a one percent fuel burn step model that updates the aircraft weight each iteration. I adapted my MATLAB code so that each loop iteration updates weight, recomputes L/D from the polar data, and accumulates range. This research showed me exactly why the range dropped so much when I put in more realistic physics, and it gave me a path to recover range through better wing designs rather than unrealistic assumptions.
- I spent time understanding how to connect the MATLAB–Python–OpenVSP loop so that the optimizer can "talk" directly to the geometry and aerodynamic tools without me manually exporting and importing files. I read into how OpenVSP's analysis manager structures results, what the .slc and. polar files actually contain, and how to call these analyses programmatically. Based on that, I designed a workflow where MATLAB sends geometry parameters to Python, Python edits the VSP model, runs VSPAERO, and then returns the updated polar data back to MATLAB. I began implementing the MATLAB functions that expect a new polar on each design evaluation, so that the range equation always uses aerodynamic data that matches the current wing geometry.
- I also researched how to keep the genetic algorithm from "cheating" the problem by wandering into nonphysical regions of the design space just to inflate range. I studied GA constraint handling, penalty functions, and variable scaling, then adjusted my optimizer settings and bounds so that span, taper, sweep, and twist stay inside what is structurally and operationally realistic for the Valkyrie. In the code I tightened variable limits, added penalties for designs that violate geometric or performance constraints, and tuned population size and mutation rates. This work increased my confidence that any future ranges approaching 3000 NM come from legitimate aerodynamic and weight improvements, not from the optimizer exploiting unrealistic corners of the design space.
- Finally, I used the new tools I wrote to explore how sensitive the dynamic range output is to each design variable and to the input data quality. I ran controlled sweeps in MATLAB where I changed one variable at a time, using the same polar and TSFC data, to see which

parameters actually move the range in a meaningful way. I compared results from my early "single L/D" model with the new polar-based, stepwise model to see how much each piece of added fidelity changed the answer. This research guided my next steps: focusing effort on variables that truly impact range, improving the fidelity of the CL–CD table and TSFC inputs, and using the GA in a more targeted way to search for designs that can realistically push the Valkyrie's predicted range back toward the 3000 NM goal.

Next Steps:

Interest in adding mission segments to the optimization for next iterations

- The next step I am interested in is breaking the mission into separate segments instead of treating the entire flight as a single cruise block. In practice, the Valkyrie mission contains climb, possibly multiple cruise or dash segments at different Mach numbers and altitudes, maybe a loiter or on station period, and then descent. Each of those segments experiences different TSFC, different aerodynamic efficiency, and different weight evolution. By using my one percent fuel burn loop inside clearly defined mission segments, I can assign different CL CD targets and TSFC values to each phase rather than forcing everything into one average cruise point.
- In code, this means defining a mission profile as a list of segments, each with its own speed, altitude, target Mach, and required time or distance. The optimizer would still choose geometry parameters like sweep, taper, span, and twist, but the evaluation function would step through climb, level cruise, loiter, and descent, updating weight along the way. For example, I could use one polar for climb and another for high speed cruise if needed, while keeping TSFC realistic and limiting the total fuel burn to the available 1440 lb. The output of that loop would be a mission range and fuel margin, which can be checked against the 3000 NM target and any reserve requirements.
- Structuring the problem this way makes the optimization more honest. Instead of the GA chasing a design that only looks good at one idealized cruise condition, it will have to find a wing that supports a full mission. That allows me to add constraints at the segment level, such as minimum climb gradients, acceptable cruise lift coefficients, or limits on angle of attack. It also sets me up for multi point objectives later, where I can balance total range, fuel usage in each segment, and maybe even robustness to changes in weight or TSFC, all while staying tied to the real mission profile Lockheed cares about.

GA tuning and parameter refinement

- I have also been focusing on tuning the genetic algorithm so it does not run into logical issues or game the problem. Early on, the GA tried to push span, sweep, and taper to unrealistic extremes just to improve the simplified range equation. After reviewing constraint handling and variable scaling, I tightened the bounds on sweep at the leading edge, taper ratio, span extension, and twist so that they stay in a realistic neighborhood for the XQ 58A wing and still respect structural and handling considerations. In the code I use penalty functions to push back on designs that break geometric or performance constraints, so high range scores only appear when designs are both feasible and effective.

- On the objective side, I reworked the flightScoreObjective wrapper so it returns the negative of the range increase computed through the dynamic, segment ready range loop. Inside that function I call the main analysis that uses OpenVSP polars, TSFC around 0.55, and the one percent fuel weight update. I also experimented with GA settings like population size, crossover fraction, mutation rate, and stopping criteria to balance exploration and convergence. The goal is to give the GA enough freedom to discover non intuitive sweep and twist combinations but not so much freedom that it drifts into nonphysical or numerically unstable corners of the design space.
- Finally, I started running controlled sensitivity studies using the GA framework. I hold the polar and TSFC data fixed and vary one design variable at a time to see how much it moves the dynamic range output. That helps me understand which parameters the optimizer is actually using to approach the 3000 NM target and which ones are mostly noise. Based on those results, I can refine the bounds and even rethink how many design variables I expose to the GA at once. This combination of cleaner mission modeling, polar based aerodynamics, and tuned GA settings should help ensure that any future range improvements come from meaningful physics and credible geometry changes rather than from numerical loopholes in the optimizer.

Conclusion:

Overall, I feel like this semester pulled a lot of separate pieces together into a coherent workflow. In class, I moved from treating the range equation as a theoretical concept to actually implementing and debugging it in MATLAB for a specific aircraft with real geometry and weight numbers. The presentations forced me to explain that progression clearly: how the early “optimistic” Breguet model broke down once I added realistic aerodynamics, and how that pushed me toward a more dynamic, stepwise range calculation. Each check-in with our supervisor and sponsor made me tighten the assumptions, clean up the plots, and clarify what the numbers really mean for a 3000 NM mission, instead of just chasing big range values.

