Multidisciplinary Design Optimization of Electric Aircraft Considering Systems Modeling and Packaging

by

Benjamin J. Brelje

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Aerospace Engineering and Scientific Computing)
in the University of Michigan
2021

Doctoral Committee:

Professor Joaquim R. R. A. Martins, Chair
Professor Krzysztof J. Fidkowski
Dr. Justin S. Gray, NASA Glenn Research Center
Professor Kevin J. Maki
Dr. Richard Snyder, Air Force Research Laboratory

Benjamin J. Brelje

bbrelje@umich.edu

ORCID iD: 0000-0002-9819-3028

To the love of my life Andrea,

and the love of ours, Louise Magdalena

Ad maiorem Dei gloriam

Acknowledgments

Completing a PhD is never a solo effort. Having benefited from the generosity and kindness of so many people, writing this acknowledgement is in some ways the most exciting part of the dissertation.

I owe my deepest gratitude to my wife, Dr. Andrea Brelje. Her constant encouragement and love has helped me through the most difficult times during graduate school. What is all the more amazing is that she was able to provide this encouragement despite dealing with both medical school and life-altering challenges of her own. Her practical support, especially taking care of our daughter while I focused on writing the dissertation, has been irreplaceable.

I am also profoundly thankful for the support of my parents, Steve and Pam Brelje. They are seemingly always available to listen when I call, and the countless hours of child care help have not only been a huge practical help, but also a real blessing for us to be able to enjoy quality time together. Likewise, I appreciate my mother-in-law Janet Berkemeier, my sister-in-law Gretchen, and family friend Zoe Walters who collectively spent many days watching Louise. Other family members, especially my grandparents and my brother Jon, have provided a helping hand at various key points along the way.

I can honestly say that I would never have *started* a doctorate, let alone finished, without my advisor and committee chair, Joaquim Martins. At time we met in undergraduate aircraft design, I couldn't have predicted that I would make MDO my career. As I explored returning to graduate school after time in industry, Quim enthusiastically answered my questions and welcomed me with open doors. With his advice and support, I was able to win the NSF fellowship, which transformed my experience in grad school and enabled me to pursue an entirely new branch of research. Every paper I coauthored with him turned out much better than if I had been flying solo, from the up-front research scoping advice that kept me from wild goose chases, to the detailed comments about how I can communicate my ideas more clearly. He also struck the perfect balance, never micromanaging but always there when I needed his advice. Graduating from the lab will be very bittersweet, but I don't plan on quitting the Slack channel any time soon.

I describe in this dissertation all the reasons I love using OpenMDAO, but the best feature is that it gives me an excuse to talk to Justin Gray, who gradually became a trusted mentor and friend.

Justin has probably spent 60 hours giving me free tech support on OpenMDAO and optimization writ large. The legendary Cleveland hackathon of 2019 produced at least three papers worth of research ideas by itself. This dissertation quite literally would not exist without him. Beyond research, his advice on job hunting and fatherhood helped put me at ease at big moments in my life.

Rich Snyder of the Air Force Research Lab has been a major supporter of this work from nearly day one. Our biweekly check-ins have helped keep my eye on the ball, and he's connected me with other research that's paid big dividends for this dissertation, especially the ESP software. I have greatly enjoyed being a part of the CCAVD collaboration and I'm sure our paths will cross again many times in industry/government relations.

I am also grateful to the other two Michigan faculty members on this dissertation committee. I have had the pleasure of knowing Professor Fidkowski since undergrad and his clear teaching on numerical methods helped crystallize concepts that I've used many times since. I haven't known Professor Maki for nearly as long, but I appreciate his willingness to spend time serving as the cognate member on this committee — even if I still can't beat his rollerski segment times on Strava, at least maybe he'll let me graduate.

I would be remiss if I didn't acknowledge the impact of my Boeing mentors on this dissertation. Without Marcus Dunavan giving me the chance to work on MDAO at Boeing Commercial Airplanes, I almost certainly wouldn't be where I am today. John Dixon originated the kernel of an idea that led to the work in this dissertation on spatial optimization. Thank you also to my Boeing Research and Technology managers Steve Villa, Jim Guglielmo, Dinesh Naik, and Vincent Lee for their support and helping me navigate the dual hats of engineer and student. Gabe Burnett supported me as I wrote my first academic paper and got my first patent. The tech fellow network including Sean Wakayama, Steve LeDoux, Kumar Bhatia, Dave Paisley, Ron Engelbeck, Ken Bouvier, Frode Engelsen, Dan Newman, Marty Bradley, Tom Grandine, and John Sabino all played unique roles in my journey here. Perhaps an MDAO dream team equal to Brent Robbins, David Scott and me will be assembled again someday, but I guarantee you they won't have as much fun as we did. If I were to mention all my colleagues and friends this page could go on for ages, but in particular my friend and former roommate Greg Cass has helped me to remember to contextualize my efforts in the broader industry.

Thank you to my colleagues in the MDOlab for making Michigan such a fun place to work and do research. First, thank you to my coauthors. John Jasa and Justin Gray contributed important ideas for the thermal optimization in Chapter 5. Anil Yildirim, Josh Anibal, and Sandy Mader helped me get up and running with the MACH framework and greatly reduced my pain and suffer-

ing in obtaining the Chapter 8 results. Eytan Adler has been a valuable ally in the Chapter 7 trade study — OpenConcept will be in good hands. While our direct work together was relatively brief, Andrew Lamkin's pyCycle wizardry greatly accelerated the Chapter 7 study as well. Neil Wu and Eirikur Jonsson are the glue that holds the lab's codes together.

While all of my MDOlab colleagues have been a joy to work with, a few hold an especially prominent place. John Jasa and Shamsheer Chauhan were particularly generous mentors. Anil Yildirim has been not just a prolific source of research ideas but also a great friend who has helped me place the various challenges of grad school in perspective.

I would also like to thank the NASA OpenMDAO team for their efforts in building this revolutionary tool, as well as the AFRL team. Special thanks also to Rodger Dyson at NASA Glenn who supported Eytan and me in the continued development of OpenConcept. The empirical validation in Chapter 6 would have been impossible without Pipistrel graciously allowing me to use their flight test data — in particular, I wish to thank Jernej Pinar who assembled the dataset, as well as Blaž Močan and Tine Tomažič.

Portions of this dissertation were supported by the National Science Foundation Graduate Research Fellowship Program under Grant DGE 1256260. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. This work was also supported in part by the U.S. Air Force Research Laboratory (AFRL) under the Michigan–AFRL Collaborative Center in Aerospace Vehicle Design (CCAVD). I also acknowledge the Texas Advanced Computing Center (TACC) at The University of Texas at Austin for providing HPC resources that have contributed to the research results reported within this paper.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE

LIST OF TABLES

TABLE

LIST OF APPENDICES

APPENDIX

LIST OF ACRONYMS

AC	alternating current
AD	automatic differentiation
ADODG	Aerodynamic Design Optimization Discussion Group
AIAA	American Institute of Aeronautics and Astronautics
ANK	approximate Newton–Krylov
API	application programming interface
ASO	aerodynamic shape optimization
BFL	balanced field length
BLI	boundary layer ingestion
BPR	bypass ratio
CAD	computer-aided design
CESTOL	cruise-efficient short takeoff and landing
CFD	computational fluid dynamics
CFR	Code of Federal Regulations
CFRP	carbon fiber reinforced polymer
COP	coefficient of performance
CPU	central processing unit
CRM	common research model
CSV	constant system voltage
DC	direct current
DEAP	Distributed Electrical Aerospace Propulsion
DIE	discrete induced exponential
DOD	depth of discharge
DOE	design of experiments
EASA	European Aviation Safety Agency
EIS	entry into service
EP	electric propulsion
ESP	Engineering Sketch Pad
eVTOL	electric vertical takeoff and landing
FAA	Federal Aviation Administration
FCEV	fuel cell electric vehicle
FEM	finite element method
FFD	free-form deformation
GPU	graphics processing unit

GTF geared turbofan
HALE high-altitude long-endurance
HPC high-performance computing
HTS high-temperature superconducting
HWB hybrid wing body
ICAO International Civil Aviation Organization
KS Kreisselmeier-Steinhauser
LEAPtech Leading Edge Asynchronous Propellers Technology
LPC low-pressure compressor
MACH MDO of Aircraft Configurations with High Fidelity
MDAO multidisciplinary design analysis and optimization
MDO multidisciplinary design optimization
MDP multi design point
MIP mixed integer programming
MKL Math Kernel Library
ML machine learning
MTOW maximum takeoff weight
NACA National Advisory Committee for Aeronautics
NAE National Academy of Engineering
NASA National Aeronautics and Space Administration
NATO North Atlantic Treaty Organization
NPSS Numerical Propulsion Simulation System
ODE ordinary differential equation
OEI one-engine-inoperative
OEM original equipment manufacturer
OEW operating empty weight
OML outer mold line
OPR overall pressure ratio
OR operations research
PCA principal components analysis
PSFC power specific fuel consumption
RANS Reynolds-averaged Navier–Stokes
RTO rolling takeoff
SCEPTOR Scalable Convergent Electric Propulsion Technology and Operations Research
SFCL superconduction fault current limiters
SIMD single-instruction multiple-data
SLS sea-level static
SNOPT Sparse Nonlinear Optimizer
SOA state of the art
SOC state of charge
STL stereolithography file
STOL short takeoff and landing

SUGAR	Subsonic Ultra-Green Aircraft
TACC	Texas Advanced Computing Center
TCDS	type certification data sheet
TMS	thermal management systems
TOC	top-of-climb
TOGW	takeoff gross weight
TOW	takeoff weight
TSFC	thrust specific fuel consumption
UAS	unmanned aerial system
VSV	variable system voltage
VTOL	vertical takeoff and landing

LIST OF SYMBOLS

Latin Symbols

A	Area
c	Chord length, cost
c_p	Specific heat (isobaric)
C	Elec. charge, thermal conductance, various coefficients
C^0, C^1	Function continuity
C_D, C_L, C_M	Drag, lift, pitch moment coefficient
C_{fg}	Gross thrust coefficient
C_P	Propeller power coefficient
COP	Coefficient of performance
d, d_h	Diameter, hydraulic diameter
D	Drag force
e	Specific energy
E	Energy
$f(\square)$	Function of a variable, objective function
f	Darcy friction factor
g	Gravitational const., optimization constraint
h	Altitude, convective heat transfer coefficient
$\mathscr{H}$	Height
H	Degree of hybridization
I	Electric current
J	Propeller advance ratio
k	Thermal conductivity
KS	Kreisselmeier-Steinhauser function
ℓ	Length
L	Lift force
m	Mass (colloquially, "weight" where unambiguous)
$\dot{m}$	Mass flow rate
M	Mach number
n	Number or quantity, propeller rotational speed
Nu	Nusselt number
p	Specific power, pressure
P	Power
q	Heat transfer rate (direction as noted)
r	Spatial variable

R	Range, distance, radius
$\mathcal{R}$	Residual equation
Re	Reynolds number
SOC	State of charge
S_{ref}	Wing reference area
t	Time, thickness (where subscripted)
T	Thrust, temperature (where subscripted)
T_4	Turbine inlet temperature
$\bar{T}$	Volume averaged temperature
U	Overall heat transfer coefficient
V	Airspeed, volume
$V_1, V_R, V_{\text{stall}}$	Decision, rotation, stall airspeeds
$\dot{V}$	Volumetric mass flow rate
W	Weight force
$\dot{W}$	Work rate (mechanical power)
$\mathscr{W}$	Width
x	Distance (in x coordinate direction); design variable

Greek Symbols

α	Angle of attack, tuning parameter in discrete induced exponential function
β	Chiller bypass parameter
γ	Flight path angle
Δ	Change in a parameter
ϵ	Heat exchange effectiveness, a small tolerance
η	Efficiency (see subscript)
Θ	Throttle parameter
μ	Coefficient of rolling friction
ξ	Pressure loss factor
ρ	Density, tuning parameter in KS function
σ	Stress
Φ	A semi-empirical function

Subscripts

0	Stagnation or total quantity
1	Decision (as in V_1)
act	Actual
b	Battery
bus	Electrical power bus
comp	Component
cond	Conductive

conv	Convective
cycle	Relating to a thermodynamic cycle
des	At the design point
D	Drag (as in C_D)
E	Energy
e	Electrical system
exit	Quantity measured at nozzle exit
empty	Quantity measured while airplane empty
eng	Engine
f	Fuel
fg	Gross thrust (as in C_{fg})
g	Generator
hex	Heat exchanger
i	Inverter
in	Directionality in from someplace else
∞	Free stream quantity
int	Integration
L	Lift (as in C_L)
m	Motor
M	Moment (as in C_M)
max	Maximum
min	Minimum
nom	Nominal
OC	Open circuit
p	Propulsive
P	Power
prop	Propeller
r	Rectifier, radial
R	Rotation (as in V_R)
rated	Rated (generally, maximum continous design value)
req	Required
struct	Structural
theo	Theoretical
TO	Takeoff
tot	Total (not stagnation quantity)
TOW	Takeoff weight
ts	Turboshaft

Others

$\boldsymbol{x}$	Vector quantity

ABSTRACT

Electric aircraft propulsion is an intriguing path towards sustainable aviation, but the technological challenges are significant. Bulky and heavy electrical components such as batteries create spatial integration and aircraft performance challenges, especially for longer-range aircraft. A common thread among all aircraft with electric propulsion is the close coupling of aircraft design disciplines, such as aerodynamics, structures, propulsion, controls, and thermal management. Multidisciplinary design optimization (MDO) is a promising technique for solving design problems with many closely-coupled physical disciplines.

The first half of this dissertation focuses on MDO of electric aircraft considering systems modeling. First, design of electric aircraft is reviewed in detail from the perspective of the various disciplines. Next, methods and models for electric aircraft propulsion systems are introduced. A case study involving a general aviation airplane is explored in order to validate the performance of the methods and generate some insight into the tradespace for series hybrid aircraft. The systems modeling approach is then extended to include basic thermal management systems. The prior case study is revisisted while considering thermal constraints. Impact of thermal management on aircraft performance is assessed. The thermal management analysis methods are validated using flight test data from the Pipistrel Velis Electro, finding good agreement between experiment and simulation. Finally, an MDO model of a parallel hybrid electric transport aircraft with a liquid-cooled thermal management system is constructed. Sensitivities of aircraft performance with respect to important technologies parameters are computed.

This first half introduces the first publicly-available simulation tool that can handle unsteady thermal states and that offers efficient and accurate gradients. The methods are very efficient, en-

abling users to perform dozens or hundreds of optimization runs in a short amount of time using modest computational resources. Other novel contributions include the first empirical validation of thermal management models for MDO against real flight test data, as well as the only comprehensive look so far at the unsteady thermal management of a transport-scale parallel hybrid aircraft.

The second half of the dissertation introduces novel methods for performing high-fidelity shape optimization studies subject to packaging or spatial integration constraints. A new mathematical formulation for generalized packaging constraints is introduced. The constraint formulation is demonstrated on simple aerodynamic shape optimization test cases. Next, a wing design study involving optimal battery packaging is conducted in order to demonstrate the coupling of outer mold line design and propulsion system component design via spatial integration. Finally, a more complex aerostructural optimization involving the wing of a hydrogen aircraft is constructed and solved. These test cases demonstrate the interdisciplinary coupling introduced by packaging constraints, as well as the impact of spatial integration on aircraft performance.

This latter half contributes a powerful new way for MDO engineers to pose realistic spatial constraints in their shape optimization problems, thus solving an important practical barrier to the industrial adoption of MDO for certain relevant problems. This work also represents the first time an MDO problem has been posed and solved for an aircraft using hydrogen fuel in the wing. Altogether, this dissertation significantly advances the state of the art in modeling, simulation, and optimization tools for aircraft with electric propulsion architectures and introduces new insights into the design spaces for several diverse aircraft configurations.

CHAPTER 1

Introduction

1.1 The Electric Age of Aviation?

A confluence of external forces has primed the field of aeronautics for rapid change. The COVID-19 pandemic altered the economic landscape in commercial aviation, weakening the financial position of incumbent manufacturers and airlines. At the same time, investment flowed into aerospace startups, giving them a cash infusion needed to get from a proof of concept to certification. Technologies initially developed in other fields have now become practical for aerospace applications, such as large-scale lithium-ion batteries and lightweight, efficient electric motors. Solar and wind power continue to decline rapidly in price, and renewable energy sources are finally cheaper than natural gas and coal in certain markets. For the first time, airlines have faced mainstream public campaigns to reduce carbon emissions (such as the *flygskam* movement in the Nordic countries).

This confluence of forces has resulted in a burst of activity in the area of sustainable aviation, and in particular, electric aviation. The number of startups developing electric vertical takeoff and landing (eVTOL) aircraft for urban air mobility is nearly uncountable. As of this writing, three startups have received valuations over one billion USD: Joby Aviation, Archer Aviation, and Lilium. Other potential competitors in the eVTOL space include huge incumbents such as Airbus and Bell Helicopter.

The fixed-wing electric aircraft market is not nearly as crowded but arguably closer to a viable product. The Pipistrel Velis Electro has been certified in Europe as a primary trainer aircraft under the Light Sport Aircraft category. Bye Aerospace is developing similar two- and four-seat aircraft for training and recreation. MagniX is pursuing certification for an electric motor sized for retrofit applications in aircraft such as the DHC-2 Beaver and Cessna Caravan. What these concepts lack in range they potentially recover in low operating costs and sustainability. Alternative approaches using hybrid-electric, turboelectric, or hydrogen architectures are also being actively pursued. ZeroAvia, Boeing, and H2Fly have flown manned fuel cell electric vehicle (FCEV) demonstrators, and Universal Hydrogen and ZeroAvia have announced plans to commercialize the technology. Electra.aero is constructing a manned flight demonstrator with a distributed hybrid-electric architecture designed for very short takeoff and landing (STOL), while Airflow pursues a very similar configuration for logistics applications.

Aviation historians often mark the introduction of an important new technology as the beginning of an "age". The interwar period following the invention of the metal monoplane became known as the Golden Age of Aviation. After World War II followed the Jet Age. If the startup founders are right, we are witnessing the beginning of the Electric Age.

1.2 Open Questions

Despite widespread optimism about continued progress in electric aviation, significant technical challenges loom, both in the short term and on the horizon. While the pace of progress is accelerating, the industry as a whole simply does not have much service experience with electric propulsion architectures. Thus, we cannot rely on historical data typically used in early stages of aircraft design to estimate important parameters (such as empty weight). Estimates of aircraft performance must be built from the bottom up using reasoning rooted in the physics. Furthermore, it is not controversial that it will be challenging for battery-electric aircraft to fly reasonable and

useful distances. Careful attention must be paid not only to the propulsion system, but also to the aerodynamics and structures, if electric aircraft are to reach their full potential.

In the absence of historical data, and where weight and drag are critical, there is clearly a role for computational models and optimization. Computational models of increasing fidelity can produce the required bottom-up performance estimates. Optimization can reduce the structural weight, drag, and propulsive losses, allowing electric aircraft to make the most of their limited energy source.

However, at the time I started the research detailed in this dissertation, computational models and optimization applications for electric aircraft had glaring limitations. Nearly all authors neglected to model the physics of removing waste heat from the electric propulsion system, and most models were incompatible with the gradient-based optimization methods necessary for solving large problems or performing broad tradespace sweeps. The issue of spatially integrating (packaging) the bulky electric propulsion system into an optimized outer shape was totally unexplored in the public literature. Finally, most of the higher-quality modeling frameworks were closed-source and proprietary in some way, resulting in needless duplication and wasted effort across industry and academia.

Thus, I identified the two central research objectives of this dissertation:

1. Demonstrate *gradient-based* optimization of electric aircraft with aircraft systems modeled at an appropriate level of fidelity (including *transient thermal effects*)

2. Develop (and demonstrate) methods for performing high-fidelity MDO subject to general packaging or spatial integration constraints

Both objectives necessarily require developing new computational tools. It was a high priority that the computational tools be transferable to industry and others in academia. Therefore, the software must be open source, high quality, and validated.

1.3 Overview of the Dissertation

This dissertation is roughly divided into two parts, preceded by a "prelude" and followed by a "postlude". The prelude introduces the reader to electric aircraft design principles and reviews the commercial and academic landscape, as well as presenting a brief introduction to MDO. Part One focuses on MDO of electric aircraft considering *systems modeling*. I introduce an open-source computational framework which enables users to model and optimize aircraft propulsion systems, with a particular focus on managing waste heat. I validate the models using flight-test data from the Pipistrel Velis Electro and explore two optimization case-studies: one for a general aviation airplane, and the other for a transport aircraft. In Part Two, I shift focus to the problem of *packaging optimization*, introducing a new theoretical approach to posing spatial constraints in MDO and demonstrating the technique on two relevant electric aircraft wing design problems. The postlude offers concluding remarks and includes computational details of two open-source software packages I developed for this dissertation.

CHAPTER 2

A Comprehensive Review of Electric Aircraft Design

2.1 Historical Context

In the last decade, aircraft concepts using electricity for some or all of their propulsive power have captured the public imagination and garnered great attention in the popular press. Several start-up ventures have formed to commercialize aircraft EP. Electric propulsion has also become a topic of widespread study in academia. From 2006 to 2009, there was about one paper per year on electric and hybrid electric aircraft design and analysis. By 2017 the volume of similar papers increased to more than 20 per year. In 2018 the Electric Aircraft Technologies Symposium (EATS) emerged as an annual venue exclusively dedicated to the topic, and in 2019 the North Atlantic Treaty Organization (NATO) Science and Technology Organization held a special symposium on military applications of electric and hybrid propulsion.

A major factor motivating public interest in electrification is the urgent need to reduce environmental impact. Recognizing the urgency of climate change and aviation's sizable contribution to global carbon emissions, governments and international organizations have established voluntary and mandatory targets for emissions reductions. The International Civil Aviation Organization (ICAO) has established certification standards for noise and NO_x in the 2020s [2] and a voluntary carbon offset scheme aiming to hold overall sector carbon emissions at 2020 levels [3]. National Aeronautics and Space Administration (NASA)'s Subsonic Fixed Wing program estab-

lished aggressive goals for energy consumption, nitrogen oxides (NO_x), and noise for three generations of airplanes extending out to the 2030s [4, 5]. The most aggressive performance targets are for the "N+3" generation with projected entry into service (EIS) dates in the mid-2030s: -55 dB noise at the airport boundary, -75% NO_x, and -70% fuel burn relative to 2006-era technology. NASA-funded study results (discussed in Section 2.3) indicate that electrification can improve performance on carbon, noise, and NO_x, enabling the civil aviation fleet to meet N+3 goals [5].

There are also potential economic benefits to electrification which can be divided into two parts: first, reduced operating costs compared to conventional aircraft on existing missions; and second, completely new capabilities that may open new and lucrative markets. Reduced operating costs can be achieved through replacement of jet fuel with electricity, through a reduction in total energy consumption, or through reduced maintenance costs. Figure 2.1 shows that jet fuel has generally been more expensive than electricity over the long term, although in the more recent past the gap has been less pronounced.

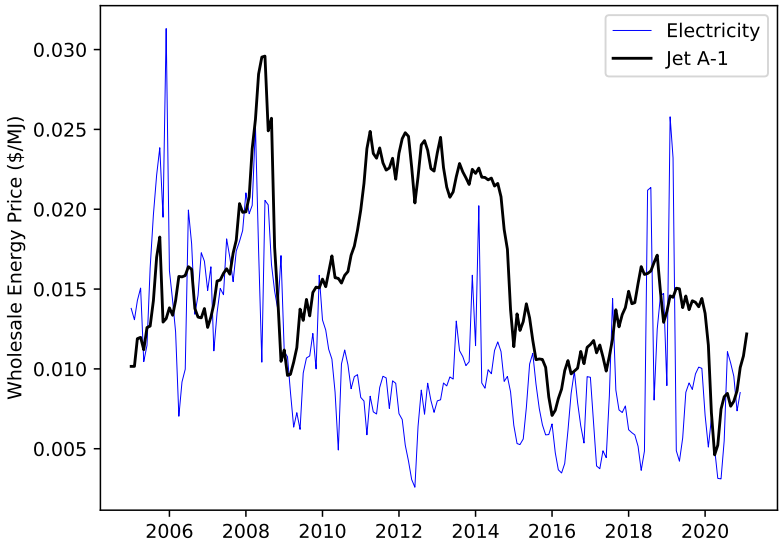


Figure 2.1: Normalized energy cost of Northwest U.S. wholesale electricity versus U.S. Jet A-1 (Data from [6, 7])

Electrification may also enable concepts of operations that are not currently served with conventional aircraft propulsion architectures. eVTOL concepts have been launched by numerous start-up and incumbent firms worldwide, including VoloCopter, Ehang, Wisk, Joby Aviation, and Airbus. Technology and transportation firm Uber released the "Elevate" white paper in 2016, arguing that a sizable market exists for point-to-point urban air mobility, catalyzing activity in this new segment [8]. For noise and cost reasons, eVTOL proponents argue that traditional helicopters are not a suitable architecture for this application. This dissertation (and thus, this review) focuses on fixed-wing aircraft, though many of the principles also apply to vertical takeoff vehicles as well.

There are several published survey articles providing partial coverage of the fixed-wing aircraft EP field. Thomson et al. [8] provide a particularly broad and readable, though non-technical, summary of aviation electrification from a business perspective. Hepperle [9] presents an overview of EP architectures and some basic sensitivity analyses based on the Breguet range equation. Pornet [10] covers practical conceptual design considerations of hybrid electric passenger aircraft using lower-order sizing methods and graphical methods, but is missing coverage of higher-fidelity optimization tools and a comprehensive survey of design studies and demonstrator programs. A U.S. National Academy of Engineering (NAE) subcommittee published a study report evaluating underlying EP technologies and making recommendations on high-level research priorities [11]. Several other reviews cover aircraft EP as a sidebar to another primary topic. Gohardani et al. [12] review distributed propulsion with an extended discussion of EP; Gohardani [13] later updated and expanded the review. Sarlioglu and Morris [14] present an excellent review of more-electric aircraft systems that includes a sidebar on propulsion. Perullo and Mavris [15] focus only on higher-fidelity modeling of energy management in hybrid configurations; Wall and Meyer [16] likewise focus only on hybrid electric.

In spite of all the work cited above, there was a need for a review article that provides an entry point to the field of EP for aircraft designers, modelers, and technologists, who are versed in aircraft design principles but do not necessarily have an electrical background. I addressed this

need by publishing [17], a comprehensive review article covering EP fundamentals, concepts and demonstrators, technologies, practical design trades, and simulation capabilities. This chapter and Chapter 3 contain substantially condensed and updated material from [17].

2.2 Aircraft Electric Propulsion Fundamentals

2.2.1 Classification

Aircraft can be categorized based on the *degree of hybridization* of their power (H_P) and energy (H_E) sources. A definition of hybridization with respect to power and energy was developed by Isikveren et al. [18], namely:

$$H_P = \frac{P_m}{P_{\text{tot}}} \tag{2.1}$$

$$H_E = \frac{E_b}{E_{\text{tot}}} \tag{2.2}$$

where P_m is the motor power, P_{tot} is the total propulsive power, E_b is the battery energy, and E_{tot} is the total energy storage including fuel. By convention, hybridization is usually given with respect to P_m and E_b, though in principle the same method of analysis could be used for other sources, such as hydrogen fuel.

Conventional aircraft use no electric power or electric energy for propulsion ($H_P = 0$, $H_E = 0$). *All-electric* aircraft (Figure 2.2a) use exclusively electrical energy and power for propulsion

Table 2.1: Classification of electric propulsion architectures

Architecture	H_P	H_E	Diagram
Conventional	0	0	
All-Electric	1	1	2.2a
Turboelectric	>0	0	2.2b
Series Hybrid	1	<1	2.2c
Parallel Hybrid	<1	<1	2.2d

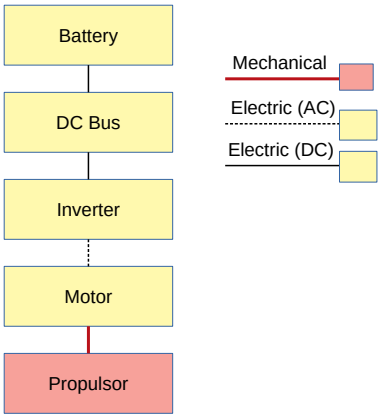
($H_P = 1$, $H_E = 1$). Some authors prefer the term *universally-electric*. *Hybrid electric* aircraft rely on a mix of fuel and electrical energy storage and propulsive power ($H_P > 0$, $0 < H_E < 1$). Finally, *turboelectric* aircraft (Figure 2.2b) use combustible fuel for energy storage but electrical power transmission instead of mechanical power to drive the propulsor(s) ($H_P > 0$, $H_E = 0$). Hydrogen aircraft are not typically described using the H_P/H_E formalism, but a hydrogen fuel cell aircraft could be said to have $H_P = 1$ and hydrogen combustion $H_P = 0$.

Hybrid electric architectures are broadly similar to hybrid automobiles but have some important distinctions. Non-plug-in hybrids like the Toyota Prius generate electrical power only through regenerative braking. For transport-class aircraft, regenerating power during descent is less efficient than using a non-regenerative continuous descent trajectory [19, 20]. Proposed hybrid aircraft concepts operate more like the plug-in hybrid Chevrolet Volt, where electricity from grid power may be used for short trips and fuel is used as a *range extender*. Turboelectric designs are simply hybrids without significant battery storage.

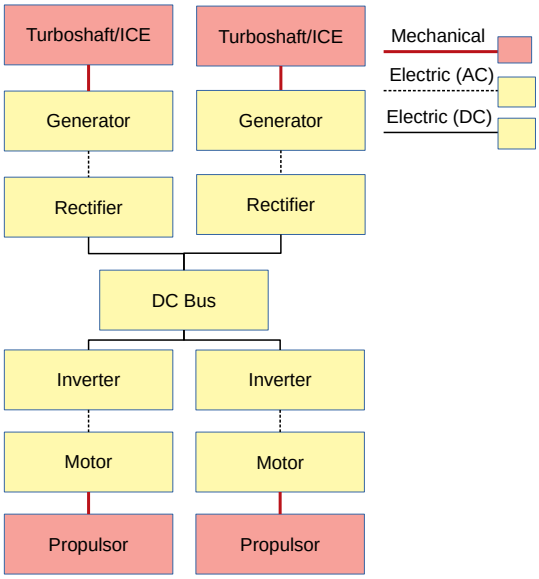
Hybrid electric architectures can be further divided into parallel and series hybrids. *Series hybrid* designs (Figure 2.2c) generate electrical power using a combustion engine and deliver both battery and fuel energy to the propulsor via electrical buses ($H_P = 1$, $0 < H_E < 1$). *Parallel hybrid* architectures (Figure 2.2d) deliver combustion power to the propulsor mechanically ($H_P < 1$, $0 < H_E < 1$). The combustion engine may operate continuously and use electrical power to reduce fuel flow [19], or the engine may disconnect via a clutch to enable full-electric operation during some portion of the flight envelope [21]. Finally, the Ampaire Cessna 337's push-pull configuration with one combustion engine and one electric motor operating independently can be conceptualized as a type of "parallel hybrid", though it is really in a category of its own.

2.2.2 Key Technological Parameters

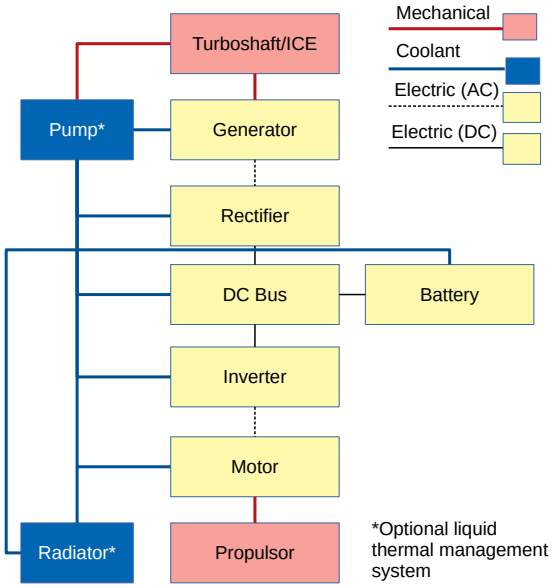
When evaluating the feasibility of aircraft EP today, the two most important technological parameters are (arguably) *specific energy* (energy per unit mass of energy storage) and *specific power*



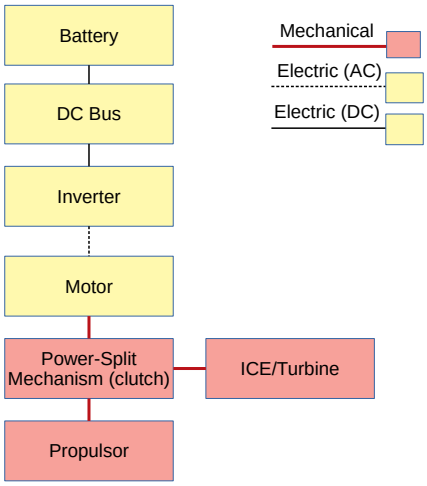
(a) All Electric



(b) Twin Turboelectric



(c) Series Hybrid (with Liquid Cooling)



(d) Parallel Hybrid

Figure 2.2: Notional electric propulsion architectures

(power of a component per unit mass). Specific energy is especially applicable to batteries, while specific power is especially relevant for electric motors, power conversion electronics, and energy storage devices. There does not seem to be a consensus in the literature on how to abbreviate specific energy and specific power symbolically. Like Pornet et al. [22], I adopt the convention of fluid dynamics, representing specific quantities as the lowercase of the extensive quantity: battery specific energy as e_b, fuel specific energy as e_f, motor specific power as p_m, and so on. By convention, e_b is tabulated in units of Whr/kg, specific power in units of kW or MW/kg, and heat values (e_f) of fuel in MJ/kg.

The most fundamental challenge facing aircraft EP is that batteries have on the order of 50 times lower specific energy than liquid fuels. For Jet-A, $e_f = 11900$ Whr/kg, while lithium-ion batteries are in the range of $e_b = 200$ Whr/kg [23]. The critical impact of specific energy is illustrated through the Breguet range equations. For consumable fuels,

$$R_f = \frac{L}{D}\eta_p\eta_{int}\eta_{eng}\frac{e_f}{g}\ln\left(\frac{1}{1-m_f/m_{TO}}\right). \tag{2.3}$$

where L/D is the lift-to-drag ratio, η_p is the propulsive efficiency, η_{int} is the efficiency due to propulsion integration losses, η_{eng} is the engine thermal efficiency, e_f is the fuel specific energy, and m_f/m_{TO} is the ratio of fuel weight to takeoff gross weight.

For battery-powered aircraft, a different version of the Breguet range equation can be derived as [9, 24]

$$R_b = \frac{L}{D}\eta_p\eta_{int}\eta_e\frac{e_b}{g}\frac{m_b}{m_{TO}}. \tag{2.4}$$

where η_e is the total efficiency stackup of the electric propulsion system, e_b is the battery specific energy, and m_b/m_{TO} is the ratio of battery weight to takeoff gross weight. These equations differ due to the decrease in fuel mass (and therefore, induced drag) during a mission for fuel-powered aircraft.

For both fuel-burning and battery-powered configurations, range is directly proportional to specific energy (e_b, e_f). At state of the art (SOA) battery specific energies, aircraft with significant H_E will have very short useful ranges or very poor payload fractions. For a realistic set of mission requirements (including a given design range), low e_b leads to dramatically heavier takeoff gross weight (TOGW) and higher total energy consumption. The effect of specific energy is mission-, scale-, and platform-dependent, but sensitivities obtained for particular concepts can be found in the literature [25–28].

Since e_b is important to other industries (such as automotive and consumer electronics), academic and industrial research in higher-specific-energy batteries is ongoing. For aviation, advanced concept studies require an estimate of battery technology available near the first-flight date. Projecting future battery technology is outside the scope of this dissertation, but study authors are assuming e_b on the order of 400–1000 Whr/kg for advanced concepts, depending on the time frame [23, 29]. Individual e_b assumptions for specific studies are tabulated in Table 2.3. The SOA of energy storage technologies, including alternative battery chemistries, is discussed by Rheaume and Lents [23]. The NAE committee report projected that e_b will reach 400–600 Whr/kg by 2035 [11].

Volumetric energy density is a related consideration for energy storage devices. Jet fuel is Pareto optimal in e_f and volumetric energy density. While compressed hydrogen has very high e_f, its volumetric density is very low. Lithium-ion batteries also have significantly lower volumetric energy density than jet fuel, but the performance gap is a less-acute problem than e_b [9].

Specific power of electrical devices (p_e), including motors/generators and power conversion, is another key technological parameter in the design problem [29]. Since no single number captures the full operational envelope of an electrical device, the power considered in this metric could be rated power, maximum power, or design power. Electric motors are often rated for short-term burst power, which is usually substantially higher than the maximum continuous power; the analogous rating for a turbofan is takeoff thrust and maximum continuous thrust, respectively.

Unless otherwise specified, motor specific power (p_m) values listed in this dissertation refer to short-term burst power.

In all-electric and series hybrid architectures, electrical power systems must be rated for the entire propulsive power of the airplane. If p_e is too low, the airplane will be too heavy, possibly resulting in increased fuel burn compared to a conventional design. The electrical system weight appears in the empty weight component of m_{TO} in the range equations (2.4) and negatively impacts range.

Like turbines, fuel cells convert liquid fuel to power [9]. Fuel cells do so with high efficiency, but their poor specific power historically made them uncompetitive with batteries or turbogenerators except in niche applications where hydrogen is available and low thermal signature is paramount [23]. However, recent advances in fuel cell technology have resulted in renewed interest by startups including Universal Hydrogen and ZeroAvia.

Specific power of batteries p_b can be an important constraint on the design problem. Certain portions of the flight envelope, such as takeoff, may require substantially more power than cruise. This problem is particularly apparent for eVTOL aircraft, particularly those with high disc loading such as the Lilium Jet. Batteries must be capable of discharging quickly enough to meet this demand over the duration of the maneuver; unfortunately, high p_b strictly trades off with e_b [30]. See Section 2.5.2 for a more detailed discussion of battery characteristics and design trades.

The efficiency of electrical components is a third key technological factor. The overall electrical efficiency for a direct current (DC) turboelectric configuration can be written as,

$$\eta_e = \eta_g \eta_r \eta_{bus} \eta_i \eta_m, \tag{2.5}$$

where $\eta_{g/r/bus/i}$ are the efficiencies of the generator, rectifier, bus, and inverter, and η_m is the motor efficiency. I define these components in more detail in Section 2.5. All-electric and parallel hybrid designs do not require a generator or rectifier. Using an alternating current (AC) architecture elim-

inates the rectifier and inverter. Multiplied together, the electrical efficiencies are analogous to the thermal efficiency of a combustion engine. The propulsive efficiency η_p is considered separately. Pornet et al. [22] provides a good discussion of electrical component efficiency stackups and power constraints for a series hybrid aircraft.

Unlike specific energy, efficiencies are bounded at 1 and are already relatively large, so the influence on aircraft range is not as strong as that of e and p. The larger problem comes from thermal management of waste heat inside the aircraft. If we assume that all electrical inefficiency results in resistive heating, improving η_e from 97% to 99% results in 2% higher range but a 67% reduction in waste heat. Efficiencies of specific types of power electronics are addressed in Section 2.5 and TMS in Section 2.6.

2.3 Products, Prototypes, and Concepts

At least 17 manned, electric, fixed-wing aircraft have flown since 2000, three of which are commercially available products. Two more technology demonstrators are reasonably expected to fly by 2020. There are also numerous industry- and government-funded advanced concept studies focused on higher technology and power levels. This section reviews most of the flight-tested electric aircraft and the well-developed studies.

Table 2.2 lists noteworthy manned electric aircraft with first-flight dates since 2000. Table 2.3 lists major design studies by government, established industry firms, and start-ups. The National Academy report includes a related table focusing on technology targets for batteries, motors, and generators [11].

Table 2.2: Summary of flyable, manned electric aircraft

Name	1st flight yr	Arch. [1]	Seats	TOGW (kg)	Max power (kW)	e_b (W hr/kg)	Range (nmi)/ Endur	Remarks	Ref
Lange Antares 20E	2003	E	1	660	42	136	NA	1st elec. aircraft w/ airworthiness cert.; commercially avail. motor-glider	[31]
Fishman Electraflyer C	2008	E	1	283	13.5		90min	Converted motorglider; Li-Po battery	[32]
Boeing HK-36 FCD	2008	FC	1	860	75	NA	45min	30 kW fuel cell; experimental	[33]
Yuneec E430	2009	E	2	470	40	154		Clean-sheet composite airframe; commercialization abandoned	[12, 34]
Siemens/Diamond E-Star	2011	SHE	2	800	70			30 kW Wankel engine, experimental	[35]
Pipistrel Taurus Electro G2	2011	E	2	450	40			Commercially avail.	[36]
Pipistrel Taurus Electro G4	2011	E	4	1500	150	180	244	Won NASA Green Flight Challenge; experimental; 400+ pmpg	[31, 37]
IFB Stuttgart eGenius	2011	E	2	950	60	204	244	Competed in NASA GFC; experimental	[37]
Embry-Riddle Eco-Eagle	2011	PHE	2	1075	105	125	170	Unofficial participant in GFC; 75 kW rotax, 30 kW elec; exp.	[31, 38]
Fishman Electraflyer ULS	2012	E	1	238	15		120min	Commercially-available under US ultralight rules	[32]
Chip Yates Long ESA	2012	E	1	680	192			Experimental	[39]
Siemens/Diamond E-Star 2	2013	SHE	2	800	80			5 kW/kg motor; experimental	[40]
Airbus E-Fan	2014	E	2	600	60	207	60min	2x30 kW fan; experimental	[41, 42]
Cambridge SOUL	2014	PHE	1	235	20	144		12 kW elec, 8 kW petrol; recharges in flight; exp.	[43, 44]
Pipistrel Alpha Electro	2015	E	2	550	60	171	70	Commercially avail.	[45, 46]
Airbus E-Fan 1.2	2016	SHE	2	600	60			2x30 kW fan, 50 kW range extender; experimental	[42]
Siemens Extra 300 (330LE)	2017	E	1	1000	260			95% eff. motor, >5 kW/kg incl. inverter, 580VDC; exp.	[47]

[1] E=All-electric; SHE/PHE=Series/parallel hybrid; TE=Turboelectric; FC=Fuel cell

Table 2.3: Summary of electric fixed-wing aircraft concepts and studies

Name	Target EIS yr	Arch. [2]	Seats	TOGW (kg)	Max power (MW)	e_b (W hr/kg)	Design Range (nmi)	Remarks	Ref
NASA X-57 "Maxwell"	2018	E	2	1360	0.144	130		2x72 kw tip motors; manned demonstrator; leading-edge DP	[48–50]
NASA STARC-ABL	2035	TE	154	60000	2.6 [3]		3500	-9.4% fuel burn; tube/wing config. w/ tailcone boundary layer ingestion (BLI) prop.	[51, 52]
NASA N3-X	2045	TE	300	227000	50		7500	-10% FB due to EP; supercond.; hybrid wing body (HWB) w/ BLI dist. prop.	[29, 53–58]
Boeing SUGAR Volt	2035	PHE	154	68040	1.0 [4]	750	3500	-10.9% FB due to EP; Strut-braced wing w/ battery pods	[19]
Bauhaus Luftfahrt Ce-Liner	2035	E	189	109300	33.5	2000	900	C-wing, supercond. motors, aggressive tech assumptions	[59, 60]
Airbus VoltAir	2035	E	∼33	∼33000		750+	∼900	Superconducting electronics; BLI; laminar-flow wing	[61, 62]
Airbus/R-R E-Thrust	2050	SHE	90		9.0	1000	Rgnl	Superconducting; BLI and high bypass ratio; embedded fans	[63]
ESAero/Wright ECO-150R	2035	TE	150	60-75k	12.7		1650	FB ≈ 737-700; 16 motors embedded in wing; no supercond.	[64]
Eviation Alice	2019	E	11	6350	0.780	260	560	3 pusher props; Kokam Li-Ion batteries	[65, 66]
XTI Tri-Fan 600	2024	SHE	6	2404	1.5		1200	VTOL; tilt-fan configuration	[67]
Ampaire Tailwind	2020s	E/SHE					350	Tail-mounted boundary layer ingestion thruster	[68]
Zunum	2020s	SHE	12	5216	1		700	Conventional regional jet layout; straight NLF wing	[69]

[2] E=All-electric; SHE/PHE=Series/parallel hybrid; TE=Turboelectric; FC=Fuel cell

[3] tailcone propulsor power only

[4] includes electric motor only

2.3.1 Flight Test Vehicles, Prototypes, and Commercial Products

The experimental history of electric aviation dates back to over a century ago, when Santos-Dumont devised an electric airship [31]. In the 1980s and 1990s, very high aspect ratio solar aircraft demonstrations were pursued by NASA and others. This experimental heritage resulted in high-altitude long-endurance (HALE) or "pseudosatellite" systems such as the QinetiQ/Airbus Zephyr. The technical challenges of these aircraft are distinct from manned electric aircraft and include aeroelasticity and control problems related to the excessive aspect ratio and aggressive empty weight requirements. Even at low speeds and power levels, solar cells typically do not justify the additional weight and drag in manned applications [37].

The first manned, electric fixed wing aircraft was the Brditschka MB-E1, which flew for less than 10 minutes in 1973 [26]. Since the 2000s, several self-launching electric motor gliders have been offered for sale. The first to earn a type certificate was the Antares 20E [31]. Several lightweight electric aircraft prototypes by independent designers flew in the late 2000s, including the Electraflyer C, Yuneec 430, and an all-electric modified Cri-Cri [37]. Larger firms then converted demonstrators from glider airframes, including the Boeing HK-36 fuel-cell demonstrator [33], series hybrid Diamond DA-36 E-Star [35, 40], and all-electric Pipistrel Taurus Electro [36].

The NASA/CAFE foundation Green Flight challenge in 2011 inspired three new experimental electric aircraft with extended range, including a four-seat variant of the Taurus Electro, Embry-Riddle Aeronautical University's parallel hybrid Eco-Eagle, and IFB Stuttgart's eGenius [31, 37, 38]. The culmination of these lightweight, manned demonstrators was the Pipistrel Alpha Electro, which was designed as a two-seat trainer aircraft and targeted at the Light Sport Aircraft category [45]. An experimental version of the Alpha Electro was offered for sale in the EU and US. In 2020, the Pipistrel Velis Electro (an evolutionary upgrade of the Alpha Electro) was awarded a type certificate by European Aviation Safety Agency (EASA) under the LSA rules, becoming the first electric aircraft to be type certified by a major regulator. The Velis Electro is expected to

cost dramatically less to operate than piston trainers and does not emit lead pollution from burning 100LL aviation fuel.

Other recent demonstrators have focused on scaling up power to 100 kW and above. Chip Yates set multiple speed and time-to-climb records in an all-electric Long EZ, followed by Siemens AG which installed 260 kW of electrical power in an Extra 300 [39, 47]. The Siemens effort set specific power records for a flight-rated electric motor: 5.2 kW/kg including power conversion with a liquid TMS [47, 70]. In 2019, magniX flew a converted DHC-2 Beaver powered by a 500 kW motor, joined in 2020 by an all-electric Cessna Caravan using the same motor. While these aircraft set records for *propulsive* electrical power, the Boeing 787's electrical system generates over 1 MW of electrical power using mechanical off-takes from turbine engines [71, 72]. To the author's knowledge, this is the highest-power electrical system flying on an aircraft today. However, it would not be surprising to learn that some undisclosed military applications may generate even more power (for mission systems such as radar and directed-energy weapons).

Figure 2.3 shows the progression of tested and proposed manned electric demonstrators. There is a clear linear power/weight trend for battery and turboelectric demonstrators focused on maximum efficiency up to 1500 kg; most of them are based on similar motor glider airframes. Pipistrel's Velis Electro and NASA's upcoming X-57 Maxwell demonstrator fall neatly into this trend. More conventional airframes fall above this trend line, such as the Yates and Siemens speed records and the magniX Beaver and Caravan conversions.

However, recent progress has not been free of obstacles. In late 2017, Airbus, Rolls-Royce, and Siemens announced a partnership to build a hybrid electric flying testbed known as "E-Fan X". Built on a BAe-146 platform, the demonstrator was to replace one of the four turbofan engines with a 2 MW electric motor and was intended to "explore the challenges of high-power propulsion systems, such as thermal effects, electric thrust management, altitude and dynamic effects on electric systems, and electromagnetic compatibility issues" [73]. The platform was targeted for first flight by 2020 [73], but was later placed on hold or cancelled. NASA was anticipated to flight test

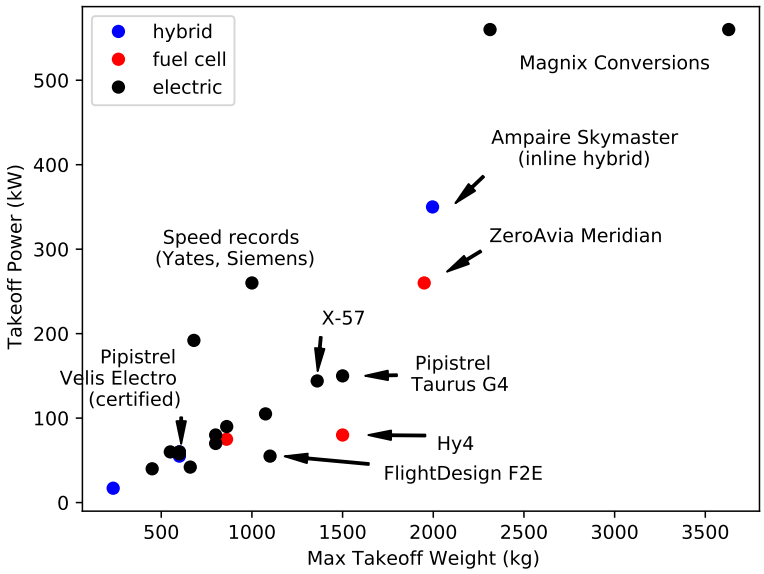


Figure 2.3: Progression of manned electric aircraft demonstrators (data from Table 2.2)

the X-57 Maxwell in 2018, but as of this writing, the aircraft has not yet flown on electrical power.

2.3.2 Concepts and Studies

This section focuses on aircraft designs which have not yet flown and are not imminently expected to fly (except X-57). Conceptual design of aircraft larger than one to four seats started with NASA-funded industry studies beginning in the late 2000s. This resulted in two concepts which have been at least partly published.

The Boeing Subsonic Ultra-Green Aircraft (SUGAR) (Subsonic Ultra-Green Aircraft) series of studies evaluated several evolutionary and revolutionary designs against NASA N+3 goals. This study is particularly valuable because it provides a fair comparison between a hybrid concept and turbofan concepts with equal technology and mission rules. All of the concepts were sized for 900 nmi economic missions with 154 seats. The concepts were a tube-and-wing baseline with current technology (SUGAR Free); a tube-and-wing architecture with estimated 2030s technologies

(Refined SUGAR); a high aspect ratio strut-braced tube-and-wing design with 2030s technology (SUGAR High); a hybrid-wing-body (HWB) configuration with turbofans and 2030s technology (SUGAR Ray); and a version of SUGAR High with parallel hybrid electric propulsion (SUGAR Volt, Figure 2.4).