On the coding side, I now have a functioning backbone that links the range equation to design variables and optimization. I built and refined scripts that take sweep, taper, span, and twist as inputs, compute L/D and weight changes, and feed the result into a genetic algorithm that searches for better wings. Along the way I wired in CL–CD polars from OpenVSP, experimented with TSFC and fuel-burn stepping, and started sketching how mission segments will be layered into the loop next. It is not perfect yet, but I now have a framework that can be iterated on instead of a pile of disconnected equations.

None of that would work without the progress my teammates made on the geometry and high-fidelity side. While I focused on the range logic and optimization, they built and refined the CAD models, set up the OpenVSP wing with the right planform and twist controls, and began running ANSYS/Fluent cases we can eventually use for validation and tuning. Their work gives me real dimensions, realistic airfoils, and aerodynamic data that I can plug into my equations, and my scripts give their models a way to be evaluated and compared systematically. By the end of the semester, we are not just running isolated CAD or CFD models anymore; we are slowly turning them into a multi-fidelity design and optimization pipeline that can be handed back to our sponsors and built on next term.

Capstone Meeting w/ Client

02/09/2026

Thinking of using a model from the NASA Common Research Model for the next airplane model for the new semester

https://commonresearchmodel.larc.nasa.gov/high-lift-crm/high-lift-crm-geometry/reference-geometry/

It will make things a lot easier that Matt Seay has already said.

CST airfoils

Capstone Meeting w/Client

02/16/2026

*ChatGPT Aided Summary of my meeting notes:

Topic summary: Advisor check-in focused on defining a realistic end-of-year plan for the wing optimization workflow, ensuring the framework is not under constrained, and expanding capabilities to include airfoil optimization, flap/high-lift effects, and credible post–re-wing weight and fuel-volume considerations so the same vehicle can be optimized for both cruise and takeoff/landing requirements.

Thomas Wells (Lockheed Advisor) and the context of Matt Seay (SAI old advisor) came up in a discussion centered on next-semester goals and how the team should forge a clearer development plan for the end of the year. At this stage, the advisors' suggestions were not highly developed because the group is still figuring out the direction and how to structure the workflow so it supports meaningful engineering decisions. It was noted that Thomas was somewhat uninformed about the specific tools currently inside the team's framework, which matters because the direction they want to pursue depends on evaluating certain performance and feasibility metrics correctly. The team wants to avoid producing results that are nonsensical or underconstrained, especially if the optimizer is allowed to exploit missing constraints or incomplete modeling.

A major theme was the benefit of leaning on open-source tool sets and building an automated pipeline around them. The rationale is that some of the calculations and trade studies the advisors want to see are difficult to do in a manual or ad hoc way. If the team can integrate existing open-source capabilities and automate the analysis loop, they can extract significantly more useful work from the tools while keeping the workflow consistent and defensible.

High-lift modeling was raised explicitly through the question of flap considerations. In the current state of the framework, there is nothing that accounts for how a trailing-edge flap would deform lift curves or change drag behavior. That omission is important if the design needs to satisfy takeoff and landing requirements in addition to cruise. The advisors and team discussed adding additional functions or modules that can represent flap effects so that the workflow can capture the performance shifts caused by flap deflection rather than treating the wing as a fixed clean configuration across all conditions.

There was agreement that coupling the wing optimization with airfoil optimization would be a strong direction for the project. The intent is not only to optimize aerodynamic performance, but also to incorporate practical constraints and outputs such as fuel volume/capacity and weight impacts so the result is meaningful at the vehicle level. The discussion emphasized taking new requirements, running them through the workflow, optimizing performance under those requirements, and then producing the deliverables that demonstrate how the optimized configuration meets the goals.

The parameters the team is best positioned to support were identified as wing planform variables and the setup of a trailing-edge flap. A key design question emerged: if the team holds the same vehicle and requires optimization for both "up and away" performance and takeoff/landing, how does that change the resulting wing? The conversation framed this as a "re-wing" derivative problem—designing a new wing that balances cruise objectives with airfoil penalties and high-lift requirements, leading to a wing that may look very different than one optimized only for cruise.

Finally, a "post-game interview" concept was discussed around how to handle weight and configuration after the aircraft is re-winged through optimization. The working assumption mentioned was that fuselage weight stays constant and fuel capacity may be treated as constant at first, but changing the wing introduces weight considerations that must be handled consistently. The team needs a defensible approach for estimating how wing weight changes with planform and design choices, including which equations or methods should be used to model a new wing's weight after optimization so the results remain realistic and internally consistent.

MATLAB Code Work

02/22/2026-02/27/2026

First, I started with prompting GPT on how to condense the code without removing the performance of the functions. I'm still having a little trouble with the GA stopping halfway through the simulations, seems to be a max generations or variability issue, so I'm trying to clean up the code so that I can search through it seamlessly.

Here is the prompt that I gave GPT: "Let's find ways to combine files or functions so that there are only around 5-6 files and I can work through them easier".

So, GPT gave me 2 options for condensing. The first was very straightforward and would lead to there only being 5 files so I went with that one. I started with making a new file called objective constraints.m and wanted to port basically everything from flightscoreobjective inside and have the parameters clearly listed all the way down the line so I can edit them in a much better way.

Bottom line is I have created an easy one-story flow for the optimization design space. GA will choose a design, runWingSimulation will run the calculations for the model with the parameters and return a single struct called sim. MissionRangeJet computes cruise range in NM using a stepwise Breguet integration (dynamic) and the optimizer uses its objectives to maximize range and set constraints for the fuel capacity and CL range feasibility. Everything now runs through one set of parameters and one struct output sim, so that I don't have scattered constants and mismatched inputs going forward. Having a lot of degrees of freedom for the constants name was kind of a hassle so I decided to use ChatGPT to help me condense them into a format that is more maintainable for future use.

Where I started:

A. I condensed the accessory code that was causing confusion before. Having multiple functions and files helping each other call and define variables in separate spaces was a headache to maneuver. So, I went ahead and made sure that the GA can now directly handle functions returned by a new file called objective_constraints.m. The objective and constraints are now located in this place, as the name states, and the simulations return a struct as opposed to 14 different variables.
B. I created a real parameters system so it's constants arent scattered everywhere throughout the code. We have a centralized control panel for all parameters now and they are placed

inside of a function called defaultParams() inside of objective_constraints.m. It now defines the cruise constants, polar usage toggles, penalties, constraint switches, and CL bounds used to keep polar interpolation valid. This was essential to prevent the discord between the simulation, objective, and constraints and their assumptions.