In Phase 1 of the study, SUGAR Volt was the only concept capable of meeting NASA's N+3 fuel burn goal of -70% (compared to the SUGAR Free baseline). The Volt used 28% less fuel than the conventionally-powered SUGAR High [74]. In Phase 2, hybrid electric propulsion was extended to the conventional tube-and-wing and HWB configurations, and similar fuel burn improvements were found (25%–46% better than conventional propulsion for the HWB; 33%–55% better for the tube-and-wing) [19]. However, the study assumed a 750 Wh/kg battery would be available by 2035, which looks like an aggressive assumption with the benefit of hindsight.

Figure 2.4: Boeing SUGAR Volt concept
(NASA/Boeing image)

A higher-fidelity model of the SUGAR Volt, including a refined hybrid electric/turbofan model from Georgia Tech, reduced the estimated hybrid electric fuel burn increment to between -10.9%

and -21.7% depending on H_P chosen. Higher H_P was associated with higher total energy consumption but lower fuel burn in this trade study. The study included a detailed discussion of technology development risks, commercialization potential, and an agenda for future development of parallel hybrid technology [19].

The second major series of industry studies on hybrid electric commercial transports was conducted by Empirical Systems Aerospace (ESAero). Two concepts were studied: a single-aisle commercial airplane (ECO-150) [75–77] and dual-use military/civilian transport [78]. Both concepts use a series turboelectric architecture, with 16 electric fans embedded in the wing. This configuration provides some blown lift and is claimed to be structurally efficient [64, 79]. The ECO-150 concept initially showed very large fuel burn reductions versus a current single-aisle benchmark (-44% conventional, -59% superconducting) [58]. A higher-fidelity assessment in 2016 (designated ECO-150R) showed no fuel burn improvement but acknowledged that the airplane could be resized for better fuel burn [64]. Due to the large change in assessed fuel burn at the latest iteration, it does not seem that this study was fully converged yet.

NASA itself is actively researching four concepts at widely varying power scales. The best-developed of these is the Scalable Convergent Electric Propulsion Technology and Operations Research (SCEPTOR) project, which launched in 2014. This project is focusing on rapidly achieving ground and flight test demonstrations of higher power levels and distributed propulsion in a phased approach. SCEPTOR is closely related to a NASA propulsive concept known as Leading Edge Asynchronous Propellers Technology (LEAPtech), which introduces numerous small propellers across the leading edge of the wing [80]. The goal of LEAPtech is to reduce drag by increasing cruise wing loading of general aviation airplanes by 2.5 times. This is achieved by greatly increasing $C_{L\text{max}}$ through blown lift, avoiding the need for complex and heavy multi-element flap systems.

SCEPTOR Mod 1 was a truck-mounted LEAPtech wing model tested at NASA Armstrong in 2015. This experiment was reportedly not successful in producing a dataset useful for quantita-

Figure 2.5: X-57 Maxwell demonstrator
(NASA image)

tive comparison or validation of powered-lift distributed propulsion computational fluid dynamics (CFD) results [81, 82]. An unexplained thrust and power consumption asymmetry at equal commanded RPM merits further investigation and highlights potential controls issues that will need to be addressed in flight demonstrators.

The manned flight test vehicle currently under development, known as the X-57 Maxwell, is a converted Tecnam P2006T airframe (Figure 2.5). SCEPTOR Mod 2 involves replacing the Tecnam's combustion engines with electric motors, powered by lithium-ion battery packs stored in the fuselage [49, 83]. Mod 3 replaces the stock wing with a deliberately-undersized carbon fiber wing, with tip-mounted propulsive motors and dummy distributed-lift motor fairings. Mod 3 will have high wing loading (by design) and uncontrollable one-engine-inoperative yawing moments due to the location of the wingtip propulsors; safety considerations of this evolution are discussed by Papathakis et al. [50] and will restrict operations to a dry lakebed. Mod 4 adds the distributed propulsion motors in a configuration similar to the Mod 1 experiment. Mod 2 ws scheduled to fly in 2018 following tests of a redesigned battery module [84], but as of this writing was still undergoing ground testing in preparation for first flight.

The largest electric transport aircraft design ever seriously studied is the N3-X turboelectric concept, which is similar in range and size to the Boeing 777. The N3-X is a hybrid wing body configuration with distributed turboelectric propulsion on the trailing edge, first derived from the NASA/Boeing cruise-efficient short takeoff and landing (CESTOL) study in 2008 [85, 86]. Subsequently, the concept has been the focus of more detailed analysis and design revision by NASA [53, 55, 56, 87], with refined weights [88] and noise and emissions [89]. Notably, the 2014 noise analysis required a major redesign of the aft body propulsion integration for noise. Rolls-Royce and the University of Strathclyde have collaborated on electrical system trades [54, 90, 91] and system safety analysis of high-power and superconducting electric aircraft [92–94] using the N3-X as a baseline. The N3-X relies on very advanced technology and claims -70% fuel burn reduction versus the 777-200 benchmark; the portion attributable to electric propulsion is closer to -20%, with much of the remainder a result of airframe and other technologies [53]. The concept will require on the order of 50 MW of power, which will certainly require superconducting electrical components and the associated cryogenic subsystem; thus, this concept introduces a large amount of technological risk and uncertainty and is envisioned for EIS in the 2040s. Jansen et al. review the individual technologies required to implement EP at such power levels, including soft magnetic materials fabrication, superconducting wires and electric machines, and insulators [29].

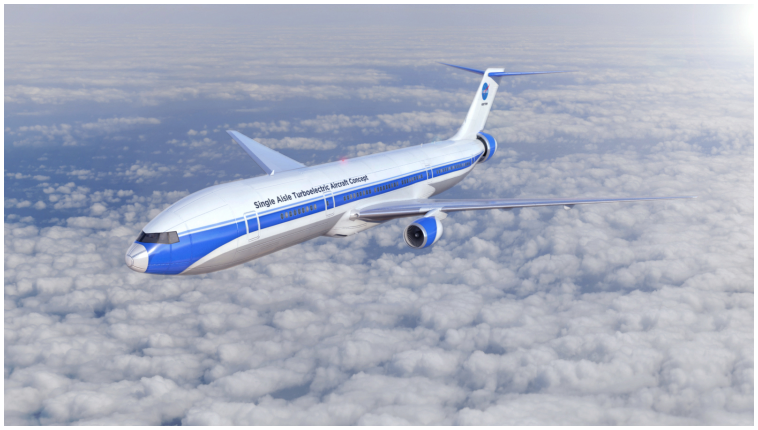


Figure 2.6: STARC-ABL turboelectric concept
(NASA image)

Because of the high technological barriers of the N3-X, NASA sought to develop a concept which would be feasible in the nearer term [29]. This resulted in a conceptual design study of a single-aisle turboelectric aircraft with an aft boundary layer propulsor (STARC-ABL) [51]. STARC-ABL adds a tailcone propulsor to a typical tube-and-wing single-aisle configuration with downsized, podded turbofans (Figure 2.6). Unlike many proposed turboelectric architectures, the STARC-ABL does not include a dedicated turbogenerator; the power for the electric propulsor is generated from the turbofans. The last NASA assessment showed -9.4% fuel burn for the economic mission [52].

From 2011–2013, the European aviation community released three new advanced EP concepts, but they are not as widely published as the NASA-funded concepts. An Airbus-funded study for an all-electric regional airliner, known as the VOLTAIR, was widely publicized in popular media but apparently resulted in just one conference paper (with minimal design definition) [61, 62]. The bulbous configuration had an aft-mounted boundary layer ingestion propulsor and was designed for natural laminar flow over the wing.

The Distributed Electrical Aerospace Propulsion (DEAP) collaboration between Airbus, Rolls-Royce, and Cranfield University developed preliminary studies for high-temperature superconducting (HTS) turboelectric propulsion [95, 96] and a concept aircraft known as the Airbus E-Thrust [63]. The E-Thrust is a series hybrid regional passenger aircraft with multiple, embedded electric fans in a striking split-tail configuration.

A team at Bauhaus Luftfahrt developed the all-electric Ce-Liner concept: a regional aircraft with very advanced technology. The Ce-Liner uses a C-wing shape for high aerodynamic efficiency and HTS electronics. The Ce-Liner study uses perhaps the most aggressive technology assumption for future e_b: 2000 Whr/kg [59, 60]. Isikveren et al. [60] present not only typical conceptual sizing data, but also detailed cabin layouts and ground handling considerations.

Figure 2.7 illustrates demonstrated and projected battery specific energy (e_b) for both flight tested prototypes and advanced concept studies. SOA batteries fall far below the requirements for

N+3 and N+4 advanced concepts. Boeing and Airbus use similar technology assumptions, while Bauhaus Luftfahrt has been using more aggressive assumptions.

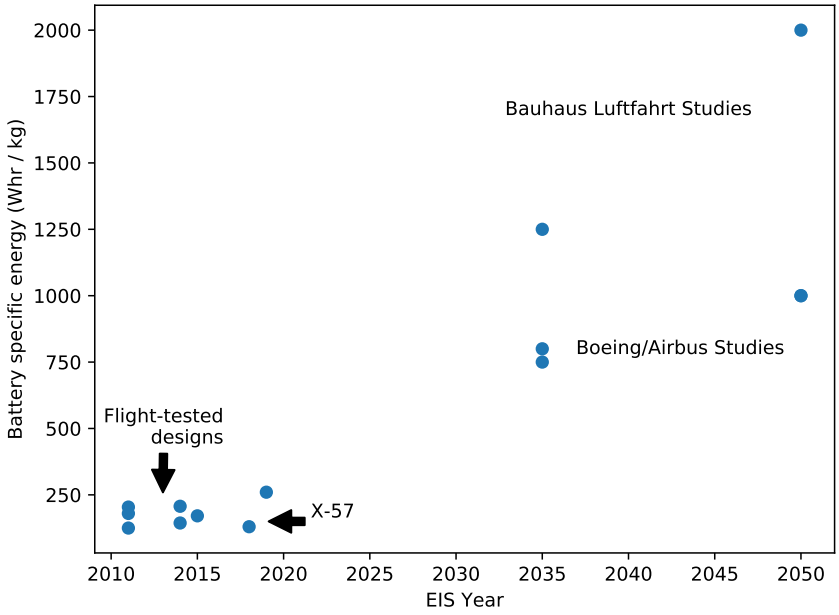


Figure 2.7: Historic and projected e_b
(data from Table 2.2 and 2.3)

Beginning the mid-2010s, many fixed-wing electric aircraft startup companies formed, generally focusing on general aviation and Part 23 aircraft in the 1 MW power range:

Zunum Aero was developing a 12-passenger hybrid electric regional aircraft designed to Part 23 rules for early 2020s EIS [69]. The company was funded by Boeing and JetBlue.

Wright Electric has been exploring single-aisle size airliners, beginning with a concept similar to the ESAero ECO-150 concept. Wright's design use electric fans embedded in the wings. Their main partnership is with the airline EasyJet.

Eviation is an Israel-based firm developing the "Alice" nine-passenger concept. As of the 2019 Paris Airshow, Alice used two wingtip-mounted pusher propellers and a tailcone propeller (some boundary-layer ingestion is claimed) [65].

Ampaire was an early-stage startup, initially promoting the "Tailwind" concept (featuring an aft boundary layer ingesting propulsor similar to the STARC-ABL) in an all-electric and series hybrid configuration [68]. Ampaire eventually focused most of its activity on a push-pull gas-electric version of the Cessna 337, which flew numerous manned test flights around Hawaii before the company was acquired.

XTI Aircraft claimed to have taken more than 60 orders for the Tri-Fan series hybrid vertical takeoff and landing (VTOL) aircraft. The design features three ducted electric fans, two of which rotate 90 degrees to transition from vertical to horizontal flight [67]. In 2019, the company started test flying a 65% all-electric proof of concept test vehicle.

ZeroAvia is a British company that flew a proof-of-concept hydrogen fuel cell Piper Meridian and is initially focusing on short-haul commuter aircraft.

Project Fresson is a consortium of British companies converting a Britten-Norman Islander as a hydrogen fuel cell demonstrator by 2022.

VoltAero is developing a special series-parallel hybrid propulsion system designed to provide twin-engine safety with single-engine economics. The company flew a Cessna 337 Skymaster with the piston hybrid-electric drive unit in the rear, and two 70 kW electric motors in a tractor configuration on the wings. For production, the company envisions 4–10 seat twin-boom aircraft with a single hybrid drive unit in a pusher configuration.

A multitude of other startups have focused on eVTOL, and many of these have employed fixed wings for the cruise phase of flight; they are too numerous to list here.

Several academic authors have conducted one-off studies on general aviation and commuter aircraft [97–100]. Others focused on small unmanned aerial system (UAS), such as the early work at the Air Force Institute of Technology by Harmon et al. [21, 101, 102] and one study of a UAS propulsion subsystem by Merical et al. [103]. In general, these studies do not rely on technology

as advanced as the large commercial transport studies.

2.4 Airplane-Level Effects of Electrification

Even assuming substantial improvements in e_b and p_m, electrification is likely to add substantial amounts of weight compared to conventional aircraft. Until weight parity with jet fuel and turbomachinery is reached, EP will need to "buy" its way onto each configuration based on the sum of airplane-level benefits and disadvantages. This section reviews some of the claimed benefits enabled by electrification and contrasts them with the known risks.

2.4.1 Direct Electrification Effects

Hybrid and all-electric aircraft can claim benefits directly by replacing fuel with electricity on shorter missions (see the Breguet range relations in Section 2.2). As e_b grows, a larger share of economic missions can be flown on electric energy. If electricity is cheaper than jet fuel on a per-unit-energy basis, then this may result in an operating cost savings. Depending on the source of electricity, carbon emissions reductions may also result. This requires a detailed understanding of generator fuels, transmission/grid losses, and lifecycle analysis of battery production and disposal. If renewable electricity generation is assumed, carbon emissions reduction is a clear benefit of EP.

All-electric aircraft also have a theoretical efficiency advantage by eliminating a thermodynamic cycle (see Section 2.2). If superconducting wires and power electronics are used, the overall electrical efficiency fraction may be very close to 1. This theoretical advantage is negated somewhat if battery energy is generated using conventional power plant turbines (which also experience thermodynamic losses). The efficiency advantage may still be meaningful even if grid power is generated using non-renewable means, since land-based steam turbines have higher thermal efficiency than aviation turbines.

Readers of electric propulsion studies must carefully examine energy accounting to ensure that

claimed emissions, efficiency, or total energy benefits are actually fair comparisons. For example, full-electric aircraft efficiency is often overstated by neglecting losses during grid power generation, while conventional aircraft understate carbon emissions by neglecting contributions due to petroleum processing.

Replacing turbine engines and fuel systems with electric motors and batteries may reduce the maintenance cost of all-electric aircraft. An assumption seems to exist that electric motors will cost less to maintain than turbofans of equal power/thrust, and this is probably justified based on the service experience of electric cars. For example, Tesla claims that its electric drive units last upwards of a million miles maintenance-free [104]. Goldberg et al. [105] examines the sensitivity of N3-X economics to maintenance cost and concludes that uncertainty in maintenance cost is an acceptable risk. Demonstrator programs should be structured in order to gather useful reliability and maintainability data alongside performance data.

2.4.2 Propulsion Effects

In turbine engines, the rotation speed of the fan or propeller is coupled to the speed of the turbine. For DC architectures, the fan and turbine speeds are decoupled, enabling both to be operated at their ideal point [77]. Turboelectric and series hybrid architectures capture all of the benefit of a geared turbofan (GTF) engine through reduced fan speed, with an additional efficiency benefit from shaft speed decoupling. Electric propulsion also may enable higher bypass ratio (BPR) by decoupling the number of fans from the number of engines. The combined effect of speed decoupling and bypass ratio is estimated at 4%–8% of η_p for a transport-category turboelectric aircraft [106]. This effect does not necessarily require distributed propulsion.

Boundary layer ingestion (BLI) increases propulsive efficiency by ingesting slower air from the fuselage or wing boundary layer. Suction from the fan inlet changes the pressure distribution upstream, and fan outflow energizes the wake, causing some controversy as to whether BLI should be accounted as a drag decrement or a η_p increment [11, 107]. BLI benefit is proportional to the

percentage of boundary layer flow captured at the fan inlet [51].

While boundary layer ingestion is possible with conventional engines (such as the MIT/Aurora D-8 concept [108]), the unique scaling properties of electric motors make BLI more feasible. Combustion engines generally pay a performance, weight, and efficiency penalty when scaled down in size, while electric motors scale mostly linearly [80]; therefore, turbofan or turboprop aircraft tend to have as few engines as feasible. Electric motors are also physically smaller, making wingtip and tailcone installations feasible (as in the X-57 Maxwell and STARC-ABL). Since the boundary layer extends to a small, finite thickness past the wing or fuselage, numerous small electric fans may cover more of the wing and body and ingest a higher fraction of boundary layer flow than a few larger-diameter turbine propulsors.

2.4.3 Aerodynamic Effects

Electric motors enable new propulsion installation possibilities, resulting in potential aerodynamic benefits. There are three proposed aerodynamic benefits of distributed electric propulsion: installation drag reduction, high lift augmentation, and swirl cancellation. The overall benefit is unproven and will depend on the mission and configuration but is probably between 0 and 8% drag [106].

Engine nacelles and pylons introduce propulsion installation drag, comprised of friction, interference, and wave drag. Along with nacelle and fan weight, propulsive aerodynamic drag serves as a practical limit on turbofan bypass ratio. Wick et al. [109] showed an 8% installed drag reduction by embedding distributed propulsors in the wing on a transonic military transport concept. Engine installations embedded in the fuselage surface may save wetted area equivalent to nearly half of a typical podded nacelle [108]. This benefit can be booked as a reduction in drag for fixed bypass ratio, or an increase in bypass ratio for fixed drag.

High lift augmentation through distributed propulsion reduces cruise drag by enabling higher wing loading (and therefore, lower wetted wing area and viscous drag). With higher overall C_L,

approach speed and takeoff field length constraints can be met with a smaller wing. The NASA LEAPtech wing concept and X-57 demonstrator are designed to test this idea; Stoll et al. [110] predict that cruise L/D of 20 can be achieved, compared to a baseline L/D of 11 through a 2.5–3 times larger wing loading [48]. The LEAPtech wing has been studied using CFD and is estimated to augment C_L by 1.7 to 2.4 (in absolute terms) [111–113]; however, a loss of power may result in a sudden stall if it occurs during low speed flight [77]. In the U.S., transport category airplanes may not claim a powered lift benefit to stall speed (and thus, approach speed) for certification (14 Code of Federal Regulations (CFR) 25.103 and 25.125); a small credit in takeoff speeds may be claimed. There does not seem to be an equivalent regulatory obstacle for Part 23 commuter and general aviation aircraft to take advantage of powered lift.

Finally, wingtip propulsors have been proposed in order to take advantage of cancelling some swirl in the wingtip vortices, but are not as well studied as BLI and lift augmentation. Miranda and Brennan [114] published experimentally-validated low-fidelity results. The X-57 design study booked this benefit as a drag reduction estimated between 5% and 10% using low-fidelity methods and used Miranda and Brennan's findings (though not their code) for further analysis [113]. Eviation's Alice concept also uses wingtip-mounted pusher propellers. The compactness of electric motors enables wingtip propulsion.

2.4.4 Sizing Effects

In addition to the wing loading increase possible with blown lift, EP may result in favorable control surface and engine sizing effects. The one-engine-inoperative (OEI) takeoff condition provides such an example. Multiengine aircraft must continue a takeoff at a minimum climb gradient after the loss of one engine. In conventional tube-and-wing aircraft, this results in oversized engines (to maintain minimum takeoff and climb thrust) and possibly oversized vertical tails (to trim the large OEI yawing moment); thus, the OEI condition imposes weight, drag, and cost penalties on conventional designs.

Electric propulsion provides design freedom to eliminate these penalties. The N3-X concept studies have illustrated how power distribution architectures may prevent asymmetric loss of thrust following loss of a power source [54]. Furthermore, if one or more fans are lost, compensating motors on the same side may be throttled up to higher power to cancel out the yawing moment. Modern commercial airplane vertical tails may (or may not) be sized by OEI yawing moments. Reduced thrust asymmetry through EP is likely to alleviate this constraint, possibly leading to smaller vertical tails, lower weight, and lower drag [27, 77]; however, in a tube-and-wing configuration, other constraints such as stability and crosswind landing are likely to limit the value of this benefit [27]. No studies have itemized this benefit separately from overall weight and L/D improvements.

Turbines in hybrid electric aircraft may be further downsized by using battery power during power-intensive phases of flight, and during normal operations and failure conditions. For example, a series hybrid may supplement takeoff and top-of-climb power with batteries, sizing the turbogenerators to a less critical condition [77]. Whenever turbomachinery is downsized, weight and cost reductions are sure to follow; if the size of podded engines is reduced, installation drag reductions may be achieved as well.

Turbine engines are subject to thrust and power lapse at altitude, rendering top-of-climb the engine sizing constraint in some cases [115]. Electric fans are not subject to power lapse with altitude when powered by batteries [48]. The STARC-ABL was able to reduce the thrust and size of its turbofans substantially using the improved thrust lapse characteristics of its electric tailcone propulsor, resulting in a virtuous cycle of lower weight and lower viscous drag [51].

2.4.5 Weight Effects

The most apparent disadvantage of electric propulsion today is weight. Every electric aircraft study must account for the weight of energy storage, transmission, and follow-on structural growth.

The direct TOGW penalty due to energy storage is inversely proportional to e_b and linear

with range. TOGW growth is especially pronounced at long range. Since batteries do not lose mass during flight, they carry a "hidden" weight and induced drag penalty compared to fuels. It is difficult to overcome the weight penalty on longer missions with all-electric or significantly hybridized designs with today's technology; the current frontier is 244 nmi range (for the Pipistrel Taurus G4 and eGenius) [37].

Electronic, propulsive, and thermal management components affect empty weight directly. This may be a benefit or penalty depending on the systems architecture, mission, configuration, and technology level. For example, the STARC-ABL claims a net propulsive weight *decrease* due to synergistic propulsive efficiency and turbofan sizing benefits [51]. The XTI Tri-Fan switched from a triple turboshaft design to a series hybrid electric architecture and booked a 37% gross weight reduction [116]. At the opposite extreme, the ECO-150R propulsion system is one-third of the overall empty weight [64]—about three times heavier than the 10.5% engine empty weight fraction of a 737-900ER or 777-300ER. Distributed propulsion may allow fan structural weight reduction by reducing tip speeds, and by relaxing bird strike requirements through redundancy [54]; on the other hand, BLI fans will need to be distortion-tolerant, which may add weight [108]. Distributed propulsion requires more aircraft structure to be reinforced against fan-blade loss (14 CFR 33.94), which adds weight.

Energy storage, electronic, and TMS weight also affect structural weight. For example, lower e_b leads to heavier batteries and higher rated maximum takeoff weight. This increases structural loads, structural gauge weight, and empty weight and reduces range in a vicious cycle. This cycle effectively prohibits long-range, manned, all-electric flights at today's technology levels.

2.4.6 System Safety Effects

Electric propulsion presents new and unknown challenges for airplane designers and safety engineers, but also exciting opportunities to eliminate known risks. All-electric aircraft substitute the hazard of flammable jet fuel with the new hazard of lithium batteries (or some future chemistry).

Aircraft manufacturers and regulatory agencies have long service experience with liquid fuels, and the resulting engineering controls (such as fuel tank inerting) are very well developed. Batteries are also a known aviation hazard but have much less service experience and fail in seemingly more complex modes.

The Federal Aviation Administration (FAA) has imposed special certification conditions on the use of rechargeable lithium-ion batteries in aircraft [117]. The primary hazard is *thermal runaway*, when rapid, self-sustaining increases in temperature and pressure occur in battery cells and may lead to an external fire. Toxic gases may also be released. Thermal runaway may result from overdischarging, overcharging, and internal short-circuits (see Section 2.5.2 for additional details on battery charge/discharge characteristics). Several lithium battery thermal events led to the grounding of the entire 787 fleet, and personal electronics lithium batteries carried in cargo are the suspected cause of multiple fatal air freighter accidents. A lithium battery module designed for the X-57 demonstrator required a major redesign following a failed thermal runaway during test, adding 45 kg empty weight [84]. Thermal runaway was identified as one of two critical hazards for the X-57 program because of combined loss of power and potential structural damage [49].

Crashworthiness is an area of potential benefit but high uncertainty. Liquid fuel, an obvious hazard following a crash, is replaced with battery modules. While only a few aircraft have faced survivable crashes with sizable lithium battery packs, electric automobiles will be a valuable source of data; automotive design practices for crash-tolerance of lithium batteries are reviewed by Arora et al. [118]. Much more data and testing will be required to refine the design of aerospace propulsion battery packs.

High-voltage electrical systems present another novel hazard in aircraft design. In-flight hazards include the release of energy through short circuits and arcing (see Section 2.5). At a minimum, this will result in degradation of propulsion system performance, and could lead to fire. High-voltage electrical systems also present a hazard to maintainers and ground handlers. This will likely be a similar hazard to aircraft hydraulic systems, which require special engineering

controls and lockout/tagout procedures to avoid injury.

All-electric aircraft eliminate turbines and their associated high-energy bladed disks. The FAA treats turbine disk burst zones as special hazard areas and requires that certain flight-critical systems not be exposed; this introduces spatial integration constraints. Electric fans turn at much slower speeds, reducing the potential hazard from blade loss. Aircraft with distributed propulsion may be more tolerant to bird strikes and engine loss than SOA twin turbofan configurations [54]. Distributed fans provide more redundancy and could potentially continue providing thrust in the rare circumstances that have caused dual engine failures; however, *embedded* propulsors (as in the ECO-150) may expose primary wing structure or systems to high-energy blade impacts.

N3-X study collaborators Rolls-Royce and the University of Strathclyde produced conceptual safety analyses and trade studies of high-power EP systems; Armstrong et al. is the most general [54]. More detailed studies included fault tree analysis of loss of thrust [92, 94] and superconducting fault protection considerations [93, 119]. Specific hazards related to NASA's current electric demonstrator efforts are analyzed by Clarke et al. [49] and Papathakis et al. [50].

In summary, electric propulsion potentially improves system safety during failure conditions and eliminates hazards associated with jet fuel, but introduces new hazards such as thermal runaway of battery packs. Methods to substantiate safety of EP systems, service experience from demonstrators, and design best practices will be crucial in reducing risk in this area.

2.4.7 Noise and Heat Signature Effects

Noise reduction is a widely claimed benefit of electric propulsion [43, 80, 120, 121]; unfortunately, the published literature does not include many quantitative noise predictions. One of the most relevant is a noise analysis of the SUGAR High (high-technology turbofan) and SUGAR Volt (parallel hybrid electric). The study found that SUGAR Volt was only about 1 EPNdB quieter than SUGAR High [19]; however, both are relatively conventional configurations with podded engines. Moore and Fredericks note that the noise savings from electric propulsion are primarily due to an

enlarged design space for propulsion integration, which enables more effective shielding than can be achieved with conventional engines [80]. SUGAR High and Volt do not use shielding in this way.

The original N3-X concept required an extensive redesign to meet noise goals, but it ultimately achieved a margin of 64 EPNdB compared to current Stage 4 community noise requirements [89]. Huff et al. [122] used empirical and low-fidelity methods and predicted that a 1 MW electric motor's contribution to external sound levels will be small compared to the noise of a low pressure ratio fan. Bryson et al. [123] describe the trade space between noise and range for a small, quiet UAS.

For military applications, reducing heat and noise emissions is desirable to avoid detection and improve survivability. Donateo et al. [124] describe a UAS with an electric-only mode to avoid generating a thermal signature.

2.5 Electrical System Architecture

In the past, electrical systems were a small portion of overall airplane weight, and engineers could afford to use rough empirical weight estimating relationships for conceptual design without making architectural decisions. Where an appreciable amount of electric thrust is produced, this is no longer the case. Electrical architecture choices will be strongly coupled with performance, weight, and flight safety. The key elements of an electric propulsion system include the following:

Energy storage: a battery (or alternative technology such as ultracapacitor or fuel cell) designed for high e_b, p_b, (dis)charge rate, and safety.

Generator: converts mechanical shaft power into AC electrical power

Rectifier: converts AC power to DC electrical power

Motor: converts AC or DC current to mechanical shaft power

Inverter: converts DC electrical power to AC power. Can be used as a variable-frequency drive for AC motors.

Bus: an electrical conductor that transfers electrical power from source to destination

Motor controller: closely related to inverters, motor controllers use DC power to generate time-varying currents in the armature coils of brushless DC and switched-reluctance machines. Motor controllers respond to position or speed feedback from the motor and do not necessarily generate sinusoidal currents.

Fault current limiter: prevents large currents during short-circuit events (essentially a circuit breaker). For DC superconducting applications, superconduction fault current limiters (SFCL) are used, which operate on different physical principles.

2.5.1 Electrical Machines and Power Conversion

Motors and generators (*electrical machines*) operate using the same general principles, and there are several types which may have advantages and disadvantages for flight applications. Electrical machines use the interaction between the magnetic fields of a rotating component (*rotor*) and stationary component (*stator*) to generate a mechanical torque. The main magnetic field in an electric machine may be constant or time-varying and can be generated using permanent magnets, soft magnetic materials (reluctance), passive field coils ("squirrel cage"), or active field coils (wound-rotor and doubly-fed machines). Typically, the main field is generated in the rotor [125].

When operated as a motor, windings in the stator generate a time-varying magnetic field and consume the large majority of the electrical power of the machine. Stator currents can be driven by a sinusoidal source (AC machines) or by arbitrary, actively-controlled waveforms (DC machines). AC machines typically operate in three phases, whereas DC machines may employ an arbitrary number of independently-controlled coils. In generator mode, the rotating main field induces a current in the stator coils.

Jansen et al. [29] summarize the state of the art of electric machines for flight applications, including efficiencies and specific power ratings. NASA expects machines in the 1 MW power class to achieve 13–16 kW/kg $p_{m/g}$ and 96–99% $\eta_{m/g}$. The NAE report envisions non-cryogenic generators with p_g up to 9 kW/kg by the N+3 time frame [11].

Lightweight and efficient power conversion (including inverters and rectifiers) is an evolving research area. A detailed discussion of the operating principles of power converters is beyond the scope of this review, but the major design choices being explored include semiconducting materials and cooling systems. Jansen et al. provides an up-to-date summary of NASA-funded power conversion research [29]. Armstrong et al. [90] present a table summarizing estimated specific power and technology readiness of many kinds of electrical components. For megawatt-scale converters, NASA envisions $p_{i/r}$ ratings between 19–26 kW/kg and efficiencies above 99% [29]. The NAE committee projects $p_{i/r}$ of about 9 kW/kg by the N+3 generation [11].

At high power levels, large amounts of waste heat are generated with even highly-efficient conventional electronics. Freeman et al. pointed out the memorable fact that a half-megawatt motor operating at a state-of-the-art 95% η_m produces as much waste heat as a barbecue grill [115]; therefore, *superconducting* motor/generators, power electronics, and conductors have been proposed as a way to raise efficiency and greatly reduce the thermal management problems introduced by megawatt-scale electric propulsion systems. Superconducting materials exhibit zero resistance at low operating temperatures, greatly reducing or eliminating Joule heating. For example, superconducting stator windings in motors and generators may improve efficiency substantially but incur penalties for other reasons (such as eddy current losses and fault currents) [88, 93]. Conceptual trade studies so far have favored superconducting architectures for very high-power applications (such as the 300-passenger NASA N3-X) and conventional conductors for megawatt-class requirements and below (such as the 150-passenger NASA STARC-ABL) [29, 53]. Thermal management is discussed in more detail in Section 2.6.

2.5.2 Batteries

An *electrochemical cell* "converts stored chemical energy into electrical energy via the energy difference between the reactions occurring at the two electrodes" [126]. A *battery* is a practical electrical energy storage device consisting of one or more cells connected in series and/or parallel in order to provide desired output voltage, capacity, and power.

An important parameter in the design and operation of battery-powered devices is the *C-rate* [127]:

$$\zeta(t) = \frac{I(t)}{C_{nom}},$$

where $I(t)$ is the charge or discharge current (in A) and C_{nom} is the nominal charge capacity of the battery (in Ah). At a 1C discharge rate, the battery will be fully discharged from nominal capacity in one hour. The rated maximum power of a battery is directly proportional to the maximum C-rate. A high C-rate is desirable from an operational perspective, as it enables rapid recharging (and less downtime). Two related metrics are the state of charge (SOC), which measures the percentage of charge capacity remaining, and the depth of discharge (DOD) which is simply $1 - SOC$ [127].

Each battery design has a characteristic voltage profile as a function of DOD. For lithium-ion batteries, voltage decreases slowly until a precipitous drop at about 90% DOD. Discharging past a threshold DOD (specific to the battery design) can cause shortened lifetime and/or thermal runaway; maximum DOD of 80% is typical in the literature [25, 128, 129].

At higher discharge rates, internal resistance in the battery becomes significant, and the output voltage will be reduced; therefore, high discharge rates (high power) result in lower total useful energy [127]. The curve of p_b versus e_b is called a *Ragone plot*, and it is specific to the cell design/chemistry and pack design. Xue et al. [126] show Ragone curves for Li-ion cells optimized for different C-rates (Figure 2.8). Vratny et al. [127] contains a related plot of battery "efficiency" ($e_{b,act}/e_{b,0C}$) as a function of C-rate.

Cell chemistry significantly contributes to battery characteristics. In general, batteries contain

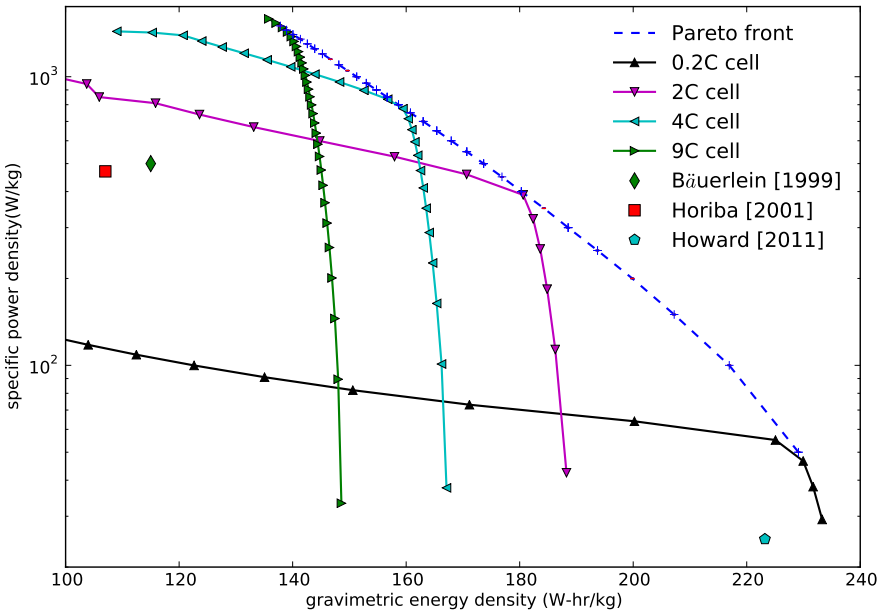


Figure 2.8: Ragone curves for Li-ion cells optimized for 0.2C–9C rates, with Pareto front (figure from Xue et al. [126])

a positive electrode (cathode), negative electrode (anode), electrolyte, and various other inactive materials for binding, insulation, and other necessities. Li^+ ions are currently favored as a charge carrier because of high potential and light weight [126]. The theoretical upper limit on cell-level specific energy is determined by the electrochemical reaction; however, practical considerations ensure that the actual e_b is lower than the theoretical maximum.

Xue et al. discuss the tradeoff between high specific energy and high specific power [126]. At the cell level, densely-packed active materials result in high e_b, but the resulting low porosity and conductivity limit the C-rate (and therefore p_b). For a given cell chemistry, a Pareto front exists between specific energy and specific power when considering detailed cell design parameters (Figure 2.8). Adding inactive materials is also required in order to prevent degradation of e_b over repeated charge/discharge cycles. Each of these factors causes actual cell specific energy to fall below the theoretical value. At the battery pack level, practical design considerations introduce additional parasitic weight (such as spacing/cooling to contain cells in thermal runaway and battery control electronics [84, 130]).

Battery modeling in conceptual aircraft studies often assumes a fixed pack-level e_b and p_b with constant battery efficiency factor [18]. This approach may be appropriate for long-term advanced technology studies where detailed discharge characteristics of the battery technology are unknown. A higher level of fidelity considers discharge rate effects by constructing an "equivalent circuit" model of the battery. The battery model consists of an internal resistance and possibly capacitance determined from empirical data [19, 127, 131, 132]. Equivalent circuit models introduce a power-dependent battery efficiency factor and create coupling between mission analysis, control strategy, and vehicle weights. Physics-based models considering internal cell design variables have been used to explore coupling between vehicle design and optimal cell design [133]; however, physics-based cell modeling introduces significant computational cost and has not yet been coupled to an aircraft design problem. Avanzini et al. contains a detailed procedure for empirical modeling of a particular battery pack [134].

2.5.3 Electrical System Trades

Key trade studies for an electric propulsion system include the choice of AC or DC in the distribution buses, the nominal system operating voltage, and whether to utilize constant-voltage or variable-voltage DC power for battery-fed hybrid- and all-electric systems. Vratny et al. [135] studied constant system voltage (CSV) and variable system voltage (VSV) in a hybrid system and concluded that VSV was more efficient overall. High system voltages may be more efficient but incur a weight or safety penalty due to the risk of electrical arcing at high altitude [90, 135]. Paschen's Law dictates that air gap insulation is safe below 327V, no matter the air density or pressure; therefore, current commercial airplanes restrict electrical bus voltages to 270V or below for intrinsic safety. Electric propulsion concepts will require higher operating voltages in the low kV range, necessitating investment in highly durable and lightweight insulative materials [29, 90, 136].

The literature currently favors DC power distribution, particularly for any application involving

the use of batteries but also for high-power turboelectric [19, 54]. DC power distribution eliminates the need to synchronize the phases of multiple electrical generators and motors and simplifies "throttling" for distributed propulsion applications, but it incurs weight and efficiency penalties due to AC/DC conversion on both ends [137]. Each unit of rated motor power requires an equal unit of inverter *and* rectifier power, plus redundancies for system safety. Using the p_e estimates given above, this means that the weight of the power electronics will be on the order of or even heavier than the motors themselves. Regardless, since batteries operate on DC, there is no need to rectify the power; therefore, where electric energy storage is used, DC power distribution will be more favored [138]. An alternative AC power distribution scheme for a turboelectric configuration was devised by Sadey et al. [139] and utilizes doubly-fed induction machines similar to wind turbines. Figures 2.2c and 2.2d illustrate the more common DC distribution architecture.

For hybrid electric applications involving turbofans, shaft power may be taken off from the low-pressure and/or high-pressure spool. While the high-pressure spool rotates over a narrower range of speeds, operability characteristics of the engine limit the amount of shaft power that can be extracted [14]. United Technologies and GE independently announced successful demonstrations of near-MW scale power offtakes from low-pressure spools [140, 141].

2.6 Thermal Management

While electric motors are more efficient at generating shaft power than turbine engines, turbines exhaust nearly all the waste heat into the ambient air. Because electrical components are located within the aircraft interior, ambient cooling is not sufficient to keep conventional (non-superconducting) electronics at their operating temperature. Superconducting electrical power systems eliminate resistive heating but introduce a different problem: keeping high-temperature superconducting (HTS) materials sufficiently cold compared to the ambient temperature. Both technologies will require TMS. A deeper review of thermal management design approaches and

models is deferred until section 5.1

2.7 Concluding Remarks

The design of economically-viable fixed-wing electric aircraft demands high technology and highly integrated design. It is widely known that specific energy of batteries (e_b) and specific power of electronics (p_e) strongly impact aircraft capabilities. Battery and electronics unit costs also influence aircraft acquisition cost. Driven by investment from the automotive industry, the performance and cost of these components should improve over time. The aerospace industry should monitor developments and independently invest in adapting the technology to meet aerospace-specific design requirements.

Regardless, the technological barriers to aircraft EP remain challenging. The NAE consensus report [11] finds that "turboelectric propulsion systems are likely the only approach for developing electric propulsion systems for a single-aisle passenger aircraft" feasible by the N+3 (2035) time frame, due primarily to limited advancement in e_b. Studies at smaller power scales and ranges are more optimistic.

As Moore and Fredericks [80] argue, the practical and economic viability of electric aircraft is also determined by the effective use of the additional degrees of design freedom opened through electrification. This is corroborated by findings that more-electric propulsion is not effective as a drop-in replacement for existing combustion engines [142]. The design space for electric aircraft is still poorly understood due to the very limited diversity of designs with service experience, the small number of well-developed trade studies, and the many configuration degrees of freedom.

Opening the design space, particularly for propulsion architecture, requires constant evaluation of safety and regulatory compliance. Electric propulsion potentially eliminates longstanding risks such as combustible fuels, but also introduces new and potentially unknown risks. Intelligent systems architecting can take advantage of EP capabilities to meet safety requirements in new

ways; however, innovative ideas may require regulatory acceptance before they become feasible for use in particular applications (such as powered lift for commercial aircraft certified by the FAA). Modeling, simulation, and optimization are promising ways to rigorously explore the design space.

CHAPTER 3

A Brief Review of Aircraft MDO

3.1 General Background

Design optimization, which combines simulation models with mathematical optimization algorithms, is an increasingly important tool in aircraft design. *Multidisciplinary* design optimization (MDO) is design optimization involving more than one engineering function or discipline. MDO models are advantageous because they capture physical coupling, such as the interplay between aerodynamic loads and structural deflection. As detailed by Martins and Lambe [143], pioneers of aircraft MDO were demonstrating aerostructural optimization problems with the limited computing resources of the late 1970s. Haftka [144] computed Pareto curves between structural weight and induced drag using linear potential flow and finite elements in 1977. Schmit [145] optimized structural designs as early as the 1960s, and by the 1980s had also demonstrated MDO of turbomachinery blades subject to aerodynamic and structural constraints. As computational power increased and methods improved, researchers were able to increase the fidelity of the model physics. Martins incorporated Euler aerodynamics into a supersonic business jet aerostructural problem [146], and also demonstrated a significant range improvement by optimizing using coupled MDO instead of a sequential approach. The level of fidelity continues to increase, from multipoint aerodynamic analysis [147] to Reynolds-averaged Navier–Stokes (RANS) physics [148] and even other aeroelastic effects such as flutter [149]. multidisciplinary design analysis and optimiza-

tion (MDAO), typically at lower fidelity, has been demonstrated and used for industrial design, e.g. [150, 151].

3.2 Overcoming Barriers to Large-Scale MDO

One of the most significant constraints on the scale of MDO problems is the computational cost. Factors which contribute to the overall computational cost include the cost of the simulation(s), the number of disciplines, and the number of design variables and constraints. Using gradient-based optimization algorithms can offset much of the computational cost of additional design variables. Gradient-based optimizers exploit the derivatives of the objective function and constraints with respect to the design variables. These methods tend to outperform gradient-free optimizers as long as the requirements for gradient-based methods are satisfied. Figure 3.1 illustrates that gradient-free optimizers scale poorly for larger-scale optimization problems with more design variables, whereas gradient-based optimizers are nearly invariant in the number of design variables.

One disadvantage of gradient-based methods is that they require computing derivatives. This can require additional development of the analysis software. It also incurs an additional computational cost, as computing the gradients of a nonlinear, implicit function can be expensive. Careful attention must be paid to the method of derivative computation to ensure minimum computational cost and high accuracy. For optimization problems with many design variables and only a few relevant outputs (which includes multidisciplinary shape optimization problems), computing derivatives in the reverse mode (also known as computing the adjoint) is significantly cheaper than using the forward mode or finite differences. Martins and Hwang [153] provide a comprehensive review of derivative methods relevant to MDO. Choosing an efficient mathematical form of the MDO problem can also mitigate computational cost — for example, it may sometimes be beneficial to allow an optimizer to satisfy a constraint equation instead of using a separate solver. A comprehensive review of MDO architectures and the associated benefits and disadvantages of each

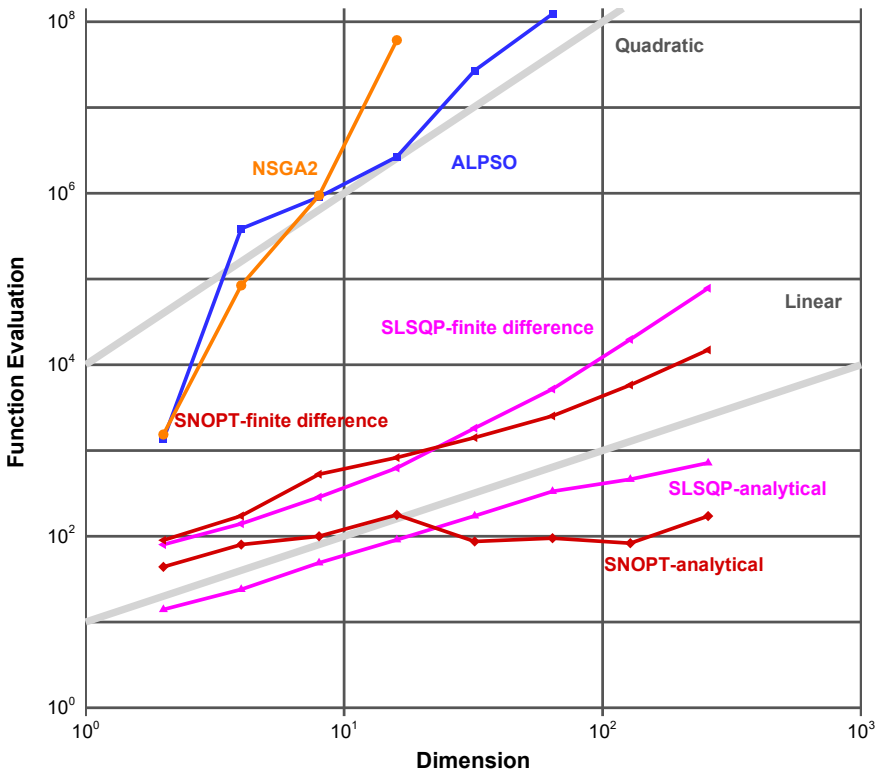


Figure 3.1: Gradient-based optimization scales well with the number of design variables (from [152])

can be found in Martins and Lambe [143].

Another significant barrier to constructing large-scale MDO models is ease of use and development time. MDO practitioners historically have been required to master not only the relevant design disciplines involved, but also scientific computation and a significant amount of applied mathematics. Some MDO projects developed *frameworks* which define interfaces between complex analysis codes, solvers, and optimizers. These frameworks tended to be specific to the task at hand, such as the MDO of Aircraft Configurations with High Fidelity (MACH) framework which was developed for aerostructural optimization of transport aircraft [154].

The recent development of the OpenMDAO framework [155] has revolutionized the field by providing a high-performance, general-purpose MDO framework. OpenMDAO allows users to define complex, hierarchical MDO models as a composition of simpler components. It provides

powerful linear and nonlinear solver methods to converge multidisciplinary problems with high performance. One of OpenMDAO's most powerful features is that it computes derivatives across large, coupled models in an efficient and accurate way [156]. This eliminates the onerous requirement for MDO developers to define custom coupled adjoints for every new problem.

3.3 Modeling, Simulation, and Optimization of Electric Aircraft

3.3.1 Electric propulsion-specific modeling challenges

While individual simulation capabilities for electric aircraft have been addressed in Chapter 2, the intent of this section is to discuss MDO in the context of electric aircraft. Models for MDO (especially for more conceptual-level MDO problems) often rely on historic data and regressions to estimate unknown parameters, but models adapted from conventional aircraft may be inapplicable to innovative configurations. Wind tunnel and flight test data is limited or nonexistent, so new empirical models for the most expensive physics (e.g., high-lift prediction for distributed propulsors) cannot easily be constructed. We also lack any significant operating cost data for electric aircraft. To understand the economic case for aircraft EP, we must also quantify the effect of electrification on acquisition, operational reliability, and maintenance; all areas that are traditionally "resistant" to physics-based modeling.

Making matters more complex for the modeler, the design freedom, close coupling, and tight packaging requirements of the electric propulsion design problem introduce competing objectives for models. Compared to conventional aircraft design, the electric aircraft design problem is likely to be even more strongly coupled between disciplines than conventional aircraft. For example, the high-fidelity aeropropulsive optimization of Gray et al. [157] demonstrated the coupling between aerodynamics and propulsion for the STARC-ABL. The ECO-150R study [64] illustrated cou-

pling between electrical, propulsion, thermal management, and aerodynamics disciplines. Novel thermal constraints are also time- and path-dependent, requiring trajectory analysis and potentially trajectory optimization tightly coupled with the rest of the aircraft design optimization problem [131, 158].

As the number of closely-coupled disciplines grows, the computational complexity (in terms of numerical cost) and the software complexity (in terms of lines of code and effort) also grow. Low-cost, moderate-fidelity analysis is necessary in order to explore the broad tradespace, while high-fidelity analysis and optimization is necessary in order to explore close coupling and spatial integration. This will require flexibility and high performance in any modeling framework, along with judicious modeling choices about where to apply higher-fidelity methods.

3.3.2 EP modeling and simulation projects

One published optimization environment for design of electric fixed-wing aircraft is ESAero's software, which has been known by the names HAPSS, DOETech, TOGW Framework, and PANTHER over the last decade. The general capabilities and architecture of these tools have been published [64], but the software itself is proprietary. The PANTHER tool has at least been used for analysis and to generate tradespace contour plots, but it is unclear whether an airplane-level MDO capability has been implemented. Multiobjective optimization has been used to design a Meredith effect radiator [64].

Another unified framework is the Georgia Tech GT-HEAT framework [159]. This is a propulsion-focused model: all components, even electrical models, are implemented in the Numerical Propulsion Simulation System (NPSS) software program. This modeling framework has good fidelity in propulsion and electronics but lacks important capabilities in aerodynamics, structure, and cost modeling. Optimization was conducted using a design of experiments (DOE) and surrogate modeling methodology [160].

NASA has been developing modeling, simulation, and optimization capabilities for the X-57,

N-3X, STARC-ABL, and PEGASUS studies, mostly at conceptual levels of fidelity. Capristan and Welstead introduced the LEAPS software package, designed as a replacement for NASA's FLOPS mission analysis software but supporting distinctive features of aircraft EP (including mixed electric and fuel-burning propulsion and independent throttling). LEAPS uses energy-based methods to provide low-fidelity, low-cost estimates of somewhat optimized mission trajectories [161, 162].