C. I updated runWingSimulation to a single output struct and plugged in VSPAERO polar usage.
D. Aerodynamic fidelity changed as well. I changed the cruise drag calculation so that its valid when using the OpenVSP model. The cruise weight, dynamic pressure, and area are also read from the polar file called x58.polar from the OpenVSP analysis. It also interpolates the CD required and CL requirements as well from the file getCDfromCL and the L/D.
E. Also made sure that the polar file had all of the required variables for reading by the simulation. It has AoA, CLtot, and CDtot. These were imperative for the physics assumptions that we had for the aircraft. Necessary for the completion of our project because we needed a decent baseline to observe and back our understanding of the bounds of our design space.
F. Created a single mission-range file because there were three files contributing to the same task: breguetCruiseLoopJet.m, breguetStepJet.m, and schedules_constant.m. I combined all of them into one file called missionRangeJet.m. Now, it pulls its inputs from the sim and params that we have set and runs a stepwise cruise integration and returns a Range in NM that is consistent with our objective convention.
G. Improved GA post-processing and made plots more reliable so they show information that we didnt have before. It required me to change some of the parameters on the fitness of certain aircraft wing designs so that it would actually filter through all 200 generations and not stop halfway or even 80% through. It was quite the challenge. Then, once it went through all generations I updated the heat map to show landscape, basically a mountain now instead of just tiny dots, and gave more scatter plots for the other variables because it gives us a decent bit of information of what the generation tool likes to cheat at.
H. Once I was done with all of those things I asked ChatGPT to completely renovate the files and clean them all up so there wasn't just an enormous amount of useless variable names and dead code. Now, the code is similar in the number of files to the one from first semester but adds a lot more information and usability that we can have gone forward.

Next Steps:

*Aided by ChatGPT exploring the next steps from the pdf document received from Matt Seay our current advisor.

Next few days summary: Lock a reproducible MATLAB baseline, wire in the two sponsor constraints (landing approach speed and thickness-to-span), then connect an airfoil CLmax module so the landing constraint is physically tied to airfoil choice. Finish by automating exports and plots so the team can drop results straight into the client deliverables.

For the next few days, make sure the current pipeline runs cleanly end-to-end with consistent `params` and `sim` outputs, and add the landing constraint plumbing by computing $Vstall$ and $Vapp=1.2$ inside `runWingSimulation`, then enforcing $Vapp \leq 120V$ kt as a nonlinear constraint. Also add clear printed "constraint margins" after each optimization run so the team can immediately see feasibility.

Over the next few days, build the airfoil module interface that returns **CLmax** and plug it into the stall speed calculation. It can start simple, but it must exist as a real function so the workflow matches the client requirement of "airfoil optimization coupled with the wing," and so the landing constraint responds to airfoil parameters instead of being hardcoded.

After that, implement the thickness-to-span constraint using a definition agreed by the team (but coded modularly so it's easy to change), and structure the simulation so it's naturally "multipoint-ready" with a cruise evaluation for the objective and a landing evaluation for the stall constraint. Once those are stable, automate outputs: export top designs to Excel with range and constraint margins, and save our final plot set (including the 3D range landscape) to a folder for documentation and slides.

MATLAB Code Work

03/22/2026-03/23/2026

AI Assisted Generation of Synopsis of Session Documentation & Code Development Log

1. Project Overview

This document summarizes the full development session for the XQ-58A Valkyrie wing optimization capstone project, developed in collaboration with Lockheed Martin. The goal is a repeatable, multi-fidelity aircraft wing optimization workflow that links parametric geometry, aerodynamic analysis, optimization logic, and higher-fidelity verification into a single defensible engineering process.

Two parallel codebases were developed and analyzed during this session:

- MATLAB capstone suite — primary XQ-58A optimization using a Genetic Algorithm (GA) with VSPAERO polar data from real CFD geometry
- Python AeroSandbox notebook — higher-fidelity simultaneous planform + airfoil shape optimization using gradient-based IPOPT

The team decision at the end of this session was to consolidate onto the Python workflow, which provides a more complete and maintainable single-codebase solution.

2. MATLAB Codebase — What Was Built and Fixed

2.1 Files Revised

The following MATLAB files were systematically reviewed and corrected during this session:

File	Primary Role
MainDriver.m	GA driver, result export, all visualization plots
objective_constraints.m	Objective function, nonlinear constraints, defaultParams
runWingSimulation.m	Full aerodynamic + weight simulation per design point
missionRangeJet.m	Stepwise Breguet range integration
loadVSPPolar.m	VSPAERO .polar file reader with startup validation
getCDfromCL.m	CD interpolation from polar with clipping warning
getIntegratedProps.m	Spanwise integration of t/c, cl, and wing volume
saveGAHistory.m	GA output function with per-generation convergence tracking
validateBaseline.m	Standalone XQ-58A baseline validation report (NEW)

2.2 Optimizer Fixes (MainDriver.m + objective_constraints.m)

Degenerate convergence — the GA was stuck at the boundary

The original run produced 100 near-identical designs all at maximum span (38.5 ft), maximum taper (0.60), and sweep around 39.5 deg. The penalty system was too weak to overcome the raw range advantage of boundary hugging.

- Penalty system replaced: global quadratic -> one-sided linear ramp activating only in the outer 20% of each variable's range. Interior designs receive zero penalty.
- Penalty weights raised: k_span 200 -> 600 NM, k_sweep 100 -> 300 NM (calibrated to exceed the ~241 NM boundary advantage seen in the data)
- Taper penalty added: taper was pinning at 0.60 with zero consequence. k_taper = 350 NM added.
- MaxStallGenerations reduced: 500 -> 40. The original allowed 75,000 evaluations of pure stalling.
- EliteCount increased: 3% -> 5% to preserve interior designs across generations.
- evalWithCache added: objective and constraint functions were each calling runWingSimulation independently for the same design point, doubling evaluation cost. A one-slot cache eliminates the redundant call.
- Deduplication added to export: the Top 100 sheet was filled with clones of the same boundary design. History is now rounded to 2 decimal places and deduplicated before export.