Basic MDO capabilities were implemented to guide NASA studies [48, 131]. More recently, Hwang and Ning [163] developed a medium-fidelity optimization tool for vehicles similar to the X-57, incorporating blade element momentum theory propulsion modeling, vortex lattice aerodynamic analysis, finite element structural sizing, automatic derivatives using the adjoint method (to handle hundreds of design variables), and full mission analysis; however, thermal modeling was not included.

Falck et al. [131] demonstrated the need for modeling thermal constraints when trajectory optimization is considered. The tool was implemented in OpenMDAO to facilitate gradient-based optimization and computation of derivatives [156]. Despite significant progress on particular studies, there is no single NASA optimization environment for electric propulsion trade studies, and high-fidelity analysis has not yet been incorporated.

Gray et al. [157] conducted high-fidelity aeropropulsive shape optimization of the STARC-ABL's aft tailcone propulsor. A RANS CFD solution was fully coupled to a 1-D thermodynamic cycle model using the OpenMDAO framework. This combination enables direct physical modeling of boundary-layer ingestion. The flow solver and thermodynamic cycle model both provide efficient derivatives, enabling the use of a coupled adjoint method to compute total derivatives and perform optimization with respect to hundreds of geometric design variables. This is the first published shape optimization of a hybrid electric aircraft using high-fidelity CFD, but the scope of the MDAO problem was limited to the region of the aft tailcone propulsor; a full-airplane optimization has not yet been conducted.

The fully coupled aeropropulsive optimization found that the fuel burn reduction contribution

from BLI was attributable to both aerodynamic effects and propulsive effects (8 to 12 counts from aerodynamic drag; 16 effective counts from propulsion). The analysis also found that aft body aerodynamic drag was very sensitive to propulsive design variables (such as fan pressure ratio). Optimizing the configuration as a multidisciplinary coupled system led to a significantly better optimum design point than an uncoupled analysis. While such multidisciplinary coupling will manifest in ways specific to each configuration, this case study illustrates the benefit of using high-fidelity, coupled modeling for analysis and optimization in order to maximize the benefit afforded by electric propulsion.

Bauhaus Luftfahrt developed simple low-fidelity methods and conducted broad tradespace exploration studies; unlike all other studies, cash operating cost was considered as an optimization variable [27].

The electric VTOL literature on modeling and simulation may also apply to fixed wing applications in some cases. The Boeing team of Duffy et al. developed an MDO framework for sizing and tradespace exploration of small eVTOL aircraft. The paper includes a simple methodology for operating cost modeling for electric aircraft, including some low-fidelity estimation of cost savings due to increased reliability [164]. A similar approach was used by Brown and Harris [165]. The Airbus Vahana team posted an open-source conceptual design tool with MDO capability on the web, but has not published on the work in a scholarly forum [151].

A summary of modeling and simulation capabilities, by discipline and project, is shown in Table 3.1. It is worth remembering that advanced modeling and simulation capability probably exists in trade secret form within industry, so the published literature may present an incomplete view of the state of the art in electric aircraft modeling. It is clear from the summary table that certain disciplines have received a great deal of attention (electrical, propulsion) while, for example, nearly no cost modeling has been incorporated (though one-off studies have tackled the issue [164]). While one-off safety analyses have been conducted, safety has not been incorporated into the analysis process in an automated fashion.

Table 3.1: Electric aircraft modeling and simulation

	GT-HEAT [132, 159, 166]	**NASA X-57**	**NASA N-3X**	**ESAero [64]**	**Bauhaus Luftfahrt**
Aerodynamics	FLOPS/drag polar; BLI benefit based on flat-plate momentum thickness	Design using vortex lattice/boundary layer codes; some CFD for analysis [48, 82, 163]	CFD results from similar configuration, with increment for BLI [53]	Drag polar	L/D correction methods from Torenbeek [18]
Structures	NA	6 DOF beam finite element method (FEM) [163]	NA	NA (for MDAO); detailed analysis of split-wing published in NASA report	NA
Weights	FLOPS tops-down methods	Parametric wing weight (from Raymer) [48], sized beam model [163]	WATE for propulsion flowpaths; tops-down kg/kW estimates for electrics/TMS [88, 89]	WATE for fan weight [78]; low-fidelity radiator model; tops-down empirical for all others	Semi-empirical structural methods; tops-down kg/kW methods for electrics [22]
GNC	Engine, motor, TMS control variables for on- and off-design analysis	Full-mission optimal control [131]	NA; some discussion of off-design conditions in [54]	NA	NA
Electrical	Moderate fidelity motor/inverter loss modeling; equivalent-circuit battery	Transient battery model based on Thevenin equiv. circuits (cell-level). Assumed efficiencies for wire/motors [131]	Conceptual: efficiency stackup method with estimates for future tech. Transient: RLC circuit model in SimPowerSystems [88, 93]	Efficiency stackup; battery model unclear	Low-fidelity efficiency stackup with empirical battery discharge curve [22]
Turbo/Propulsion	NPSS	Propeller map from manuf.; prop efficiency from theory [131], blade element momentum theory [163]	NPSS [56]	2D fan analysis using velocity triangles [78]; efficiency maps for turbomachinery	Single prop efficiency parameter [22]
Thermal	TMS sizing considering various heat sources and types of heat sinks	Analytical model for optimization; thermal FEM of motor [131, 167]	Coolant system load based on efficiency stackup (assume 100% to heat) [88]	Cooling based on flight cond. [76]; TMS model discussed in [115]	NA
Cash operating cost	NA except for fuel/energy	NA except for fuel/energy	NA	NA	Considers relative cost of fuel/elec; cash operating cost [27]
Ownership cost	NA	NA	NA	NA	NA
Noise	NA	NA	ANOPP noise simulation prompted redesign [89]	NA	NA
Safety	NA	Comprehensive FMEA [49, 50]	FMEA and FTA for loss of thrust; more work needed for other hazards [92, 94]	Qualitative	NA

3.3.3 Limitations of Prior Work

There is significant duplication of effort in the research community, particularly within the area of electrical system modeling and mission analysis. Several codes with similar levels of fidelity for integrating energy used over a mission have been announced [25, 127, 131, 132, 161–163, 168–170], but none have been open-sourced or made publicly available (with the exception of Stanford's SUAVE, which includes some limited support for EP modeling). NASA's LEAPS is being developed with the intention of open-sourcing the code [162], and as of 2020 there is a preliminary working version, but the release plan is not clear as of this writing [171].

As of 2018, no publicly-available EP mission analysis and sizing code supported thermal analysis. Multiple industry and government studies have already demonstrated the need to include thermal constraints in analysis and optimization at the conceptual level [115, 131, 132, 142]. Although the LEAPS energy integration method supports electrical and fuel energy storage options, thermal analysis is not included. The open-source SUAVE conceptual design tool also does not incorporate thermal management analyses.

Another need exists for an EP design framework with efficient gradients. SUAVE does not support analytic or automatic gradients, and it does not appear that LEAPS will either. Other NASA electric aircraft studies demonstrate the benefits of efficient gradients in mission analysis codes [131, 163].

Falck et al. [131], and Hwang and Ning [163] developed electric aircraft mission analysis codes with moderate fidelity and efficient gradients. The two codes are similar and rely on optimal control theory and collocation methods to calculate trajectories, energy usage, and thermal states. Using OpenMDAO, the codes provide efficient gradients for use in large scale optimization [163]. However, optimal control-based methods sometimes introduce robustness problems. For example, the problem may not converge if initial guesses of the states and trajectories are not close enough. There may also be cases where non-optimal trajectories form constraints on the design problem (e.g., whenever a human pilot is in the loop). Neither model supports parametric cost estimates,

and neither model is currently publicly available.

3.4 Concluding Remarks

Ideally, a fully-coupled MDAO tool employing high-fidelity physics-based models for all relevant disciplines in electric aircraft design should be developed. However, no industry, government, or academic team has publicly acknowledged the development of such a tool at the time of this publication. The *full* MDAO problem, including economic and safety analysis, remains a formidable open challenge. This dissertation details several meaningful steps toward coupling more disciplines (especially thermal) and making electric aircraft MDO models more user-friendly to develop and use.

CHAPTER 4

Conceptual MDO of a Series-Hybrid Aircraft with Efficient Gradients

We recall from Chapter 3 that a need exists for an electric aircraft mission performance and sizing tool with:

- Thermal analysis
- Component-based parametric cost
- Public availability
- Efficient gradients for use with high-fidelity MDO

With these goals in mind, I developed a conceptual design toolkit, under the working title "OpenConcept". OpenConcept is a Python-based library built on top of the NASA OpenMDAO framework [156, 172]. At the highest level, it consists of three parts: a library of propulsion modeling components; a set of reusable, analytically-differentiated mission analysis codes; and a set of example aircraft models capable of analysis only, simple resizing, or full MDO.

This chapter describes the basic methods and models used in the OpenConcept framework.

4.1 Propulsion Models

I initially developed a set of simple, conceptual level electrical and propulsion components that can easily be adjusted to reflect future technology levels, as follows:

- `SOCBattery`: electrical power source with constant specific power and specific energy
- `SimpleMotor`: constant efficiency
- `SimpleGenerator`: constant efficiency
- `SimpleTurboshaft`: constant power specific fuel consumption (PSFC)
- `SimplePropeller`: based on empirical efficiency map
- `Splitter`: combines or divides power sources or loads

These can be connected together to form all-electric, conventional, series hybrid, parallel hybrid, or turboelectric architectures.

4.1.1 Battery

The `SOCBattery` component provides an electrical power source with constant specific power and specific energy. The main sizing parameter of the battery is the total pack weight m_b. Total energy of the pack in the fully-charged state is then $m_b e_b$ where e_b is the specific energy.

State of charge is computed by integrating the ordinary differential equation (ODE):

$$\frac{d\text{SOC}}{dt} = \frac{P_b}{m_b e_b} \tag{4.1}$$

where P_b is the electrical load (power) drawn on the battery. The battery produces heat

$$q_{\text{comp}} = (1 - \eta_b) P_b \tag{4.2}$$

where η_b is the battery electrical efficiency. Since battery (in)efficiency is already typically booked in e_b, it is only used for computing heating. The battery sizing ratio is computed as:

$$\phi_b = \frac{P_b}{m_b p_b} \tag{4.3}$$

where p_b is the battery specific power. The optimizer constrains $\phi_b \leq 1$ at every point in the mission such that the battery does not exceed its maximum rated power. Finally, a linear estimate of battery cost is computed as:

$$\text{cost} = \text{base cost} + \text{incremental cost} \times P_b \tag{4.4}$$

4.1.2 Motor

The `SimpleMotor` component is agnostic of the motor design and simply assumes a constant electrical efficiency across the operating envelope. The motor is sized by the rated *electrical* power $P_{\text{m,rated}}$ which represents a max continuous power setting. The electrical load P_m of the motor can be computed as:

$$P_m = \Theta_m P_{\text{m,rated}} \tag{4.5}$$

where Θ_m is the motor throttle setting. $\Theta_m = 1.0$ represents maximum continuous power; for takeoff, $\Theta_m > 1.0$ may be permissible for a time. Likewise, mechanical power $P_{\text{m,shaft}}$ and heat q_{comp} are computed as follows:

$$P_{\text{m,shaft}} = \Theta_m \eta_m P_{\text{m,rated}} \tag{4.6}$$

$$q_{\text{comp}} = \Theta_m (1 - \eta_m) P_{\text{m,rated}} \tag{4.7}$$

where η_m is the motor efficiency. Motor weight m_m and cost are computed linear functions of $P_{\text{m,rated}}$.

4.1.3 Generator

The `SimpleGenerator` is the mirror image of `SimpleMotor` in that it takes mechanical shaft power as input and produces electrical power. The primary sizing parameter is the generator power rating ($P_{\text{g,rated}}$). Electrical power P_g is computed as

$$P_g = \eta_g P_{\text{shaft,in}} \tag{4.8}$$

where η_g is the generator efficiency and $P_{\text{shaft,in}}$ is the input shaft power. Heat, weight, and sizing ratio are computed in a similar fashion as `SimpleMotor`.

4.1.4 Turboshaft

The `SimpleTurboshaft` converts fuel to shaft power using a constant specific fuel consumption model. The mechanical power rating $P_{\text{ts,rated}}$ is the primary sizing parameter for the engine, and defines shaft power as follows:

$$P_{\text{ts}} = \Theta_{\text{ts}} P_{\text{ts,rated}} \tag{4.9}$$

where Θ_{ts} is the turboshaft engine throttle setting. The fuel flow $\dot{m}_f$ is proportional to the shaft power:

$$\dot{m}_f = P_{\text{ts}}\,\text{PSFC} \tag{4.10}$$

where PSFC is the power specific fuel consumption of the engine. Typical units of PSFC would be lb / hp / hr. Weight and cost are again linear functions of the power rating.

4.1.5 Propeller

The `SimplePropeller` uses an empirical efficiency map for a constant-speed propeller to convert shaft power into thrust. Thrust T of a propeller in flight is computed as follows:

$$T = \frac{P\eta_p}{V} \tag{4.11}$$

where P is the shaft power driving the propeller, V is the airspeed, and η_p is the propulsive efficiency. For a given propeller design in flight, efficiency data can be made to collapse well onto two nondimensional parameters. The first dimension, advance ratio J, is defined as:

$$J = \frac{V}{nd_{\text{prop}}} \tag{4.12}$$

where V is the airspeed, n is the number of revolutions per second, and d_{prop} is the propeller diameter.

The second parameter, power coefficient C_p is defined as:

$$C_p = \frac{P}{\rho n^3 d_{\text{prop}}^5} \tag{4.13}$$

where ρ is the air density and P is the shaft power into the propeller. The propeller map defines the propeller efficiency $\eta_p = f(C_p, J)$ as a surrogate model. I used a propeller map for a turboprop airplane from [173]. Weight of the propeller is estimated using a historical regression [174], as follows:

$$m_{\text{prop}} = 0.108(d_{\text{prop}} P_{\text{rated}} n_{\text{blades}}^{0.5})^{0.782} \tag{4.14}$$

where m_{prop} is weight of the propeller in pounds, d_{prop} is propeller diameter in feet, P_{rated} is the rated power in horsepower, and n_{blades} is the number of propeller blades.

At low speeds (such as during takeoff), the velocity V goes to near-zero and Equation 4.11

has a singularity. To work around this issue, a separate static propeller map adapted from [175] is used at lower speeds ($J < 0.1$) and thrust is linearly interpolated between the two maps at low-intermediate speeds ($0.1 < J < 0.2$).

4.1.6 Splitter

For hybrid architectures, electrical or mechanical power from a source generally needs to be split across multiple destinations. This is accomplished using the `Splitter` component. The user can elect to use either a fixed-amount strategy or a ratio strategy to allocate input power P_{in} to destinations P_A and P_B. If the fixed-amount strategy is used, P_A is specified directly as a control parameter and $P_B = P_{\text{in}} - P_A$. If the ratio strategy is used instead, the control parameter becomes $0 \leq \Theta \leq 1$ and $P_A = \Theta P_{\text{in}}$, $P_B = (1 - \Theta) P_{\text{in}}$.

4.1.7 Power Matching

The OpenConcept modeling approach is that, in general, machinery "pushes" shaft power and electrical consumers "pull" electrical load proportional to rated power and and some control setting. A turboshaft engine pushes shaft power based on its throttle setting. An electric motor both pushes shaft power and pulls electrical load on an upstream source, which might include a generator or a battery.

Where mechanical power (typically, a turboshaft) drives a generator, there is an implicit gap where electrical power produced by the generator in must equal the electrical load on the generator and the required shaft power is unknown. OpenConcept uses a Newton solver to find the throttle setting for the engine by driving the following residual equation to zero at every flight condition:

$$\mathcal{R}_g = P_g - P_{\text{req}} \tag{4.15}$$

where P_g is the instantaneous generator power produced and P_{req} is the summed electrical load on the generator. Batteries do not require this special treatment, as state of charge will deplete at the correct rate given the summed electrical load on the battery.

4.2 Example Propulsion Model

Figure 4.1 shows a twin-motor series hybrid propulsion system built in OpenConcept and used for the case study in Section 4.5. Two motor components are connected to individual propellers. Each motor drives the propeller with a given shaft power depending on the throttle setting. The throttle settings may be different depending on the flight condition; for example, to model an engine-out takeoff condition, one motor may be set to idle while the other uses full emergency power. The combined electrical load of both motors is then split between the battery and the generator using a hybridization factor. The OpenMDAO Newton solver then computes the turboshaft throttle setting required to balance the system across the gap.

4.3 Analysis Routines

OpenConcept analysis routines can be used to model portions of the flight which impose design constraints. Takeoff analysis computes the takeoff field length assuming "critical" propulsion failure and often sizes the wing area, high-lift systems, and installed thrust. Mission analysis computes the fuel and/or energy used during the climb, cruise, descent, loiter, and any other relevant phases of flight. Together, these analysis routines define sufficient objective functions and constraints to perform conceptual-level aircraft MDO.

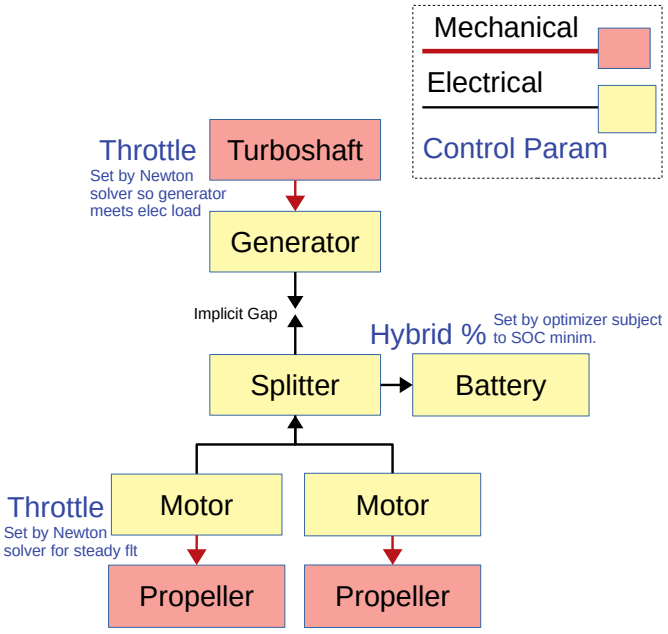


Figure 4.1: Example of a twin-motor series hybrid electric propulsion model in OpenConcept.

4.3.1 Takeoff Analysis

OpenConcept's `takeoff` module calculates balanced field length (BFL) and propulsion system states during the takeoff run, using methods and assumptions presented in [175]. For this flight phase, control inputs are specified by the user (e.g. 100% throttle) and accelerations are determined using a force balance equation. In order to compute balanced field length, `takeoff` divides the takeoff into five segments:

1. Takeoff roll at full power from rest to V_1 (the takeoff decision speed)
2. Takeoff roll at OEI power from V_1 to V_R (takeoff rotation speed)
3. Rejected takeoff with zero power and max braking from V_1 to rest
4. Transition in a steady circular arc to the OEI climb-out flight path angle and speed

5. Steady climb at V_2 (climb safety speed) and OEI power until an obstacle height h_o is reached

During the takeoff roll (segments 1, 2, and 3), the force balance equation is:

$$\frac{dV}{dt} = T - D - \mu(mg - L). \tag{4.16}$$

where V is the velocity, T, D, and L are total thrust, drag, and lift forces, μ is the effective "friction" coefficient accounting for rolling resistance and braking, m is the aircraft weight, and g is the acceleration due to gravity.

The accelerate-go distance combines segments 1, 2, 4, and 5, while the accelerate-stop distance includes 1 and 3.

$V_R = 1.1V_{\text{stall}}$ and $V_2 = 1.2V_{\text{stall}}$, where stall speed is calculated as a function of maximum takeoff weight (MTOW) [175]. The speed at "rest" is assumed to be 1 m/s in order to avoid singularities in analysis codes at zero forward speed. Default μ is 0.03 during the takeoff roll, and 0.4 during emergency braking in a rejected takeoff, though this could be overridden to simulate wet or snowy runways or improved aircraft brake systems. The obstacle clearance height is set at 35 feet by default (14 CFR 23), but can be trivially changed to 50 feet to model a Part 25 transport aircraft.

Equation (4.16) is an ordinary differential equation and must be integrated to obtain distances for segments 1, 2 and 3. For example, the distance travelled during run up to decision speed (segment 1) is:

$$R_{V_1} = \int_{V_0}^{V_1} \frac{dr}{dt}\frac{dt}{dV}\,dV = \int_{V_0}^{V_1} \frac{V}{a}\,dV. \tag{4.17}$$

The integration method is described in more detail in Section 4.4

The `takeoff` module uses a Newton solver to vary the chosen V_1 speed until the accelerate-go and accelerate-stop distances are equal, or until the accelerate-go distance is longer than the accelerate-stop distance and $V_1 = V_R$. The accelerate-go distance is then equal to the *balanced*

field length, which can be used as an optimization or sizing constraint.

4.3.2 Mission Analysis

The `mission` module accomplishes two main tasks: setting condition-dependent control inputs necessary for steady flight, and integrating quantities such as fuel burn and energy over the mission profile. Each mission is defined as a series of segments. For example, a transport mission without a reserve would be modeled as follows:

- Climb at constant vertical speed and indicated airspeed to the cruise altitude.
- Cruise at constant indicated airspeed and altitude.
- Descent at constant indicated airspeed and vertical speed to the landing altitude.

Figure 4.2 illustrates flight conditions and aircraft states for a representative hybrid-electric aircraft mission.

During mission analysis, the aircraft is treated as a point mass, which changes as fuel burns. At each flight condition, OpenConcept calculates the value of the residual equation

$$\mathcal{R}_{\text{thrust}} = T - D - mg\sin(\gamma) \tag{4.18}$$

where γ is the flight path angle. OpenMDAO's Newton solver drives these residuals to zero at every flight condition in the mission by varying the primary thrust control parameter (usually either motor or engine throttle). If more than one independent thrust control parameter is available (for example, high-lift and cruise propellers as in the X-57), the user can specify some of the parameters and let the Newton solver find the remaining one. Alternatively, an optimizer can find the optimal value for every control parameter by treating the thrust residual as an equality constraint.

Conceptually, during each iteration of the mission solver, the following steps occur:

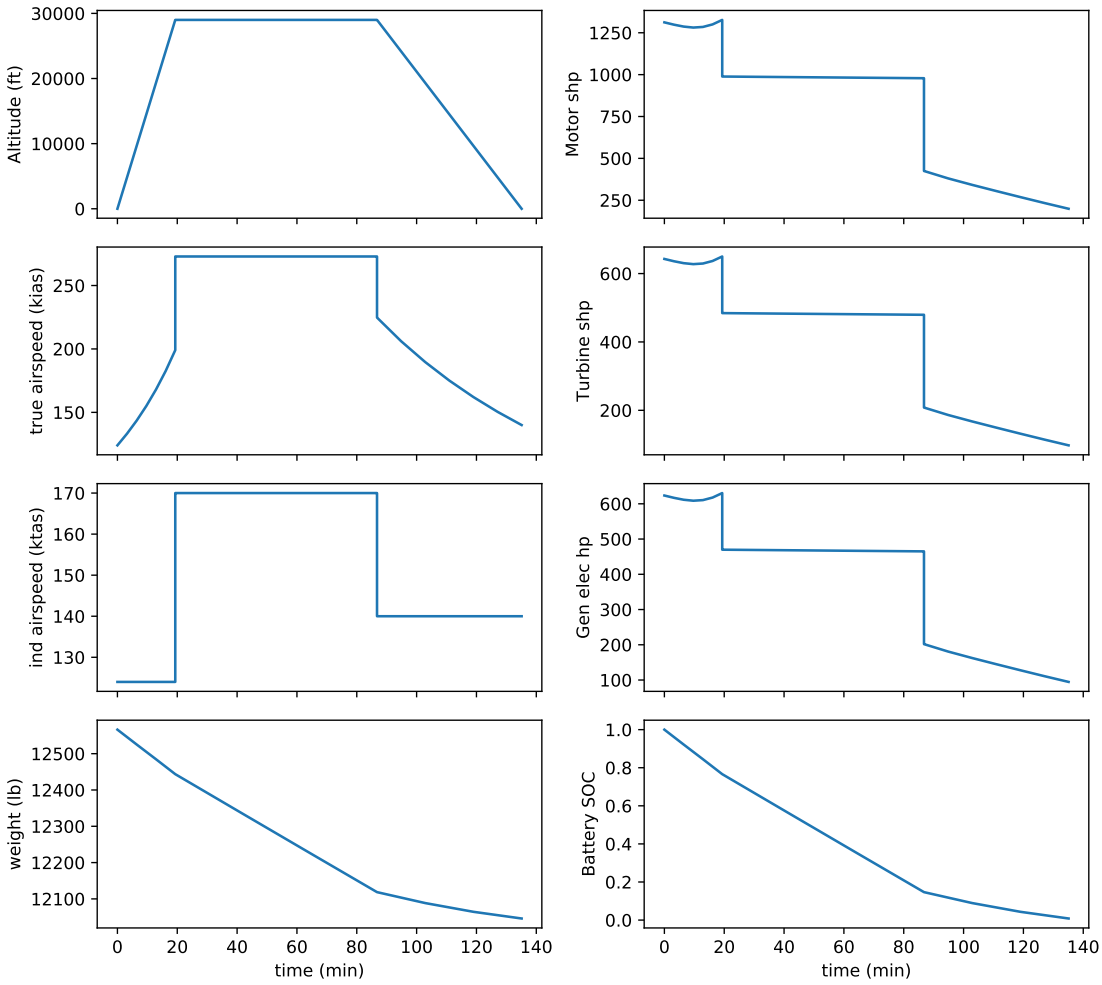


Figure 4.2: Representative mission profile from Section 4.5 case study ($e_b = 500$ Wh/kg, range$= 500$ nmi)

1. Generate vectors representing the flight condition at each point in time during the mission.

2. Calculate atmospheric properties [176].

3. Compute climb and descent phase distances and times to obtain cruise distance and time.

4. Run an OpenConcept propulsion model at each flight condition to obtain fuel flows, battery loads, thrusts, and constrained quantities like heat output.