- failIfOutsidePolar changed to false: hard rejection of designs slightly outside the polar CL range was shrinking the feasible space. Graceful pchip clamping in getCDfromCL is safer.

Stall and approach speed constraint

The original code used CL_max_land = 1.20 as a fixed assumption with no connection to the actual airfoil. This was upgraded to the full PDF workflow model:

- CLmax2D_clean = 1.33 — from the multi-point transonic airfoil optimizer result (optimized section, clean, stall at 12 deg AoA)
- CLmax3D = eta_CLmax_3D x CLmax2D_clean, with eta = 0.90 for sweep and tip losses
- Vapp = Vapp_margin_factor x Vstall <= Vapp_max_kt (FAR 25 form: 1.3 x Vstall)
- All stall calculations moved to runWingSimulation Section 7. nonlcon becomes a clean one-liner: c = sim.Vapp_kt - params.Vapp_max_kt
- W_landing = W_landing_frac x TOGW (0.85), replacing the previous ZFW + reserve estimate

2.3 runWingSimulation.m Fixes

- Duplicate fuel bookkeeping eliminated: the same fuel block ran twice, with the second pass overwriting the first. One block remains, correctly placed after the sanity check.
- Sanity check now fails cleanly: previously overwrote values with junk sentinel data and continued with sim.ok = true. Now sets sim.ok = false and returns immediately.
- Cruise CL uses capacity-limited fuel: was using fixed baseline fuel (1440 lbf) regardless of tank size. Small wings were being evaluated with more fuel than they could carry, inflating CL and triggering false constraint violations.
- Fuel volume corrected: getIntegratedProps assumes rectangular cross-sections. Applied k_airfoil_section = 0.68 (NACA 6-series form factor) to correct the ~32% overestimate of tank volume.
- Baseline comparison fields added to sim struct: L_D_base, CD_total_base, CL_base, alpha_deg_base, span_ft_base, AR_base, W_wing_lbf_base, W_ZFW_lbf_base all packed into sim for downstream reporting.
- Version-keyed persistent guard: CONST_VERSION = 'v4' detects and wipes stale persistent state automatically, eliminating 'Unrecognized field name' crashes after file updates.
- Baseline twist constant named: base.twist_deg = 0.0 replaces a magic number buried in the Oswald efficiency formula.

2.4 missionRangeJet.m Fixes

- Unit derivation documented: V in knots = NM/hr, TSFC in 1/hr. Their ratio gives NM directly. No conversion factor needed. This is the standard Raymer/Brandt form.
- Zero fuel guard: fuel_avail_lbf <= 0 now returns RangeNM = 0 immediately instead of running a near-infinite loop.
- L/D and weight guards: non-positive or non-finite values return NaN so the GA treats them as failures rather than minimizing a negative range.
- error() replaced with NaN return in breguetStepJet: a thrown error inside a GA evaluation crashes the entire run. NaN propagates cleanly.
- stepFrac upper bound tightened: 0.25 -> 0.05. At 0.24 only 4 integration steps are taken for the whole cruise, degrading Breguet accuracy significantly.

2.5 loadVSPPolar.m and getCDfromCL.m

- Mach and Re/1e6 columns extracted from the polar file and stored in polar.Mach and polar.Re_e6 for cruise-condition consistency checking.
- Startup validation printout: every time the polar loads, it prints CL range, CD range, best L/D, and whether the optimizer's CL bounds fall inside the polar. Your x58.polar covers CL -0.266 to +0.569; optimizer bounds 0.05 to 0.55 are fully inside.
- One-time clipping warning in getCDfromCL: fires once per session when CL falls outside the polar range, keeping the GA log readable without flooding it.

2.6 getIntegratedProps.m

- Input validation added: negative span, zero chord, taper outside (0,1] now throw descriptive errors instead of silent NaN propagation.
- Strip count increased: 20 -> 50 for margin if nonlinear airfoil schedules are added later.
- Area self-check: integrated semi-span area compared against closed-form trapezoid result. Discrepancy over 0.5% triggers a warning.

2.7 saveGAHistory.m

- Per-generation convergence table added: each generation records best score, mean score of feasible individuals, feasible count, and best range in NM. Exported as ga_convergence table to base workspace.
- Convergence summary printed at run end: total generations, evaluations, best range, and final-generation feasibility percentage.
- Population size guard: protects against out-of-bounds writes if MATLAB shrinks population near stall.

2.8 validateBaseline.m (New File)

A completely standalone script that runs the XQ-58A reference design through the full pipeline and prints a structured 8-section report. Run it before any optimization to confirm the model is behaving correctly.

- Section 1 — Wing geometry: span, chord, area, AR, sweep (both LE and quarter-chord), avg t/c, twist
- Section 2 — Weights: wing weight, ZFW, fuel, TOGW, W/S
- Section 3 — Cruise atmosphere: altitude, ISA temperature, density, speed of sound, dynamic pressure
- Section 4 — Aerodynamics: CL, full CD breakdown (CD0, CDi, CD_wave, CD_creep), L/D analytic vs polar, M_dd, AoA
- Section 5 — Fuel and tank: capacity, fuel carried, reserve, cruise fuel available
- Section 6 — Mission range: Breguet cruise range
- Section 7 — Constraint PASS/FAIL: each constraint evaluated exactly as nonlcon does with margin values
- Section 8 — Summary table: key outputs side by side with VSPAERO polar reference values (polar best L/D = 18.62 at CL = 0.1994)

---

3. Edwin's MATLAB Files Built and Fixed

Randal Richards

3.1 X58_Optimization_Driver.m

A standalone single-aircraft optimizer for the X-58 with embedded local functions. Required five hard fixes to run at all:

- Typo fix: 'op retions.PopulationSize' -> 'options.PopulationSize' in saveGAHistory iter case. This was a hard parse crash before any GA generation ran.
- strato() moved to top-level local function: was illegally nested inside runWingSimulation which is itself a local function inside a script. MATLAB prohibits this.
- objective_constraints(params) interface fixed: the function now accepts an optional customParams input argument matching the batch suite pattern.
- Wave drag corrected: 50*(Mach-Mdd)^2 -> 20*(Mach-Mdd)^2. The old coefficient overstated wave drag by 2.5x.
- params.badConstraint = 1e6 added: was missing from defaultParams, causing the bad-constraint return to have no value.