5. Integrate fuel flow and battery load with respect to time using Simpson's rule (as described

in Section 4.4) to obtain aircraft weight and battery SOC vectors.

6. Calculate flight C_L and drag.

7. Calculate the thrust-drag residual.

Once the Newton solver has converged the mission, thrust balances drag (and weight, if climbing or descending), lift matches weight, and any hybrid turbomachinery components are producing the correct shaft power to meet electrical loads. In addition, the user needs to make sure that the aircraft's design weights are consistent with the mission being flown. This can be posed as a set of inequality constraint equations:

$$\begin{aligned} m_{\text{TO}} &\geq m_f + m_{\text{empty}} + m_{\text{payload}} \\ m_{\text{f,max}} &\geq m_f \end{aligned} \tag{4.19}$$

where m_{TO} is the takeoff weight. For aircraft with batteries, one more constraint is required:

$$E_{\text{b,max}} \geq E_{\text{b,used}} \tag{4.20}$$

The Newton solver does not automatically drive these constraints to zero, which enables analysis of aircraft where not all of the fuel or battery is consumed during a mission. Instead, the optimizer should enforce these. Takeoff weight (TOW) and battery weight are then set to the minimum required to fly the mission. When using the `mission` module without an optimizer in the loop, the user must manually ensure that the mission weights are feasible.

4.4 ODE Integration

For performance, OpenConcept uses vectorized computations in each segment, meaning that time-marching ODE integration approaches cannot be used (since vectorized quantities must be

computed all-at-once). Instead, OpenConcept uses an implementation of Simpson's Rule for numerical integration. An integral can be approximated using Simpson's rule as follows:

$$\int_{x_L}^{x_U} f(x)\,dx \approx \frac{1}{3}\Delta x(f_0 + 4f_1 + 2f_2 + 4f_3 + 2f_5 + ... + 2f_{2N-2} + 4f_{2N-1} + f_{2N}) \tag{4.21}$$

$$\Delta x = \frac{x_U - x_L}{2N}, \tag{4.22}$$

where N is the number of Simpson subintervals and Δx is the constant spacing between the points f_i. This method always requires evaluating a function at $2N + 1$ points. Simpson's rule integrates polynomials up to third order exactly [177].

The number of Simpson subintervals in each mission segment is a major driver of the size of the linear algebra problem solved by OpenMDAO and thus, computation time. I performed a convergence study of the integration method using a representative hybrid aircraft model which uses significant battery and fuel energy (described in Section 4.5). Table 4.1 illustrates that very accurate fuel burn and BFL results can be obtained with a relatively minimal number of points per mission segment. Five intervals per segment is the default and integrates fuel burn and BFL nearly exactly.

Table 4.1: Simpson integration convergence

Simpson Intervals	Points	Fuel Burn	FB Error	BFL	BFL Error
15	31	**314.18975**	0.000%	**1357.0047**	0.00%
8	17	**314.18975**	0.000%	**1357.0047**	0.00%
5	11	**314.19**058	0.000%	**1357.00**53	0.00%
4	9	**314.19**259	0.001%	**1357**.1920	0.01%
3	7	**314.19**254	0.001%	**1357**.2149	0.02%
2	5	**314.19**330	0.001%	**135**6.2602	−0.05%
1	3	**314**.31917	0.041%	**1357**.5849	0.04%

(a) Daher TBM 850 (photo by Gyrostat, CC-BY-SA)

(b) Beechcraft King Air C90GTi (photo by Joao Carlos Medau, CC-BY)

Figure 4.3: OpenConcept benchmark aircraft

4.5 Case Study: Design of an Electric Aircraft for Minimum Operating Cost

To validate the code, I conducted a case study with the notional goal of converting a Beechcraft King Air C90GT to series hybrid electric propulsion.

4.5.1 Conventional Baseline

To test OpenConcept's propulsion modeling and analysis routines on a simple case, I first modeled a single engine turboprop, the SOCATA/Daher TBM 850. Structural and system weights were estimated using textbook formulas [174, 175], with a constant factor of 1.6 applied to structural weight in order to match published empty weights [178]. Cruise drag was estimated using a drag polar formulation, with induced drag and zero-lift drag estimated using tops-down methods [175]. $C_{L\text{max}}$ was set to match the nominal takeoff rotation speed to handbook values [179]. Initial model runs using the *a priori* estimate of C_{D_0} resulted in a fuel burn total close to the published value; I adjusted C_{D_0} to match fuel burn and maximum range.

Propeller maps from manufacturers are closely held. For the case study, propeller efficiency was estimated from a published map [173]. I "compressed" this map in the C_P axis so that the peak efficiency point better matched the anticipated operating point, and visually extrapolated the map into the higher C_P region. I also adjusted the very low speed propulsive efficiency downward to reduce spuriously high thrust levels during the takeoff roll. The balanced field length for the single-engine TBM is simply the takeoff distance with full takeoff power since the one-engine-inoperative takeoff distance is not defined.

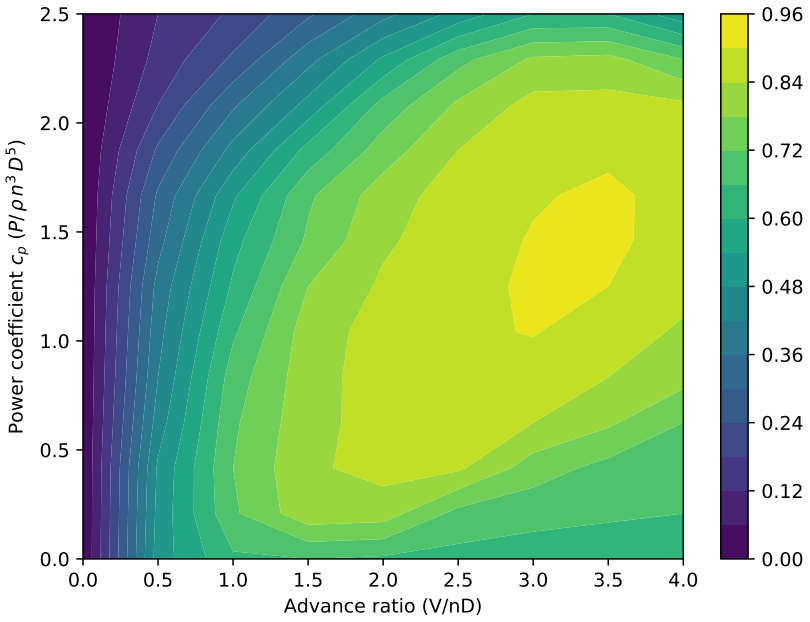


Figure 4.4: Propeller efficiency map for the Section 4.5 case study

I modeled the Beechcraft King Air C90GT in a nearly identical way, except that the King Air has two engine and propeller components, and I applied a structural weight factor of 2.0 to match published empty weight [180]. On this King Air model, the PT6A-135A engine is derated by about 25% (from 750 hp to 550 hp) for structural reasons. Balanced field length was calculated using 25% derated takeoff power, and zero power in Engine 2 following V_1. Table 4.3 illustrates input and model output data for the TBM 850 and King Air. Balanced field length for the King

Air matched published figures quite closely, but the simulated fuel burn was about 25% lower. It is possible that the propeller map is not representative of the King Air at the cruise flight condition, or that in reality the derated PT6A-135A engine is operating significantly below peak efficiency at the specified cruise throttle setting and altitude. I did not attempt to vary C_{D0} upward enough to match the published (higher) fuel burn for the King Air.

4.5.2 Sizing the Propulsion System of a Series-Hybrid Conversion

Next, I examined the feasibility of a drop-in replacement of the twin turboprop architecture of the King Air with a series hybrid system. The purpose of this study was *not* to prove or disprove the viability of electric propulsion for light aircraft retrofit, but simply illustrating the use of OpenConcept for an aircraft study. A modern clean-sheet design would have weight and drag advantages which may make electric propulsion feasible at longer ranges and lower e_b.

Following federal regulations for commuter aircraft, I assume that any single component of the propulsion system may fail and that the aircraft must be able to continue safe flight and landing during the takeoff phase. To achieve this, the series hybrid architecture includes the following features:

- Two motors and propellers (providing redundancy in the event of motor or propeller failure)
- Batteries split into at least two independent packs, with battery power alone used for takeoff (providing redundancy against single engine failure on takeoff)

These conditions ensure a level of safety on takeoff equivalent to a twin turboprop. Specific power, efficiency, and cost assumptions for individual powertrain components are stated in Table 4.2.

[1] does not include 104kg base wt

Table 4.2: Powertrain technology assumptions

Component	Specific Power (kW/kg)	Efficiency	Cost	PSFC (lb/hp/hr)
Battery	5.0	–	\$50/kg	–
Motor	5.0	97%	\$100/hp	–
Generator	5.0	97%	\$100/hp	–
Turboshaft/Prop	7.15[1]	–	\$775/shp	0.6

The `scipy.optimize` SLSQP algorithm was used to size the propulsion system components (motor, engine, and generator sizing) for minimum fuel burn on the design mission. MTOW, wing area, and all other parameters remain equal to the King Air baseline. The optimization problem is formulated as:

$$
\begin{aligned}
\text{minimize: } & \text{fuel burn} \\
\text{by varying: } & \\
& m_b \\
& P_{m,\text{rated}} \\
& P_{\text{ts,rated}} \\
& P_{g,\text{rated}} \\
& H_E \text{ (degree of hybridization w.r.t energy)} \\
\text{subject to scalar constraints: } & \\
\mathcal{R}_{\text{TOW}} &= m_{\text{TO}} - m_f - m_{\text{empty}} - m_{\text{payload}} - m_b \geq 0 \\
\mathcal{R}_b &= E_{\text{b,max}} - E_{\text{b,used}} \geq 0 \\
\text{BFL} &\leq 4452 \text{ ft (no worse than baseline)} \\
\text{and vector constraints: } & \\
\boldsymbol{P}_m &\leq 1.05 P_{m,\text{rated}} \\
\boldsymbol{P}_{\text{ts}} &\leq P_{\text{ts,rated}} \\
\boldsymbol{P}_g &\leq P_{g,\text{rated}} \\
\boldsymbol{P}_b &\leq m_b \cdot p_b
\end{aligned}
$$

The optimizer successfully sized the motor, generator, battery, and turboshaft and found the mix of electric and fuel energy (*degree of hybridization*, H_E) which minimized fuel burn. Design variables and simulation outputs are listed in Table 4.3. Using battery specific energy of 750 Wh/kg, the series hybrid conversion could not meet the 1000nmi design range of the original

King Air. At a maximum range of 762 nmi, the optimizer converged on a design with essentially the minimum allowable amount of battery (sized by power, not energy, at takeoff). Since the 762 nmi mission was at the very limit of the airplane's capability, I changed the design range to 500 nmi and resized the propulsion system again. This time, e_b significantly affected the sizing (at 250, 500, and 750 Wh/kg). All three designs had identical motor sizing (to meet the takeoff constraint at MTOW), but the generators and engine power increased with decreasing e_b due to the larger fraction of power from fuel. The 750 Wh/kg case burned 38% less fuel than the 250 Wh/kg case.

To further validate the speed and flexibility of OpenConcept, in 2018 I conducted a rough feasibility study of an all-electric conversion of the Cessna C208B Grand Caravan. The objective was to assess the feasibility of using all-electric propulsion to handle cargo flights of one hour or less, as proposed by the start-up firm magniX. From start to finish (including gathering input data on the Grand Caravan online and assessing control surface areas using photogrammetric methods), the study took less than 90 minutes. The results are not tabulated here, but indicated that the idea could be feasible with current technology (which, of course, it was — magniX flew the eCaravan in 2020).

4.5.3 Multidisciplinary Design Optimization for Minimum Fuel Burn

Following successful demonstration of the simple sizing capability, I added MTOW, fuel volume, wing area, and prop diameter variables to the optimization problem. The expanded optimization problem is:

$$
\begin{aligned}
\text{minimize: } & \text{fuel burn} + 0.01\text{MTOW} \\
\text{by varying:} & \\
& \text{MTOW} \\
& S_{ref} \\
& d_{prop} \\
& m_b \\
& P_{m,\text{rated}} \\
& P_{\text{ts,rated}} \\
& P_{g,\text{rated}} \\
& H_E \text{ (degree of hybridization w.r.t energy)} \\
\text{subject to scalar constraints:} & \\
& \mathcal{R}_{\text{TOW}} = m_{\text{TO}} - m_f - m_{\text{empty}} - m_{\text{payload}} - m_b \geq 0 \\
& \mathcal{R}_b = E_{\text{b,max}} - E_{\text{b,used}} \geq 0 \\
& \mathcal{R}_{\text{vol}} = m_{\text{f,max}} - m_f \geq 0 \\
& \text{BFL} \leq 4452 \text{ ft (no worse than baseline)} \\
& V_{\text{stall}} \leq 81.6 \text{ kt (no worse than baseline)} \\
\text{and vector constraints:} & \\
& \boldsymbol{P}_m \leq 1.05 P_{m,\text{rated}} \\
& \boldsymbol{P}_{\text{ts}} \leq P_{\text{ts,rated}} \\
& \boldsymbol{P}_g \leq P_{g,\text{rated}} \\
& \boldsymbol{P}_b \leq m_b \cdot p_b
\end{aligned}
$$

Results for 250, 500, 750, and 1000 Wh/kg on a 500 nmi design mission are listed in Table 4.3. In these four optimizations, hints of discontinuities emerge in the hybrid electric design space. At 1000 and 750 Wh/kg, the airplane prefers to use no fuel and fly the design mission completely on batteries. At 500 Wh/kg, the optimizer hits the MTOW upper bound (5700kg, above which EASA and the FAA require pilots to obtain a type rating). It uses as much battery as possible, supplementing with just enough fuel to meet the required range. At 250 Wh/kg, the optimizer prefers to reduce battery weight to the takeoff power-constrained minimum and *reduce* MTOW below the baseline.

This fuel burn optimization was repeated over a grid of 252 individual combinations of design range and e_b (Figure 4.5). The primary finding from this set of optimizations is that, for an airplane

with King Air-like structural efficiency and aerodynamics, hybrid propulsion is generally only preferable to all-fuel or all-electric operation when an upper limit exists on MTOW. Practical MTOW limits might include a retrofit application with an existing airframe, or regulatory limits (as mentioned above). This finding may not apply generally to more aerodynamically and structurally efficient clean-sheet designs.

At short range and high e_b (the upper left corner of Figure 4.5), the optimizer can eliminate fuel altogether. Since fuel burn is zero everywhere in this triangular region, using pure fuel burn as the objective function will fail to converge on a reasonable airplane. A small additional term proportional to MTOW was added to the objective function in order to encourage the optimizer to reduce MTOW (and therefore, battery weight) as much as possible, once fuel burn is reduced to zero.

4.5.4 Multidisciplinary Design Optimization for Minimum Cost

I ran an additional grid of 252 optimizations with respect to operating cost. The optimization problem is formulated as follows:

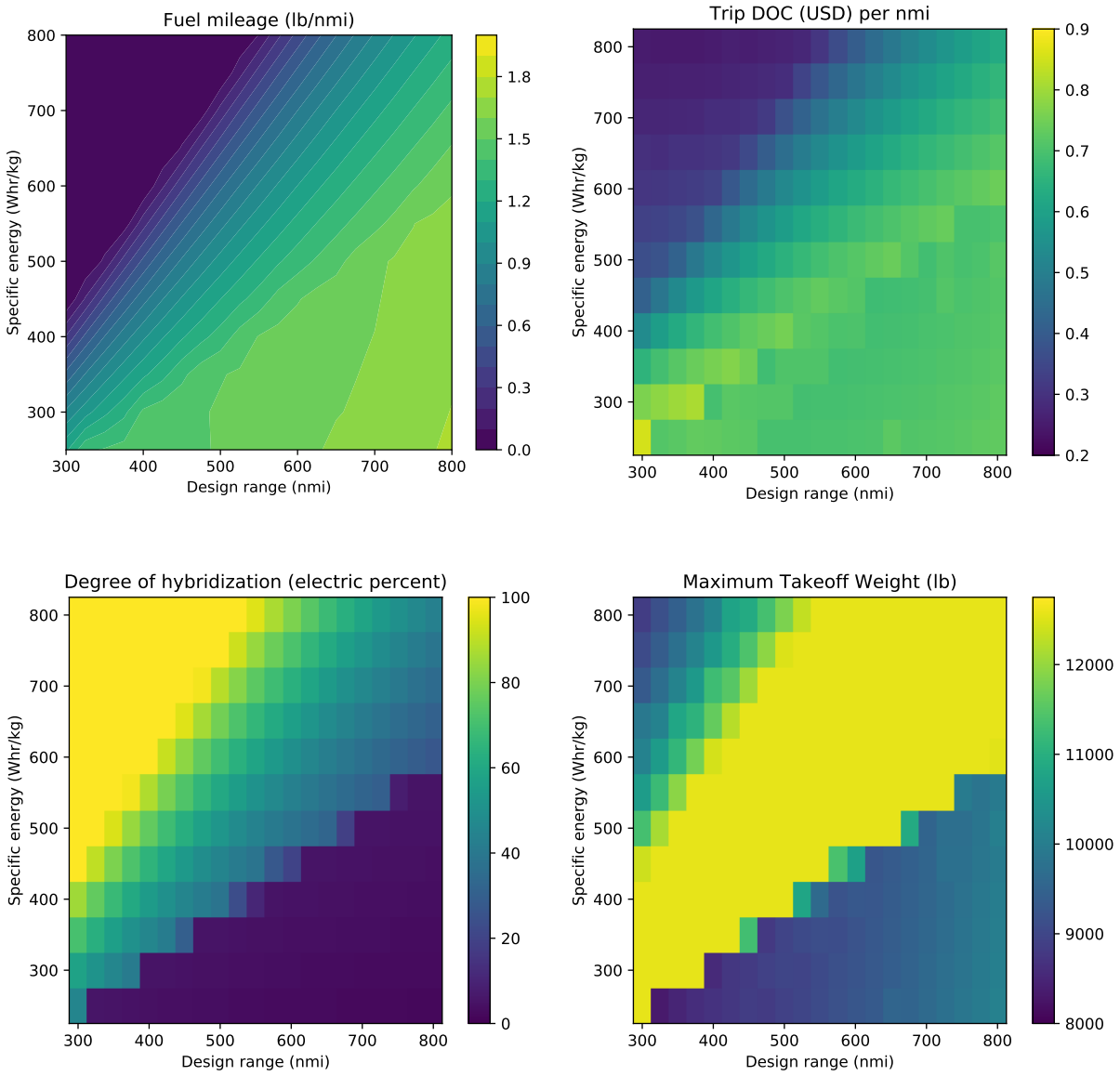


Figure 4.5: Minimum fuel burn MDO results.

minimize: trip cost

by varying:

MTOW

S_{ref}

d_{prop}

m_b

$P_{m,\text{rated}}$

$P_{\text{ts,rated}}$

$P_{g,\text{rated}}$

H_E (degree of hybridization w.r.t energy)

subject to scalar constraints:

$$\mathcal{R}_{\text{TOW}} = m_{\text{TO}} - m_f - m_{\text{empty}} - m_{\text{payload}} - m_b \geq 0$$

$$\mathcal{R}_b = E_{\text{b,max}} - E_{\text{b,used}} \geq 0$$

$$\mathcal{R}_{\text{vol}} = m_{\text{f,max}} - m_f \geq 0$$

$$\text{BFL} \leq 4452 \text{ ft (no worse than baseline)}$$

$$V_{\text{stall}} \leq 81.6 \text{ kt (no worse than baseline)}$$

and vector constraints:

$$\boldsymbol{P}_m \leq 1.05 P_{m,\text{rated}}$$

$$\boldsymbol{P}_{\text{ts}} \leq P_{\text{ts,rated}}$$

$$\boldsymbol{P}_g \leq P_{g,\text{rated}}$$

$$\boldsymbol{P}_b \leq m_b \cdot p_b$$

A notional trip cost model was constructed as follows:

$$
\begin{aligned}
\text{Trip cost} &= c_f + c_{\text{electricity}} + c_b + c_{\text{depreciation}} \\
&\text{where:} \\
c_f &= (\$2.50/\text{gal})\, m_f/\rho_f \\
c_{\text{electricity}} &= (\$36.00/\text{MWh})\, E_{\text{b, used}} \\
c_b &= (\$50.00/\text{kg})\, m_b\, /n_{\text{batt cycles}} \\
c_{\text{depreciation}} &= c_{\text{aircraft}}/n_{\text{flights,daily}}/365 \text{ days}/n_{\text{years}} \\
&\text{and:} \\
c_{\text{aircraft}} &= (\text{OEM premium})\ (c_{\text{airframe}} + c_{\text{engine}} + c_{\text{motors}} + c_g) \\
c_{\text{airframe}} &= (\$277/kg)\, (\text{OEW} - m_{\text{engines}}) \\
c_{\text{engine}} &= (\$775/\text{shp})\, P_{\text{engine,rated}} \\
c_{\text{motors}} &= (\$100/\text{shp})\, P_{\text{motors,rated}} \\
c_g &= (\$100/\text{shp})\, P_{g,\text{rated}} \\
n_{\text{batt cycles}} &= 1500 \\
n_{\text{flights,daily}} &= 5 \\
n_{\text{years}} &= 15 \\
\text{OEM premium} &= 1.1
\end{aligned}
$$

Fuel and electricity prices were picked as representative wholesale values for 2018. Battery cost was estimated assuming \$200 per kWh, and specific energy of 250 Wh/kg. Airframe cost factor includes everything except propulsion components and was estimated based on general light turboprop pricing trends (in current USD currency). Engine price per shaft horsepower was estimated based on new PT6A prices listed on various online marketplaces. The original equipment manufacturer (OEM) premium assumes a 10% operating profit above Tier 1 supplier costs. Note that the cost model does *not* contain any contribution from maintenance cost (for either conventional or electric propulsion), nor does it include crew costs or landing fees. Crew cost is neglected, as the missions being tested are equal in block time. Maintenance and crew cost could be easily added to the cost calculation for future studies.

The price of aerospace-certified propulsion motors and generators, and the cycle life of aerospace-grade batteries, were unknown since they were not commercially available at the time this study was conducted — certification rules for such motors were only established in Europe in 2021.

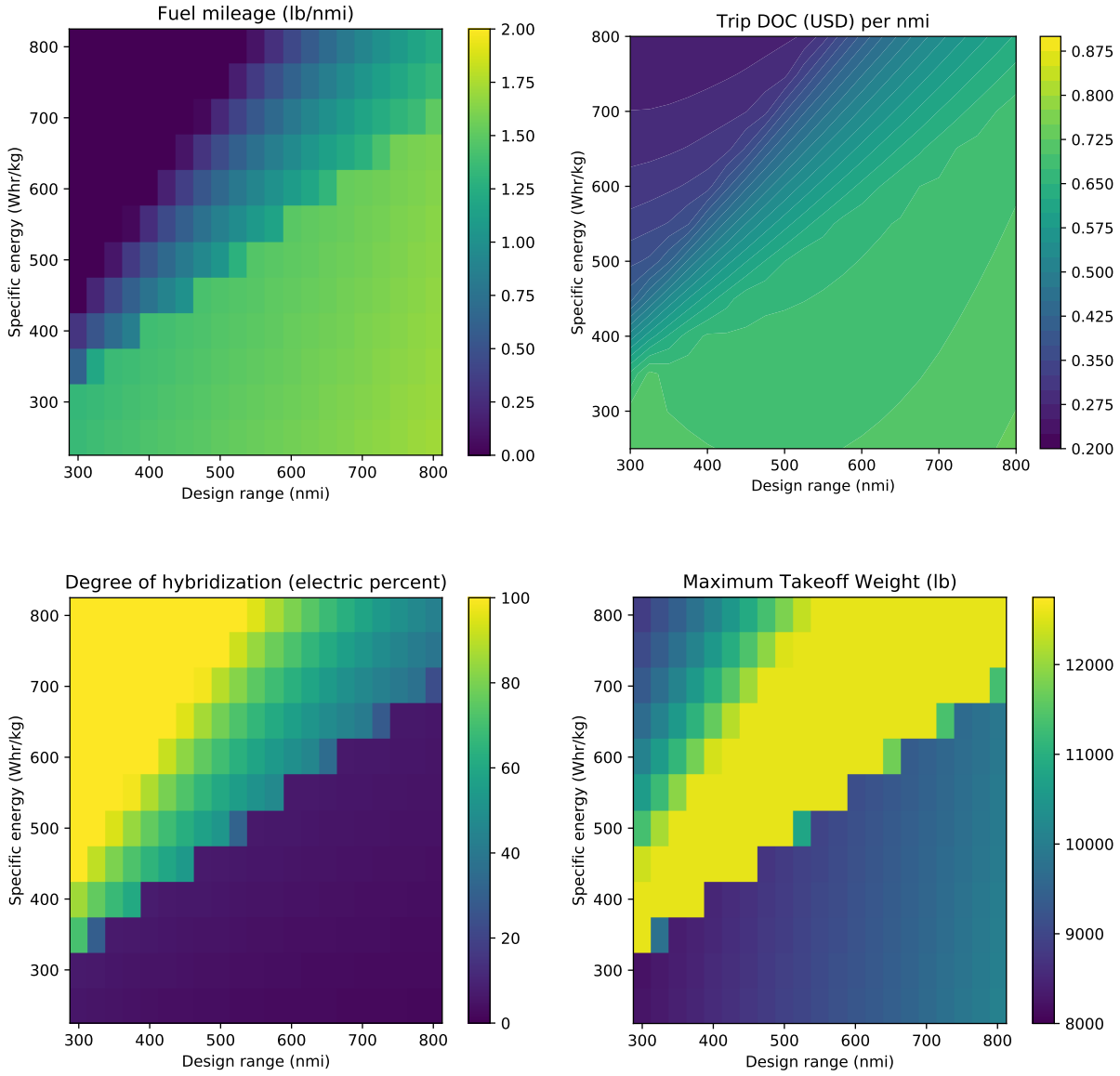


Figure 4.6: Minimum cost MDO results

The maintenance cost of electric propulsion is also unknown, although it can be expected to be lower than for turboprop engines on an hourly basis. The effect of these economic assumptions on optimal design can be easily tested using OpenConcept.