3.2 Master_Optimization_Suite.m

Multi-aircraft batch optimizer running 737, C-130, and RQ-4 through the same framework with class-specific constants. Five fixes required:

- params.badConstraint = [] -> 1e6: empty matrix caused ones(n,1) * badConstraint to return [], giving the GA an empty constraint vector it treated as unconstrained.
- set(fig4, 'Position') -> fig4.Position = [100 100 1300 480]: missing value argument crashed immediately.
- objective_constraints(params) interface fixed: same as X58 driver.
- Wave drag coefficient corrected: same 50 -> 20 fix.
- useVSP = true removed from multi-aircraft loop: x58.polar is geometrically wrong for 737/C130/RQ4 and was doing nothing anyway.

3.3 Plot_Wing_Overlays.m

- set(fig, 'Position') with no value fixed: the original crashed on the second executable line. Replaced with fig.Position = [100 100 1300 480].
- Hardcoded optimized data now has a clear comment explaining how to replace placeholders with real GA output.

---

4. Python AeroSandbox Notebook — Assessment

4.1 What the Python Code Does (That MATLAB Does Not)

Category	Python capability	MATLAB equivalent
Optimizer	IPOPT gradient solver with exact CasADi derivatives	GA — global but gradient-free
Airfoil	CST shape optimization (32 variables)	Fixed airfoil (t/c only)

Aero fidelity	NeuralFoil viscous drag + VLM spanload	Raymer analytic formulas
Fuselage	Body of revolution drag model	Not modeled
Reynolds number	Varies with altitude + chord (NeuralFoil)	Not modeled
Wing weight	Solved as NLP variable (algebraic loop)	Sequential evaluation
Multi-class	GA / transport / fighter / UCAV presets	Single XQ-58A model
Visualization	Interactive 3D Plotly wing + airfoil profiles	Static MATLAB figures

4.2 What MATLAB Does That Python Does Not

- VSPAERO polar lookup: MATLAB uses x58.polar from an actual VSPAERO run on the XQ-58A geometry. Python uses AeroBuildup which is AeroSandbox's own VLM + NeuralFoil and is NOT OpenVSP/VSPAERO.
- Explicit reserve fuel accounting: MATLAB tracks reserve as a separate quantity. Python's propulsion_factor form has no reserve term.
- Baseline validation report: validateBaseline.m confirms the model behaves correctly before any optimization. Python has no equivalent.
- Global search: the GA explores 150 x 40 generations across the whole design space. IPOPT finds the nearest local optimum from each starting point.

4.3 Critical Issues Found in the Python Notebook

CRITICAL: propulsion_factor = 1.2e6 is the GA piston aircraft value. For a military turbofan it should be ~2.5e6. This alone underestimates range by approximately 2x.

CRITICAL: The solver returned 'Converged to a point of local infeasibility.' The results shown (L/D = 20.34, range = 8186 km) are from a technically infeasible solution. The fuel volume or stall constraint is binding and cannot be fully satisfied at the current settings.

NOTE: AeroBuildup is ~2x optimistic on L/D vs VSPAERO for this geometry. The aero-only L/D of 31.82 reflects NeuralFoil's parametric model, not CFD-calibrated data. VSPAERO showed best L/D = 18.62 at CL = 0.199.

4.4 Team Decision: Consolidate to Python

After reviewing both codebases the team decided to move to the Python AeroSandbox notebook as the primary tool going forward. The rationale:

Randal Richards

- AeroSandbox combines airfoil design and planform optimization in one solver, eliminating the two-step MATLAB/OpenVSP workflow
- The parameter dashboard (ipywidgets) allows the team to change design inputs without editing code
- The 37-variable simultaneous optimization (planform + airfoil shape) is a higher-quality design process than optimizing planform with fixed airfoils
- Note: the Python AeroBuildup aero model is at conceptual design fidelity. For final verification, the top candidate designs should still be run through OpenVSP/VSPAERO as described in the capstone workflow PDF

5. Airfoil Optimization Results

5.1 Multi-Point Transonic Airfoil Optimizer Output

A separate multi-point airfoil optimization was run using CasADi/IPOPT targeting the XQ-58A cruise and stall conditions simultaneously. Key outputs:

Parameter	Value
Optimizer	IPOPT via CasADi (gradient-based NLP)
Starting airfoil	RAE 2822
Cruise target CL	0.192
Cruise Re	9.31 x 10^6
Stall target V	100 kt
3D CL required at stall	0.900
Clean 2D stall CL achieved	1.330 (at 12.0 deg AoA)
Maximized cruise L/D	3.55 (section, 2D)
Final max thickness t/c	0.140
Solver exit	Infeasible (stall constraint marginally violated)

The optimized root and tip airfoil coordinates were exported as:

- optimized_root_airfoil.dat — 400 coordinate points, Selig format
- optimized_tip_airfoil.dat — 400 coordinate points, Selig format

These .dat files feed directly into the MATLAB stall constraint (CLmax2D_clean = 1.33) and will be used to build the OpenVSP geometry for the next VSPAERO polar run.

6. Why the Numbers Differ Between MATLAB and Python

The two codebases produce substantially different results for the same aircraft. This is not an error — they are modeling different physics at different fidelity levels.

Quantity	X58 Driver (Python-style)	Capstone MATLAB
TOGW	6,000 lbf	4,540 lbf
Fuel	2,500 lbf	1,440 lbf
CD0_other	0.0150 (transport-level)	0.0070 (refined estimate)
Oswald e	0.85 fixed	0.96 x taper/twist factors
Wave drag coefficient	20 x (M - Mdd)^2 (fixed)	Raymer Lock formula
CD source	Analytic only	VSPAERO polar (primary)
Best L/D	8.62	~18 (from polar)
Range	2,837 NM	~2,045 NM (boundary-hugging)
Best span	25 ft (lower bound)	38.5 ft (near upper bound)

The L/D of 8.62 in the X58 Driver is physically consistent with its own constants. CD0_other = 0.0150 dominates the drag budget so heavily that increasing span (and wing area) makes things worse rather than better. The optimizer rationally picks minimum span in response to its drag model.

The MATLAB result of ~2,045 NM at maximum span reflects boundary-hugging, which was the primary optimization problem fixed during this session.

---

7. Path Forward — Goals and Recommended Next Steps

7.1 Critical Fixes Before Next Run (Python)

- Fix propulsion_factor: change 1.2e6 to ~2.5e6 for a military turbofan. This is the single most impactful number to correct — it doubles the range prediction.
- Add reserve fuel to Breguet: subtract W_reserve from W_fuel_N before computing ln(Wi/Wf). The current form burns 100% of fuel.
- Diagnose infeasibility: add a post-solve block that prints each constraint value and margin so the team can see exactly which constraint is binding.
- Integrate x58.polar as a CD override: read the polar file and replace AeroBuildup's drag with the VSPAERO interpolated value at the solved CL. This is the most direct path to matching MATLAB fidelity.

7.2 Parameter Dashboard

The team wants a widget-based UI in the Jupyter notebook so design parameters can be adjusted without editing code. The recommended approach using ipywidgets:

```
import ipywidgets as widgets
from IPython.display import display

altitude_slider = widgets.FloatSlider(value=12000, min=5000, max=18000, step=500,
                                      description='Altitude (m)')
mach_slider     = widgets.FloatSlider(value=0.72, min=0.50, max=0.88, step=0.01,
                                      description='Cruise Mach')
span_max_slider = widgets.FloatSlider(value=22, min=5, max=30, step=0.5,
                                      description='Max Span (m)')

def update_config(altitude, mach, span_max):
    config['altitude_m']      = altitude
    config['v_cruise_mps']    = mach * asb.Atmosphere(altitude=altitude).speed_of_sound()
    config['span_bounds']     = (config['span_bounds'][0], span_max)

out = widgets.interactive(update_config, altitude=altitude_slider,
                          mach=mach_slider, span_max=span_max_slider)
display(out)
```

This makes the notebook directly usable by team members and sponsors without Python knowledge. All config updates propagate automatically because the optimizer reads from the config dict.

7.3 Adding Control Surfaces

Control surfaces are the next logical geometry addition. There are two levels at which this can be done:

Level 1 — Drag and weight penalty (fast, no new solver)

Add control surface area as a fixed fraction of wing area (Scsw = 0.15 x Sw for transport, 0.10 x Sw for fighter). This is already in the Raymer wing weight equation but the drag penalty from deflected surfaces is not currently modeled. Add a CD_controls term:

```
# Deflected control surface parasite drag (Raymer Table 12.3)
# Typical aileron + elevator contribution at cruise: 0.0002 - 0.0008
CD_controls = config.get('CD_controls', 0.0003)  # add to defaultParams
_D_controls = CD_controls * _q_cruise * S_w
_D_total    = aero_results['D'] + _D_wave + _D_misc + _D_controls
```

Level 2 — Geometric control surfaces in AeroSandbox (full aero coupling)

AeroSandbox supports explicit control surface geometry through the ControlSurface object. This allows the optimizer to size aileron and elevator span and chord as additional design variables with their own hinge moment, effectiveness, and drag penalties:

```
aileron = asb.ControlSurface(
    name='Aileron',
    symmetric=False,        # differential deflection
    deflection=0,           # neutral at cruise
    hinge_point=0.75,       # 75% chord hinge
)

# Add to the outboard WingXSec
outboard_xsec = asb.WingXSec(
    ...,
    control_surfaces=[aileron]
)
```

Adding control surfaces as optimizer variables would require additional constraints: minimum roll authority (aileron effectiveness x dynamic pressure x moment arm >= required roll rate), maximum hinge moment for actuator sizing, and flutter margin based on surface chord fraction. These are medium-difficulty additions that are well within the AeroSandbox framework.

7.4 Multi-Aircraft Capability

The Python notebook's preset system is already designed for multi-aircraft use. The _PRESETS dictionary has entries for GA, transport, fighter, business_jet, and UCAV. Extending to new aircraft types requires only adding a new preset block and setting config['aircraft_class'] accordingly.

For sponsor-facing demonstrations comparing multiple aircraft:

- Loop over aircraft_class values and store results per class
- The bar chart and planform overlay infrastructure from Master_Optimization_Suite.m can be translated directly to Plotly in the Python notebook
- For each aircraft the team wants to study, add its specific parameters (W_fixed_N, W_fuel_N, root_chord_m, v_cruise_mps, altitude_m) to the config dict before calling the optimizer

Recommended near-term additions to the preset system for your sponsors:

- XQ-58A — explicitly calibrated to the Valkyrie geometry (currently using ucav preset as a proxy)
- MQ-9 Reaper — medium-altitude long-endurance UCAV for comparison
- Custom UAV — blank-slate class for exploring new design concepts

7.5 Dropping Fuel Considerations

Since your sponsors are not focused on fuel and range as primary metrics, the optimizer objective can be changed from range maximization to a different performance metric. Options:

- Maximize L/D directly: opti.minimize(-L_over_D). This removes the Breguet equation entirely and focuses the optimizer purely on aerodynamic efficiency.
- Minimize CD at a fixed CL: enforce L = W as an equality constraint and minimize drag. Clean and fast.
- Maximize endurance factor: for surveillance missions, E = (L/D) / V is the relevant metric — the optimizer would favor lower speed and higher AR.

- Multi-objective (L/D vs weight): minimize a weighted combination of -L_over_D + w x W_wing_N. Sweeping the weight w produces a Pareto front showing the efficiency-weight tradeoff.

The Breguet propulsion_factor and fuel weight inputs can be simply removed from the config, eliminating the reserve fuel issue entirely. The structural, stall, and geometric constraints all remain valid regardless of whether range is the objective.