Figure 4.6 shows the cost, fuel burn, degree of hybridization, and MTOW for the cost-optimized aircraft. Compared to Figure 4.5, optimizing for cost has moved the electric propulsion feasibility line up and to the left. Under this (debatable) set of economic assumptions, electric propulsion is more favorable for fuel burn reduction than for economics, at least at moderate specific energy levels.

In order to provide a fair comparison between a conventional architecture and the series hybrid architecture, I ran an additional 21 optimizations of a *conventional* twin turboprop, but allowed the optimizer to perform full MDO (same rules as for the hybrid MDO study, but without the electric propulsion design variables or constraints). Figure 4.7 shows the relative difference between the cost-optimized hybrid electric aircraft and the cost-optimized conventional aircraft at the same design range. In the lower right half, the series hybrid (effectively turboelectric) design actually costs more to operate than a conventional twin turboprop. A breakeven point runs nearly linearly from lower left to upper right; conventional and electric are economically equivalent along this line. Above the breakeven line (in the hybrid regime), costs fall rapidly as the optimizer can trade fuel for batteries and reduce the size of the turbogenerator system. Once turbogenerator power is reduced to zero, the (low) costs remain relatively stable even as battery specific energy improves. Figure 4.7 illustrates that the potential for cost savings is high if a significant proportion of battery power can be used. Turboelectric propulsion by itself is not an efficient replacement for turboprop engines, at least when no ancillary aerodynamic or propulsive efficiency benefit can be realized. If maintenance costs were modeled, the breakeven line would move down and to the right.

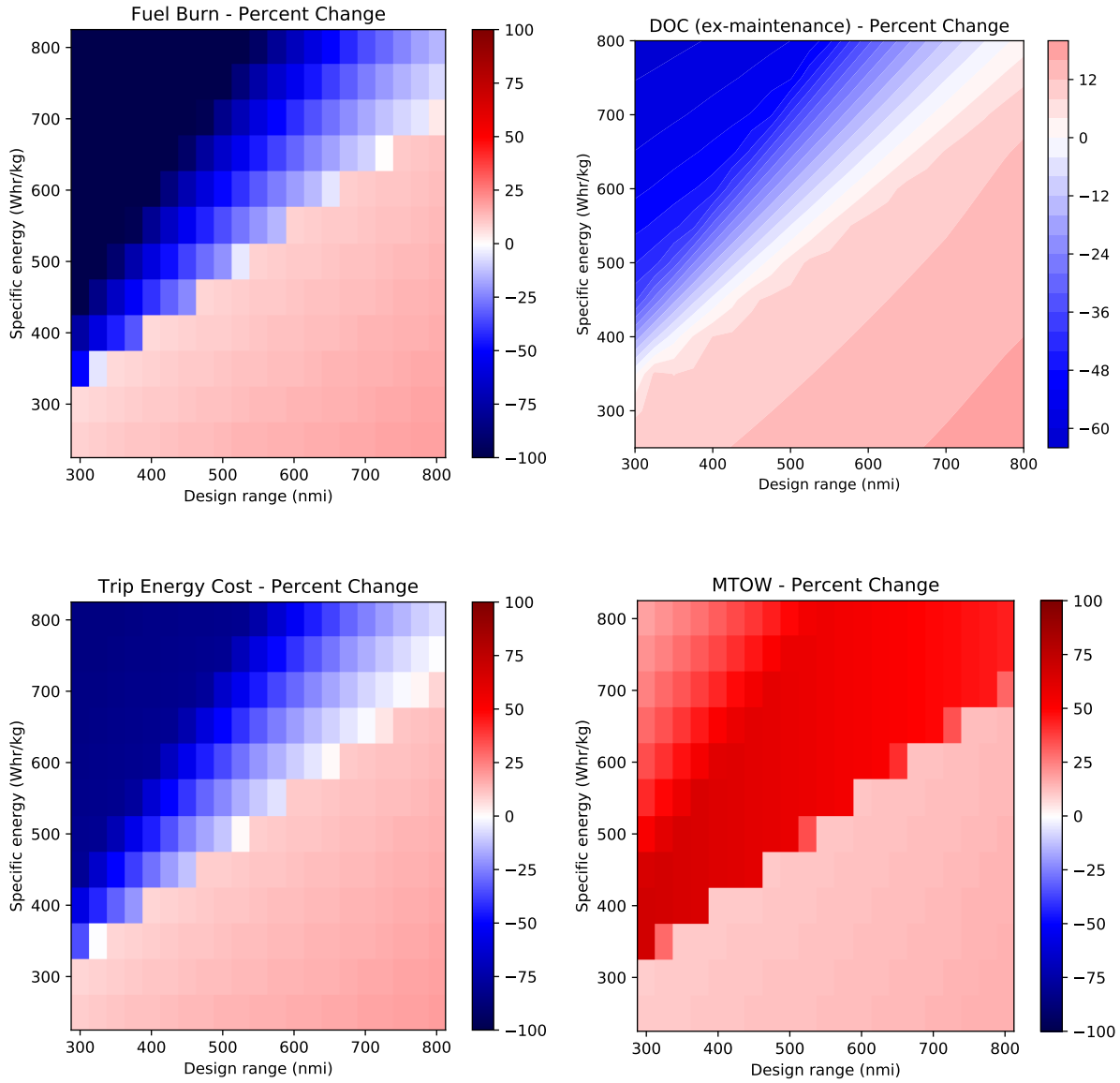


Figure 4.7: Minimum cost hybrid versus minimum cost conventional

4.6 Using OpenConcept for Technology Assessment

A fundamental challenge faced by aircraft designers in industry is finding the right time to incorporate new technology onto an aircraft family. We learned in Section 4.5 that a tipping point exists in the design space: an optimal conventional airplane will be as light as possible, but once batteries are economically favored, the best aircraft is as heavy as possible with batteries. This tendency makes retrofitting a hybrid or all-electric powertrain to an existing airframe infeasible, assuming the user wishes to fly similar mission ranges. Therefore, clean sheet designers must be able to predict at what point in time electrical component and battery technology will become economically favored for the chosen mission. OpenConcept can conduct the sort of low-cost, moderate-fidelity tradespace exploration required to answer this question.

The Section 4.5 hybrid King Air airframe was heavy and had relatively high parasitic drag. To simulate improved airframe technology, the C_{D_0} was reduced from 0.022 to 0.018, and the structural weight factor reduced from 2.0 to 1.5 (a 33% structural weight reduction; for example, using carbon composites and reclaiming margin). I then ran an additional 252 MDO scenarios on the same range–versus–e_b grid, using the minimum cost optimization rules presented in Section 4.5. I also ran the full span of design ranges on the conventional twin turboprop model, using the same structural weight and drag reductions, to provide a fair comparison. The total time (including setup) to run the optimizations and analyze results was approximately 9 hours on a laptop. Figure 4.8 illustrates the fuel burn, cost, weight, and energy.

I found that, as predicted, the lower-weight and lower-drag airplane favored electric propulsion at lower e_b across all ranges. The economic breakeven line moved down about 50 Wh/kg, nearly into the present-day feasibility zone for a 300 nmi mission (Figure 4.9). The line is also shallower, meaning that the effect of the weight and drag reduction is felt more strongly at longer ranges.

The published literature has contributed to industry understanding of the primary technical and performance drivers of electric flight. However, results from published studies are not general

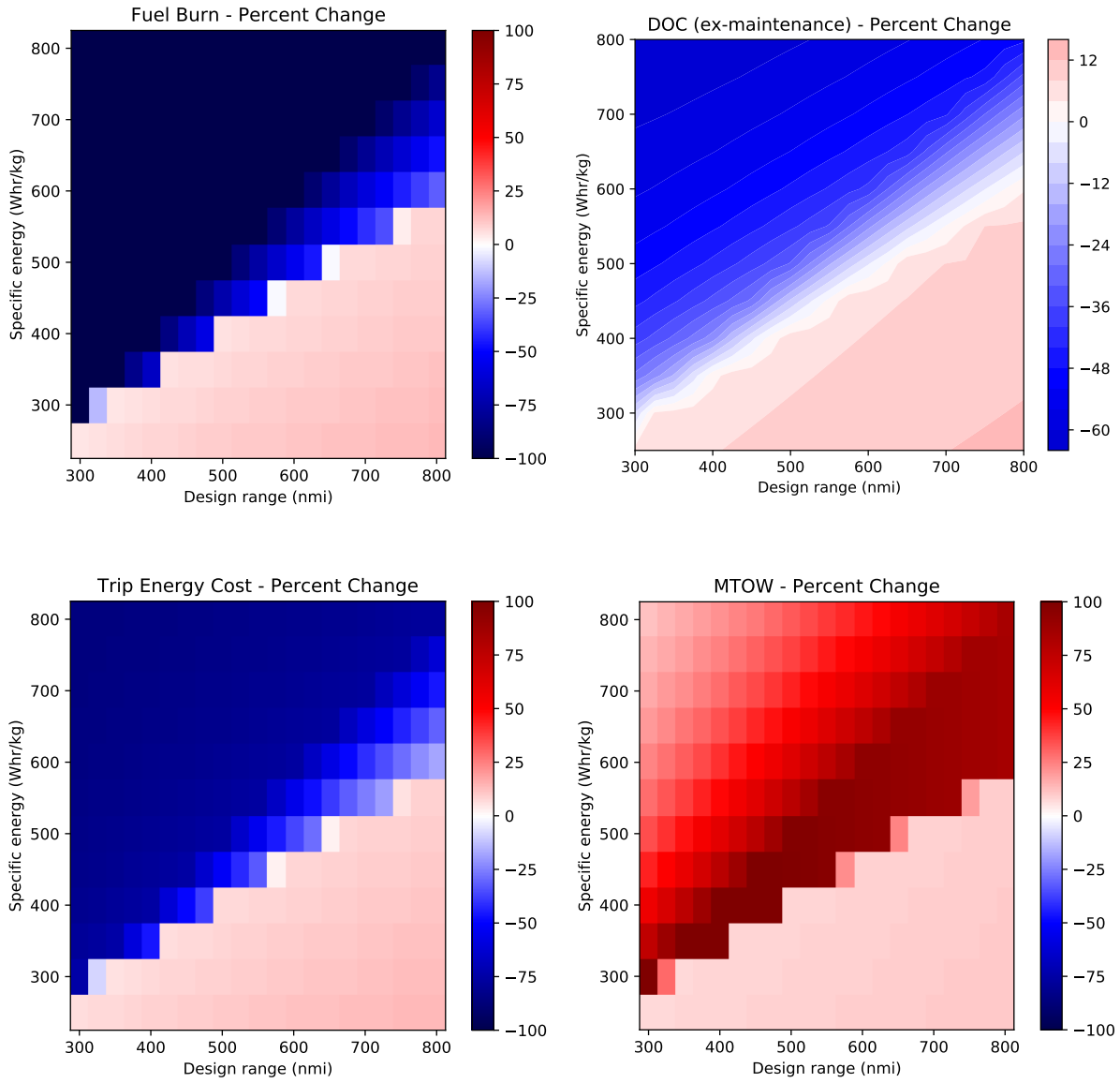


Figure 4.8: Minimum cost hybrid versus minimum cost conventional with reduced structural weight and drag

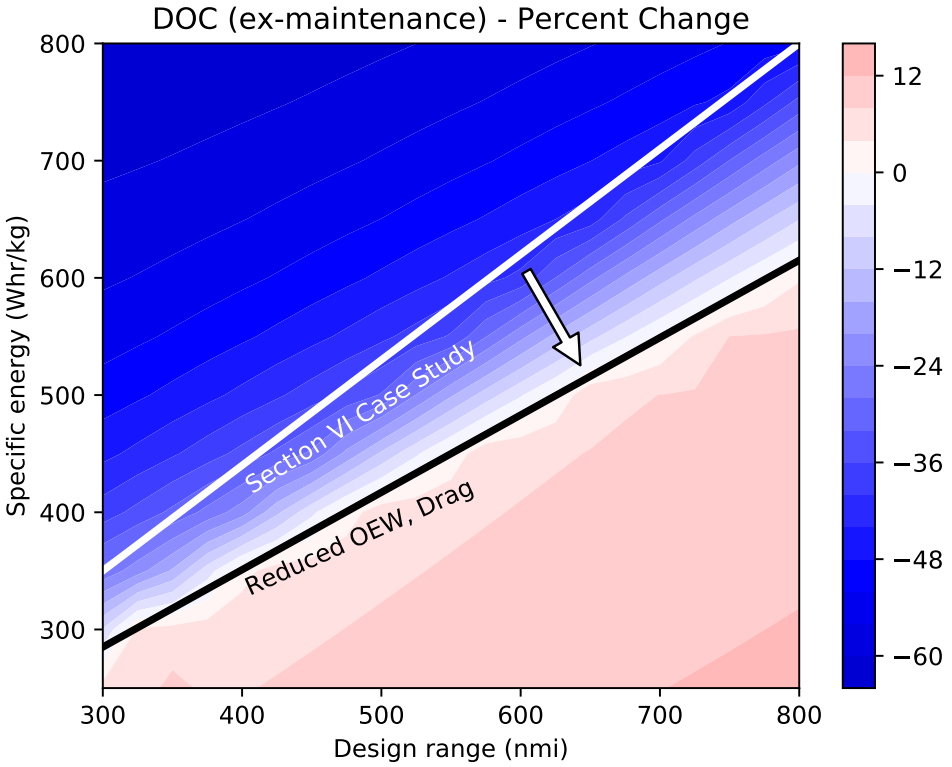


Figure 4.9: Effect of technology inputs on hybrid propulsion breakeven line: improved airframe technology makes electric propulsion economically feasible at lower specific energies

enough to tell a design team when to implement electric propulsion for their particular set of requirements. Using OpenConcept, designers can rapidly test their internal assumptions and generate large amounts of trade study data with modest setup time and computational resources. Similar studies could be conducted with respect to specific power or efficiency of individual electrical components.

Table 4.3: Baseline analysis, component resizing, and MDO results (minimum fuel burn objective)

	TBM850 (Model)	TBM850 (Published)	King Air C90GT (Model)	King Air C90GT (Published)	Hybrid Conversion - Max Range	Hybrid Conversion 750	Hybrid Conversion 500	Hybrid Conversion 250	Hybrid MDO 1000	Hybrid MDO 750	Hybrid MDO 500	Hybrid MDO 250	King Air MDO 1000nmi	King Air MDO 500nmi
Optimization Rules	Analysis	Analysis	Analysis	Analysis	Comp Sizing	Comp Sizing	Comp Sizing	Comp Sizing	MDO	MDO	MDO	MDO	MDO	MDO
Specific Energy (Wh/kg)	–	–	–	–	750	750	500	250	1000	750	500	250	–	–
Design Range (nmi)	1250	1150+100	1000	894+100	761.6 (max)		500			500			1000	500
MTOW (lb)	7392	7392	10100	10100		10100			10156	12505	**12566**	8912.8	9367	7941
OEW (lb)	4756	4762	7177	7150	7433.9	7197	7328	7410	6755	7594	7967	6828.5	6827	6279
Max Fuel Wt (vol limit, lb)	2000	2000	2570	2570		2570			**1102**	**1102**	**1102**	**1102**	1540	**1102**
MLW (lb)	7000	7000	9600	9600		9600				9600			9600	
Rated TO SHP (each)	850	850	550	550	527.2	527.2	572.2	527.2	519.4	640.0	649.9	456.6	511.5	434.6
Turboshaft SHP (each)	850	850	550	550	1061.0	629.0	867.6	1018.2	–	–	641.1	940.0	511.5	434.6
Generator SHP	–	–	–	–	1029.2	610.6	841.5	987.6	–	–	630.4	907.9	–	–
Prop Diameter (ft)	7.58	7.58	7.50	7.50		7.50			**7.22**	**7.22**	**7.22**	**7.22**	**7.22**	**7.22**
PSFC (lb/hp/hr)	0.60		0.60			0.60				0.6			0.6	
Wing Ref Area (ft^2)	193.8	193.8	294.0	294.0		294.0			296.0	364.6	366.3	259.8	273.1	231.5
Wingspan (ft)	41.5	41.5	50.2	50.2		50.2			50.4	55.9	56.1	47.2	48.4	44.6
Aspect Ratio	8.95	8.95	8.58	8.58		8.58				8.58			8.58	
Flaps-Down $C_{L\max}$	1.70		1.52			1.52				1.52			1.52	
Oswald Efficiency	0.78		0.80			0.80				0.80			0.80	
C_{D0} at Cruise	0.0205		0.0220			0.0220				0.0220			0.0220	
C_{D0} at Takeoff	0.0300		0.0290			0.0290				0.0290			0.0290	
Takeoff Rotation Speed (kias)	89.6	90	89.8	90	89.8	89.8	89.8	89.8	89.8	89.8	89.8	89.8	89.8	89.8
Battery Wt (lb)	–	–	–	–	381.4	1397.9	1079.2	754.6	2400.8	3911.4	3077.6	330.3	–	–
Takeoff Battery %	–	–	–	–		100%				100%			–	–
Cruise Battery %	–	–	–	–	5.0%	43.7%	22.3%	8.6%	100.0%	100.0%	52.5%	3.6%	–	–
Design Payload (lb)	1000	1000	1000	1000		1000				1000			1000	
Cruise Speed (KIAS)	201	201	170	170		170				170			170	
Cruise Altitude (ft)	28000	28000	29000	29000		29000				29000			29000	
Climb Rate (ft/min)	1500		1500			1500				1500			1500	
Climb Speed (KIAS)	124		124			124				124			124	
Descent Speed (KIAS)	140		130			130				130			130	
BFL, SL, ISA+0 (ft)	2844	2838	4452	4519	**4452**	**4452**	**4452**	**4452**	**4452**	**4452**	**4452**	**4452**	**4452**	**4452**
BFL % Error	0.2%		-1.5%											
Cruise Fuel Flow (lb/hr)	414	415	468	612										
Cruise Fuel % Error	-0.3%		-23.5%											
Design Mission Fuel (lb)	1605.8		1663.8		1284.2	504.5	692.7	811.0	0.0	0.0	520.0	754.0	1540.3	663.4
Design Mission Fuel (lb/nmi)	1.285		1.664		1.686	1.009	1.385	1.622	0.000	0.000	1.040	1.508	1.540	1.327

Bold numbers indicate an active design variable bound or constraint

4.7 Concluding Remarks

This chapter introduced a new, open-source mission analysis and conceptual sizing tool—OpenConcept—written in Python and running atop of the OpenMDAO framework. OpenConcept's analytic gradients enable the use of Newton solvers and efficient gradient-based optimization. Unique among extant aircraft MDO tools, the efficient solvers and gradients allow the user to hundreds or thousands of unique design points without high-performance computing hardware.

For this case study, more than 750 individual MDO cases were solved across a wide range of battery specific energy levels and design ranges, revealing discontinuities and tipping points in the design space. Another novel contribution was the MDO trade space exploration of a hybrid-electric aircraft considering operating economics in a comprehensive way. While a few other studies have considered total operating costs in the MDO problem for *eVTOL* [164], and *cash* operating cost for fixed wing [27], to my knowledge this is the first published MDO study considering both cash operating cost and acquisition cost for a hybrid or fixed-wing electric aircraft. Most of the previously published studies on hybrid aircraft use fuel burn as an objective function, but this tendency may be overstating the economic benefit realizable through electric propulsion, especially at lower specific energies. Using operating cost as an objective function balances the weight gain due to electric propulsion with the fuel burn reduction in a more realistic way.

A third novel contribution is quantifying the effect of airframe technology on the conventional versus electric economic tipping point. A more efficient aerostructure substantially reduces the technological requirements necessary for electric propulsion to be economically favorable. While the tipping points for this airframe and architecture may not be generally applicable, using OpenConcept, other researchers and designers may rigorously and rapidly examine the tradespace for their own architecture, mission, and set of economic inputs.

This material in this chapter was previously published [1] and has been lightly edited and condensed.

CHAPTER 5

MDO of a Series Hybrid Electric Aircraft Subject to Thermal Constraints

In the previous chapter, I demonstrated conceptual-level MDO for a hybrid-electric aircraft, but thermal constraints were not yet incorporated. Multiple industry and government studies have demonstrated the need to include thermal constraints in analysis and optimization at the conceptual level [115, 131, 132, 142], but no publicly-available electric propulsion mission analysis and sizing code supports thermal analysis. In the broader literature, a few attempts at physics-based TMS modeling of electric aircraft have been made [167, 181, 182], but none of the codes have been publicly released or open-sourced. The primary purpose of this chapter is to describe an approach to thermal modeling in conceptual aircraft MDO and describe the implementation of the thermal components. I then illustrate the effect of thermal constraints on the electric aircraft design problem by repeating the King Air tradespace study with TMS design variables.

5.1 A Brief Review of Thermal Management Systems for Electric Aircraft

The thermal management system of an electric (or hybrid-electric) aircraft removes waste heat from the electronic components. Unlike conventional turbine-powered aircraft, electric aircraft

have two features that significantly increase the magnitude of the thermal management challenge. First, while turbine engines have lower efficiency, they exhaust their waste heat to the free stream and away from the aircraft. In contrast, resistive and eddy current losses in electrical components generate internal heat and require designers to provide a way to carry away the heat. Second, electrical components must be kept at fairly low temperatures to operate properly. The smaller temperature differential between the components and the outside air means that the waste heat is much more difficult to reject.

For hydrogen fuel cell aircraft, the thermal management demands can be very large. A third or more of the fuel energy delivered to the fuel cell stack must be rejected to the atmosphere [183], which is comparable in magnitude to the thermal inefficiency of a modern, high pressure ratio turbofan. Unlike a turbofan, the fuel cell rejects most of the heat through a heat exchanger system rather than through the exhaust gas stream [183]. Compounding the problem, fuel cell stack operating temperatures are between 60-80°C, reducing the heat transfer driving force and necessitating large, efficient heat exchanger surfaces.

5.1.1 Thermal Management Architectures

There are two general design approaches to aircraft thermal management systems: direct air cooling and liquid cooling. The air-cooled approach uses heat sinks to enhance convection from each electrical component to freestream air. The X-57 Maxwell demonstrator uses this approach [131, 167, 184]. An advantage of this approach is system simplicity and reliability. A major disadvantage is that each electrical component requires direct access to an air flow path, increasing configuration complexity and potentially increasing drag as well.

The liquid-cooled approach uses coolant loops to transfer heat from the electrical components throughout the aircraft to a heat exchanger that can reject the heat to the air [142]. This approach usually reduces the number of cooling air ducts. Liquid cooling can be feasible when direct air cooling is not, especially for high power density components such as high-power fuel

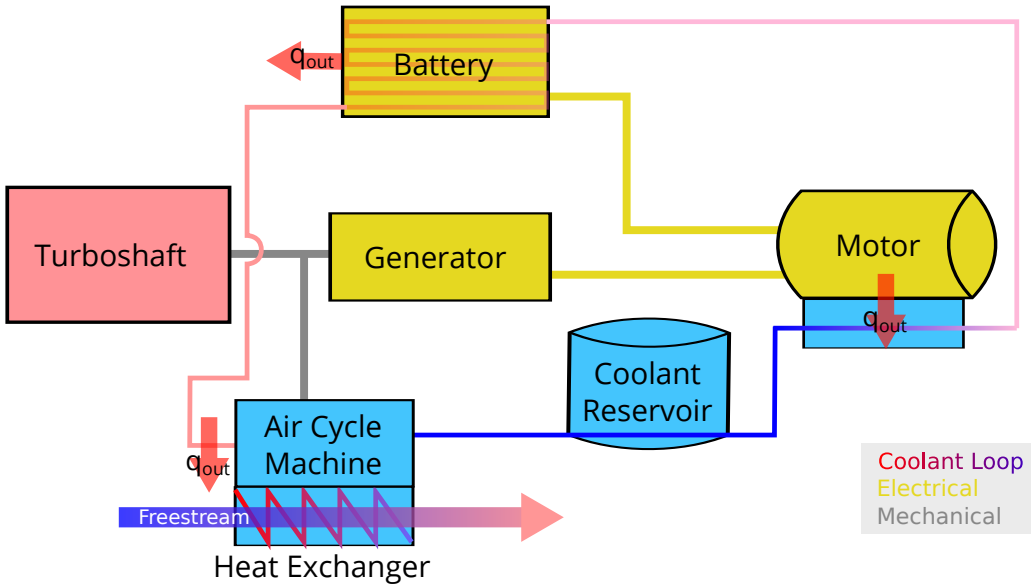


Figure 5.1: Example of liquid-cooled thermal management system architecture

cell stacks [183]. Liquid-cooled systems may also optionally use a refrigeration cycle to improve heat rejection. However, the liquid cooling architecture is arguably a more complex system design (with more failure modes and moving parts). Some aircraft may use a combination of liquid cooling and direct air cooling. A notional liquid-cooled TMS architecture is illustrated in Fig. 5.1.