7.6 Recommended Implementation Order

Priority	Task
1 — Immediate	Change objective to L/D maximization (drop fuel/range)
2 — Immediate	Fix propulsion_factor if range is still needed for any sub-case
3 — This week	Add parameter dashboard with ipywidgets
4 — This week	Add baseline validation section to Python notebook
5 — Next sprint	Add control surface drag penalty (Level 1)
6 — Next sprint	Integrate x58.polar as CD override option
7 — Later	Add geometric control surfaces as optimizer variables (Level 2)
8 — Later	Build multi-aircraft comparison loop and dashboard
9 — Final	Run top-5 designs through OpenVSP/VSPAERO for verification

---

8. Key Constants and Reference Values

8.1 XQ-58A Valkyrie Baseline (MATLAB model)

Parameter	Value
Baseline span	27.0 ft (8.23 m)
Root chord	5.53 ft (1.69 m)
Taper ratio	0.375
LE sweep	30.0 deg
Wing area (S_ref)	~99 ft^2
Aspect ratio	~7.4

ZFW	3,100 lbf
Fuel load	1,440 lbf
Cruise altitude	35,000 ft
Cruise Mach	0.72
Root airfoil	NACA 64-206 (t/c = 0.06)
Tip airfoil	NACA 64A204 (t/c = 0.04)
VSPAERO best L/D	18.62 at CL = 0.1994
VSPAERO CL range	-0.266 to +0.569

8.2 Python Notebook Configuration (current)

Parameter	**Value**
Aircraft class	ucav
W_fixed_N	3100 x 4.448 = 13,789 N
W_fuel_N	1440 x 4.448 = 6,405 N
Cruise speed	213 m/s (~414 kt)
Cruise altitude	12,000 m (39,372 ft)
Cruise Mach	0.720
Root chord	2.23 m (7.32 ft)
Tip washout	-3.0 deg
Optimized span	8.86 m (29.1 ft)
Optimized taper	0.15
Optimized LE sweep	27.27 deg
Wing area	11.37 m^2 (122 ft^2)
Aspect ratio	6.91
Net L/D	20.34 (NOTE: infeasible solution)
propulsion_factor	1.2e6 (NEEDS FIX: use 2.5e6 for turbofan)

Capstone Showcase:

04/22/2026

The University of Georgia Capstone Design Showcase is one of the most highly anticipated annual events hosted by the College of Engineering. Held at the Classic Center in downtown Athens, Georgia, the showcase serves as the culminating public presentation for senior engineering students who have spent an entire academic year working alongside real industry partners, government agencies, hospitals, and nonprofit organizations to develop functional, well-researched engineering solutions to genuine, real-world problems. The event brings together students, faculty, industry professionals, alumni, and community members under one roof in a celebration of applied engineering and collaborative innovation. This year's showcase was particularly well-attended, with teams from nearly every engineering discipline represented, spanning biomedical, mechanical, industrial, electrical, and environmental engineering.

Walking into the Classic Center felt immediately different from a standard classroom setting. The space was transformed into a sprawling exhibition floor lined with tri-fold and banner-style presentation boards, each one a visual summary of months of design work, prototyping, testing, client meetings, and iteration. The energy in the room was hard to miss. Students who had spent the better part of an academic year grinding through design reviews, sponsor check-ins, and late nights in the engineering labs were finally getting the chance to stand in front of their work and share it with the world. The pride was visible on every team's face, and it was genuinely infectious to be around.

The event was structured around a public viewing period during which attendees could move freely between presentations, ask questions, and engage directly with the student teams. Industry judges rotated through the floor on a schedule, evaluating each team based on technical rigor, design quality, communication, and adherence to engineering standards including ABET student outcomes. In addition to the formal judging component, attendees at the showcase were given the opportunity to cast votes for a Viewer's Choice Award, a special recognition selected entirely by the audience based on their impressions of the projects throughout the event.

One of the most memorable aspects of attending the showcase was simply being able to observe such a wide range of engineering disciplines applied to such different problems. In one corner of the room, a team might be discussing fluid dynamics in a medical device context, while just a few booths down, another group was demonstrating a software simulation of a manufacturing line or walking visitors through a 3D-printed prototype of an assistive technology device. The breadth of what engineering looks like in practice was on full display, and for students who are still earlier in their academic careers, it provided a meaningful and concrete picture of what capstone-level work actually entails.

The other teams presenting at the showcase were impressive across the board. It was clear that nearly every group had invested significant time not only into their actual engineering work but also into the way they communicated that work to a mixed audience. Posters were well-designed and thoughtfully laid out, with most teams using a combination of process diagrams, data tables, prototype photographs, and conclusions sections to guide viewers through the narrative of their project from problem identification to final solution. Several teams also brought physical prototypes to their stations, which drew consistent crowds of curious attendees who wanted to see and sometimes interact with the physical output of the design process. The overall quality of the showcase made it genuinely difficult to single out just three projects to evaluate, as there was something compelling to observe at nearly every station visited.

One thing worth noting about the judging process was that the formal industry judges were working under significant time constraints. With a large number of teams to evaluate and a finite window in which to complete their rotations, it was apparent that some presentations received more substantive judge engagement than others. A few teams seemed to get rushed through their presentations or received only a brief visit before the judges moved on to the next station. This is understandable given the logistical complexity of coordinating a judging effort across that many projects simultaneously, but it did mean that some groups may not have had the opportunity to fully convey the depth of their work during the formal evaluation. For those teams, the public viewing period and the Viewer's Choice component of the event provided a more open and organic opportunity to connect with attendees and share their projects without the time pressure of a judge rotation.

Despite any unevenness in the judging experience, the overall atmosphere of the event remained remarkably positive and enthusiastic. Teams cheered each other on, faculty advisors watched proudly from the sidelines, and industry partners who had spent the year collaborating with student teams seemed genuinely pleased with what they saw. The showcase did exactly what it is designed to do: it created a real-world stage for students to present professional-level work, practice technical communication, and experience what it feels like to have their engineering decisions scrutinized and celebrated in equal measure. For anyone attending the event without prior knowledge of what capstone design entails, the showcase would have served as a thoroughly convincing demonstration of the depth and seriousness of the engineering education happening within UGA's College of Engineering.

Randal Richards

Perhaps the highlight of the entire event, and certainly the most exciting moment for our team, was the announcement of the Viewer's Choice Award at the conclusion of the public viewing period. After hours of conversations with attendees, explanations of our design process, and demonstrations of our prototype and results, it was announced that our team had received the most votes from the audience and had been selected as the Viewer's Choice winner. The recognition was a genuinely thrilling and unexpected honor that validated the enormous amount of work that went into the project over the course of the year. Winning an award selected by the public rather than solely by technical judges carried its own special weight, because it reflected not just the engineering quality of our work but our ability to communicate it clearly and compellingly to a broad and diverse audience. The moment was one of the most rewarding experiences of the academic year and a fitting conclusion to what had been an intensive and deeply educational capstone experience.

Team 1: SecureWick

Project Title: Urinary Incontinence Design for Mobile Patients

External Partner: Becton Dickinson

The SecureWick team designed a secondary support device to reduce displacement of BD's PureWick Female External Catheter during patient movement. Their StatLock-compatible final design reduced average displacement from 2.64 cm down to 1.2 mm, a 95% improvement over the baseline device.

ABET Outcome	How the team addressed it
1. Math, science, and engineering	The team calculated average displacement values across three prototypes to compare performance and guide design decisions.
2. Design within constraints	They designed a wearable support device within FDA Class I guidelines, targeting under 10 mm displacement and a short application time.
3. Communicate effectively	The poster used clear visuals, labeled diagrams, and a simple results table that was easy for both technical and general audiences to follow.
4. Professional and ethical responsibility	The team considered patient comfort and infection risk reduction, noting their device follows ISO standards and is FDA-exempt.
5. Multidisciplinary teams	Four students collaborated with an industry sponsor (Becton Dickinson) and faculty advisor, each contributing to different aspects of the design.

Randal Richards

6. Experimentation and data analysis	They tested three devices under simulated patient motion and compared displacement results to select the best-performing design.
7. Acquire new knowledge	The team learned about the StatLock fixation system and ISO biocompatibility standards as part of the design process.

Team 2: NOVA

Project Title: Canine Model for Thoracic Auscultation Training

External Partner/Client: Dr. Amie Koenig, UGA College of Veterinary Medicine

The NOVA team built a low-cost canine simulation model to train veterinary students in thoracic auscultation without the use of live animals. The prototype features a 3D-printed skeleton, layered silicone ribcage, and integrated audio equipment that plays realistic lung and heart sounds across independent zones.

ABET Outcome	How the team addressed it
1. Math, science, and engineering	The team used CAD stress analysis to check joint durability and applied acoustic principles to produce realistic lung and heart sounds.
2. Design within constraints	The model was built to match Labrador anatomy, stay under a $200 budget, and use replaceable parts for long-term use.
3. Communicate effectively	Their poster included a fabrication flowchart, labeled diagrams, and a QR code linking to additional results, making it accessible to multiple audiences.
4. Professional and ethical responsibility	Replacing live animals in training reduces ethical concerns around animal use and makes simulation more accessible for veterinary programs.
5. Multidisciplinary teams	Six students worked across engineering and veterinary medicine, partnering with a clinical veterinarian and UGA CVM students for validation.
6. Experimentation and data analysis	They tested audio jack durability through repeated stress cycles and surveyed veterinary students to evaluate sound realism and usability.
7. Acquire new knowledge	The team learned silicone fabrication, 3D printing, and embedded audio integration to build the physical prototype.

Randal Richards

Team 3: NutraSky Manufacturing

Project Title: Scalable Manufacturing System for Nutritional Supplement Production

External Partner/Client: NutraSky

The team used FlexSim simulation software to model and improve NutraSky's supplement production process. They proposed a closed transfer system and dual hopper configuration that achieved a 20% reduction in pre-blending time and identified the scooping table as the key production bottleneck.

ABET Outcome	How the team addressed it
1. Math, science, and engineering	The team modeled cycle times and throughput rates for each production stage using FlexSim simulation software.
2. Design within constraints	They proposed process changes that improved efficiency without requiring facility expansion or major capital investment.
3. Communicate effectively	The poster included a process flow diagram and a side-by-side simulation comparison table that clearly conveyed results to a business audience.
4. Professional and ethical responsibility	The project supports a growing supplement manufacturer in meeting demand from major retail partners while maintaining production quality.
5. Multidisciplinary teams	Five students collaborated with NutraSky's operations team and a faculty advisor to understand and improve a real manufacturing workflow.
6. Experimentation and data analysis	Four simulation runs compared original and updated configurations, identifying the scooping table as the primary bottleneck.
7. Acquire new knowledge	The team self-taught FlexSim and gained domain knowledge in supplement manufacturing to accurately represent NutraSky's process.

Part 3: Personal Reflection

Attending the UGA Capstone Design Showcase was a useful reminder of what the next few years of coursework are building toward. Seeing finished projects displayed by real teams made the idea of capstone design feel a lot more concrete. It was helpful to see how different each project was in terms of industry, methods, and scale, but how they all shared the same basic structure of identifying a problem, designing a solution, and testing it.

A few things stood out. The SecureWick project was a good example of how small design changes can have measurable clinical impact. The NOVA project impressed me because it involved fabricating something physical and creative, not just running analysis. The NutraSky project was interesting because it showed that engineering work can look like simulation and process design rather than building hardware.

Going into my own capstone, I want to make sure I have a clear success metric early on so the project has a measurable goal. I also want to communicate with the client throughout the process rather than just at the start and end. To prepare, I plan to stay organized in my coursework and look for opportunities to work with outside partners through internships or research before capstone begins.

AI DISCLAIMER:

Portions of this notebook may have been developed with the assistance of artificial intelligence tools, including large language models. All AI-generated content has been reviewed, edited, and verified by the author. The ideas, analysis, and conclusions presented remain the intellectual work of the author, and any errors or omissions are solely their responsibility.