The more recent industry-funded electric aircraft demonstrators have tended to be liquid-cooled. The Pipistrel Velis Electro uses a liquid-cooled battery and motor TMS. MagniX's Magni250 and Magni500 motors (used in that company's Cessna Caravan and DeHavilland Beaver demonstrators) also use liquid cooling. The more recent Siemens (now Rolls-Royce) motors have also been liquid-cooled, including the SP200D used on the CityAirbus and the SP260D used on the Extra 330LE demonstrator. Finally, the European Hy4 fuel cell demonstrator uses a liquid-cooled architecture, including a prominent ducted heat exchanger reminiscent of the P-51. Liquid cooling is used in electric ground vehicles as well, such as the Tesla Model 3 (for the motors and battery).

5.1.2 Modeling and Simulation Studies of Electric Aircraft TMS

Detailed NASA design study results have been published for the X-57. Clarke et al. [49] describe wire design trade studies including resistive heating considerations. Schnulo et al. [167] describe design and analysis of a flow-through air-cooled motor and inverter. Falck et al. [131] describe trajectory optimization subject to thermal constraints; the study found that X-57's air-cooled motors reach temperature limits and constrain the climb rate. The heat exchanger area can be reduced significantly if climb rate requirements are relaxed. Edwards and Smith [184] designed and tested a direct air-cooled sink for the X-57's low-power high lift motors and power electronics that conforms to the outer nacelle line (omitting fins).

Several NASA design studies for larger aircraft included thermal management considerations for non-superconducting architectures. Two conceptual weight estimates have been published. Jansen et al. estimate that the X-57's TMS will be 5% of the overall electronic weight (including the battery) [185]. The latest STARC-ABL assessment includes TMS weight equal to about 6% of the weight of the power electronics (not including a battery) [52].

In industry, ESAero has published most extensively on the topic of thermal management of conventional electrical machines. Freeman et al. describe the general EP thermal management design problem, analysis methods, and solutions [115]. These methods were used to design and analyze the ECO-150R [64]. The ECO-150R produces nearly 1.5 MW of waste heat at the critical top-of-climb condition. The authors describe the design and analysis of a recirculating liquid cooling system with a ram-air radiator. The radiator is designed to use heated air to generate some useful thrust to offset the radiator drag. This strategy has been referred to as the Meredith effect (after a designer of the North American Aviation P-51 Mustang, the first aircraft to demonstrate the phenomenon). Including the Meredith effect, the direct cooling system contribution to drag was around 2%–3% at cruise, and the total cooling system was 20% of the weight of all the power electronics and motors.

United Technologies created a parallel hybrid GTF engine concept, including a sized liquid

cooling system [142]. The critical condition was on a hot day prior to takeoff; a fan was required to pull cooling air through the radiator duct until sufficient ram air became available in flight. The authors concluded that once weight and drag increases from the TMS were included, the concept was not competitive with a conventional GTF. The paper includes a design sensitivity of TMS weight with maximum battery temperature; heat-sensitive batteries require more cooling power and weight. A Rolls-Royce/Georgia Tech study of an parallel hybrid engine similarly identified the challenge of cooling the batteries, and that the TMS was most challenged prior to takeoff [132]. Vratny et al. [135] present analytic equations for conceptual design of an electric aircraft TMS, including a rough consideration of liquid coolant properties (density, viscosity, and specific heat capacity).

Unlike conventional electronics, superconductors require extremely low operating temperatures — 20 K to 60 K. In exchange, Joule heating and resulting diffuse waste heat from a conventional electrical system is virtually eliminated. However, a National Academy consensus report [11] summarizes technical challenges with cryogenic power systems, including cryocooler weight, inability to handle transient loads, and difficulty with voltage regulation; thus, the committee does not expect that "cryogenic power generation or power distribution will be ready for incorporation in an aircraft propulsion system within [a] 30-year time frame."

At least three more modeling and simulation projects emerged following the publication [186] of the major results in this chapter. The work of Chapman et al. [187] developed independently and (coincidentally) uses a very similar technical approach to this chapter for modeling ducted heat exchangers — like the earlier work by Schnulo et al [167], the codebase for this project is not publicly available. Bell and Litt [188] developed thermal models of similar fidelity in the MATLAB/Simulink ecosystem; these should be useful for developing thermal control laws and less useful for airplane-level MDO because efficient and accurate derivatives are not available. Byahut and Uranga [189] modeled the propulsion system of an all-electric Twin Otter aircraft considering thermal management weight, including the effect of system voltage and insulator thickness on the

wiring. However, this project did not consider unsteady thermal profiles and sized the TMS for the climb condition only.

Thus, even in 2021, there remains a need for an open-source modeling and simulation toolkit suitable for optimization and considering unsteady thermal states. The remaining sections of this chapter detail the development, implementation, and validation of this toolkit.

5.2 Thermal Management Components and Models

5.2.1 Component Temperatures

The electrical component models detailed in Section 4 (motor, generator, battery) have a `heat_out` output variable that computes the heat generation rate of the component at the given operating point. Translating these heat outputs into component temperatures can be done assuming quasi-steadiness, or including the effect of thermal mass in a fully-unsteady fashion.

The quasi-steady formulation relies on OpenMDAO's Newton solver to compute component temperatures. The implicit problem is:

$$\text{compute} \quad T_{\text{comp}} \tag{5.1}$$

$$\text{such that} \quad \mathcal{R}(T_{\text{comp}}) = q_{\text{comp}} - q_{\text{out}} = 0 \tag{5.2}$$

where T_{comp} is the component temperature, q_{comp} is the heat generation rate of the electrical component, and q_{out} is the instantaneous heat rejection rate due to cooling that component. The heat rejection rate is computed as a function of the component temperature and a number of other heat transfer parameters (introduced in Section 5.2.2) like so:

$$q_{\text{out}} = q(T_{\text{comp}}, \ldots) \tag{5.3}$$

Equations 5.1 and 5.2 depend on each other via the cooling system model (Equation 5.3), creating a circular dependency that must be solved using the OpenMDAO Newton solver.

The quasi-steady formulation becomes less accurate as the thermal mass increases. Even lightweight aerospace-grade electrical components have a significant thermal mass, and at low-speed conditions (such as the beginning of the takeoff roll), neglecting thermal mass is likely to result in unrealistic high temperatures and drive oversized TMS designs. Therefore, it is preferable to use a time-accurate model, which can be expressed as

$$\frac{dT_{\text{comp}}}{dt} = \frac{q_{\text{comp}} - q(T_{\text{comp}}, ...)}{m_{\text{comp}} c_p} \tag{5.4}$$

$$T_{\text{comp}} = \int_{t_0}^{t_f} \frac{dT_{\text{comp}}}{dt}\, dt \tag{5.5}$$

Equations (5.4) and (5.5) depend on each other and hence form an implicit cycle similar to the quasi-steady formulation.

The rate $\frac{dT_{\text{comp}}}{dt}$ is computed by the `ThermalComponentWithMass` component. A numerical scheme is required to compute the time integral in Equation (5.5), and I use a fourth-order Simpson's rule approximation (described in more detail in Chapter 4), solved implicitly in vectorized form all-at-once using the Newton solver (without time marching). This means that the time integration and the implicit ODEs are solved simultaneously as one coupled nonlinear system. The user must specify an initial component temperature, usually based on ambient conditions. Unlike the quasi-steady problem, the accuracy of the temperature profile in unsteady mode depends on the time step chosen. A smaller time step increases the size of the OpenMDAO implicit problem that needs to be solved and increases the computation time.

5.2.2 Component-Fluid Heat Transfer

Computing q_{out} from each component, which represents the convective heat transfer rate from the component to a fluid stream, requires physics-based models based on assumptions about the component design. For a liquid-cooled component, the fluid stream is a coolant like propylene glycol, whereas for an air-cooled component the fluid stream comes from freestream air. In nearly every case, designers use enhanced heat transfer surfaces, such as microchannels or finned heat sinks. The `ConstantSurfaceTemperatureColdPlate_NTU` component implements a microchannel cold plate and is a reasonable choice for liquid-cooled and air-cooled applications. I assume that the thermal conductivity of the electrical component is large relative to the cooling fluid resulting in a constant channel surface temperature in the streamwise direction. I further assume that the aspect ratio of each channel is large and thus approximates the local heat transfer properties using the theoretical result for infinite parallel plates. The convective heat transfer coefficient can be computed as

$$h_{\text{conv}} = \frac{\text{Nu}\, k}{d_h}, \tag{5.6}$$

where Nu is the Nusselt number (which is set to 7.54 by default for constant temperature infinite parallel plates [190]), k is the thermal conductivity of the fluid, and d_h is the hydraulic diameter of the channel. For a high aspect ratio channel,

$$d_h = \frac{2\mathscr{W}\mathscr{H}}{\mathscr{W} + \mathscr{H}} \tag{5.7}$$

where $\mathscr{W}$ is the fluid channel width and $\mathscr{H}$ is the fluid channel height. I neglect entrance effects for this high aspect ratio microchannel. For air cooled applications using finned heat sinks, the user may wish to modify the heat transfer coefficient to account for fin efficiency. To compute the

overall heat transfer, the heat transfer surface area is obtained from:

$$A = 2\ell(\mathscr{W} + \mathscr{H})n_{\text{parallel}}, \tag{5.8}$$

where A is the overall heat transfer surface area, ℓ is the length of the microchannel in the fluid flow direction, and n_{parallel} is the total number of individual microchannels.

Given these convective properties, I compute the actual heat transfer using the NTU-effectiveness method [190], which is typically used for fluid-fluid heat exchangers where both fluids change temperature during the exchange. For a metal component with high thermal conductivity, the heat transfer capability of the solid is effectively "infinite" for the purposes of the NTU-effectiveness method. Therefore, the heat transfer capacity of the cold plate is governed solely by the coolant material properties and flow rate. The heat transfer capacity is computed as:

$$\mathcal{C}_{\text{min}} = \dot{m}_{\text{coolant}}\, c_{p,\text{coolant}}, \tag{5.9}$$

where $\dot{m}_{\text{coolant}}$ is the coolant mass flow rate through the entire cold plate (not just a single channel) and $c_{p,\text{coolant}}$ is the coolant's specific heat capacity. The number of thermal units (NTU) is computed as

$$\text{NTU} = \frac{Ah_{\text{conv}}}{\mathcal{C}_{\text{min}}}, \tag{5.10}$$

and heat transfer effectiveness is

$$\epsilon = 1 - e^{-\text{NTU}} \tag{5.11}$$

Finally, the heat transfer as

$$q_{\text{out}} = \epsilon\mathcal{C}_{\text{min}}(T_{\text{comp}} - T_{\text{coolant,in}}), \tag{5.12}$$

and the coolant outlet temperature as

$$T_{\text{coolant,out}} = T_{\text{coolant,in}} + \frac{q_{\text{out}}}{\mathcal{C}_{\text{min}}}. \tag{5.13}$$

The user is responsible for setting reasonable values for channel geometry (ℓ, w, $\mathscr{H}$, n_{parallel}) so that the channel flow is laminar and the infinite parallel plate assumption remains reasonable, and for ensuring that the component has sufficient material volume to accommodate the cooling channels. This analysis also assumes that the cooling channel weight is accounted for in the all-up weight of the component, which may not be the case for air-cooled external heat sinks.

5.2.3 Fluid-fluid Heat Transfer

After heat from electrical components is transferred into the liquid coolant loop via the cold plate, the heat must be rejected to the atmosphere. A reasonable choice for accomplishing this is a ducted compact heat exchanger. Like the cold plate component in the previous subsection, I use the NTU-effectiveness method to compute the heat transfer rate,

$$q = \epsilon \, \frac{UA_{\text{overall}}}{\text{NTU}} \Big(T_{\text{in,h}} - T_{\text{in,c}} \Big), \tag{5.14}$$

where UA_{overall} is the overall heat transfer coefficient times the corresponding heat transfer area, $T_{\text{in},h}$ $T_{\text{in},c}$ are the fluid inlet temperatures, the number of thermal units is computed as

$$\text{NTU} = \frac{UA_{\text{overall}}}{\mathcal{C}_{\text{min}}}, \tag{5.15}$$

and the heat transfer effectiveness is

$$\epsilon = \Phi \left(\text{NTU}, \, \frac{\mathcal{C}_{\text{min}}}{\mathcal{C}_{\text{max}}} \right), \tag{5.16}$$

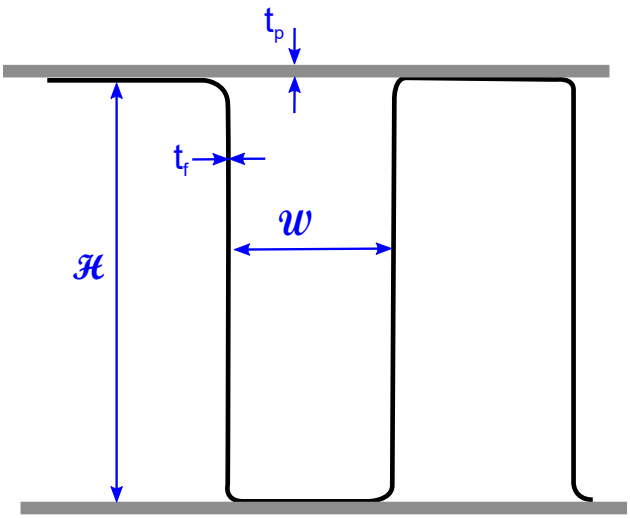


Figure 5.2: Cross-sectional geometry of the offset strip fin heat exchanger [192]

where $\mathcal{C}_{\text{min}}$, $\mathcal{C}_{\text{max}}$ are the maximum and minimum values of the fluid heat transfer capacity $\dot{m}\, c_p$ for the hot and cold sides, and Φ is an analytical or empirical function that depends on the flow arrangement of the heat exchanger (for example, crossflow) [191].

For this study, I use crossflow plate-fin heat exchangers with offset strip fin geometry as described in Jasa et al [192]. Offset strip fin heat exchangers are considered "compact" heat exchangers with high heat transfer to surface area rates [191]. The geometric design of a heat exchanger varies to satisfy heat transfer, pressure loss, weight, and volume requirements. Figure 5.2 illustrates a cross section of offset strip fin channels along with a commonly-used geometric parameterization.

I use an empirical relation from [193] to compute heat transfer and pressure loss specific to the offset strip fin configuration. By default, OpenConcept's `HXGroup` component uses geometric parameters representative of a air-liquid heat exchanger, with cold-side channel width and height 1 mm, and hot-side channel width 14 mm by 1.35 mm.

5.2.4 Fluid Reservoir

Liquid cooling systems require a reservoir to account for thermal expansion and maintenance. The thermal mass of the fluid in the reservoir may affect peak temperatures. If the fluid is perfectly mixed within the reservoir (that is, fluid entering the reservoir is instantaneously mixed with the existing fluid), the rate of change of temperature within the reservoir can be computed using

$$\frac{dT_{\text{reservoir}}}{dt} = \frac{\dot{m}_{\text{coolant}}}{m_{\text{coolant}}}(T_{\text{in}} - T_{\text{reservoir}}), \tag{5.17}$$

where $T_{\text{reservoir}}$ is the reservoir (and reservoir outlet) temperature, $\dot{m}_{\text{coolant}}$ is the coolant mass flow rate, m_{coolant} is the mass of coolant in the reservoir, and T_{in} is the reservoir inflow temperature. Quasi-steady thermal analysis cannot model the effect of a fluid reservoir, which is purely a thermal mass effect. When $\dot{m}/m$ is large, the time constant associated with the reservoir temperature becomes small. As m tends to zero, the unsteady solution approaches the quasi-steady solution. A small time constant makes the thermal ODE very stiff and introduces numerical difficulties in the overall time integration problem, so unless the reservoir is large and flow rates are small, the reservoir can be neglected in the thermal analysis (though the mass of the coolant should be considered in the TMS total mass).

5.2.5 Coolant Duct

Ducted radiators greatly reduce cooling drag compared to finned heat sinks in the freestream [194, 195]. There are two primary mechanisms for this. First, a duct that decelerates flow prior to encountering the heat exchanger element generally undergoes a lower total pressure loss. Second, the combination of duct and heat exchanger can act as a weak ramjet providing a further modest offset to the drag of the whole arrangement. For aircraft with high-temperature cooling loads flying at relatively high speeds, a large portion of the drag can be offset (or potentially, some positive

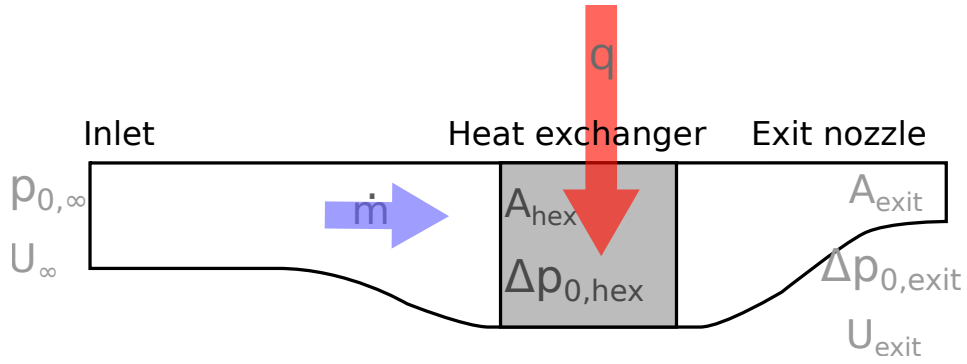


Figure 5.3: Ducted heat exchanger to reduce cooling drag.

thrust could be generated). The most famous application of this weak ramjet concept (known as the Meredith effect) is the North American P-51 Mustang's liquid engine cooling system [194].

I developed two approaches for computing cooling drag due to ducted heat exchangers. The first option is an incompressible approximation. Adapting Theodorsen's model of radial engine cylinder cooling drag from [195], I model a duct with a frontal opening, diffuser, heat exchanger, and nozzle (Fig. 5.3). The fluid density everywhere in the duct is assumed to be ρ_∞. Let A_{hex} be the free flow passage area of the heat exchanger, and A_{exit} be the exit nozzle area. Let $\Delta p_{0,\text{hex}} = f(\dot{m})$ be the pressure loss across the heat exchanger as a function of the duct mass flow rate $\dot{m}$. Let ξ_{exit} be a static pressure loss as a function of nozzle dynamic pressure, that is $\Delta p_{0,exit} = \xi_{\text{exit}} q_{\text{exit}}$. I assume that the nozzle expands the flow back to the freestream static pressure p_∞, though this assumption would not hold in all cases. The total pressure at the exit is then computed as:

$$p_{0,exit} = p_{0,\infty} - \Delta p_{0,\text{hex}} - \Delta p_{0,exit} = p_\infty + \frac{1}{2}\rho U_\infty^2 - \Delta p_{0,\text{hex}} - \Delta p_{0,exit} = p_{\text{exit}} + \frac{1}{2}\rho U_e^2 \quad (5.18)$$

Substituting $\Delta p_{0,exit} = \xi_{\text{exit}} \frac{1}{2}\rho U_{\text{exit}}^2$ and rearranging, I obtain:

$$U_{\text{exit}} = \sqrt{\frac{U_\infty^2 - \frac{2}{\rho}\left((p_{\text{exit}} - p_\infty) + \Delta p_{\text{hex}}\right)}{1 + \xi_{\text{exit}}}}, \quad (5.19)$$

By continuity:

$$\dot{m} = A_{\text{exit}} \rho U_{\text{exit}}. \tag{5.20}$$

Net force is computed by balancing the change in fluid momentum ($\dot{m}\Delta U$) and pressure forces. To account for inlet, duct, and nozzle losses not otherwise accounted for, I apply a factor ($C_{fg} = 0.98$) to gross thrust in the drag computation and obtain:

$$F_{\text{net}} = \dot{m}(U_{\text{exit}} C_{fg} - U_\infty) + A_{\text{exit}} C_{fg}(p_{\text{exit}} - p_\infty). \tag{5.21}$$

Alternatively, a more sophisticated 1D thermodynamic cycle modeling approach can be used to compute drag. Isentropic relations are used to solve for Mach numbers and flow properties implicitly using OpenMDAO's Newton solver. The compressible model captures Mach number and heat addition effects on net cooling drag. However, the additional fidelity is likely not meaningful for low-speed general aviation airplanes with small thermal management loads, and the compressible relations introduce many implicit states and some robustness issues to the overall MDO problem.

5.3 Case Study: Revisiting the Series Hybrid Twin

To exercise the TMS model and assess the impact of thermal constraints on the design space, I revisit the MDO study of the series hybrid twin turboprop from Chapter 4. The baseline aircraft is a Beechcraft King Air C90GT with a drop-in replacement series-hybrid propulsion system replacing the turboprop engines.

The series-hybrid electric propulsion architecture including TMS is illustrated in Fig. 5.4. To enable the aircraft to continue safe flight and landing after loss of any single component on takeoff, the propulsion system uses two electric motors, two propellers, and a battery large enough to provide full takeoff power in the event of engine loss. Specific power, efficiency, and cost assumptions for individual powertrain components are listed in Table 5.1 and remain unchanged from

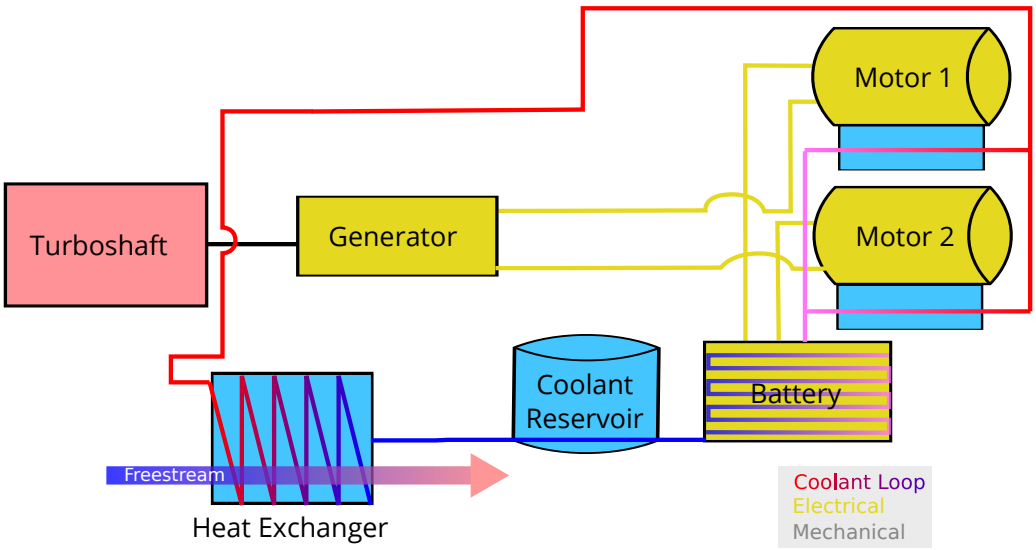


Figure 5.4: Systems architecture for the twin series hybrid case study.

Chapter 4. The airplane structural weight and drag characteristics, and the mission analysis methods, also remain unchanged, except that OpenConcept now integrates many more ODE states (the component temperatures).

Table 5.1: Powertrain technology assumptions [1]

Component	Specific Power (kW/kg)	Efficiency	PSFC (lb/hp/hr)
Battery	5.0	97%	–
Motor	5.0	97%	–
Generator	5.0	97%	–
Turboshaft	7.15[1]	–	0.6

As the revised study with thermal constraints followed some major updates to the OpenConcept code base, I began by re-running the series hybrid twin tradespace exploration from Chapter 4. The grid of MDO problems was formulated as follows:

[1]Not including 104 kg base weight

$$
\begin{aligned}
\text{minimize: } & \text{fuel burn} + 0.01\text{MTOW} \\
\text{by varying:} & \\
& \text{MTOW} \\
& S_{\text{ref}} \\
& d_{\text{prop}} \\
& m_b \\
& P_{m,\text{rated}} \\
& P_{\text{ts,rated}} \\
& P_{g,\text{rated}} \\
& H_E \text{ (degree of hybridization w.r.t energy)} \\
\text{subject to scalar constraints:} & \\
& \mathcal{R}_{\text{TOW}} = m_{\text{TO}} - m_f - m_{\text{empty}} - m_{\text{payload}} - m_b \geq 0 \\
& \mathcal{R}_b = E_{\text{b,max}} - E_{\text{b,used}} \geq 0 \\
& \mathcal{R}_{\text{vol}} = m_{f,\text{max}} - m_f \geq 0 \\
& \text{BFL} \leq 4452 \text{ ft (no worse than baseline)} \\
& \text{engine out climb gradient} \geq 2\% \\
& V_{\text{stall}} \leq 81.6 \text{ kts (no worse than baseline)} \\
\text{and vector constraints:} & \\
& \boldsymbol{P}_m \leq 1.05 P_{m,\text{rated}} \\
& \boldsymbol{P}_{\text{ts}} \leq P_{\text{ts,rated}} \\
& \boldsymbol{P}_g \leq P_{g,\text{rated}} \\
& \boldsymbol{P}_b \leq m_b \cdot p_b
\end{aligned}
$$

The results were similar to the previous study despite significant changes to the underlying mission analysis methods.

5.3.1 Optimization with Thermal Constraints

I modified the aircraft propulsion model to include thermal management of the motor and battery. I added a `ThermalComponentWithMass` to the motor (lumping both motors together) and to the battery pack. Thermal mass of the battery and the motor was computed using a specific heat of 921 J/kg/K (representative of aerospace-grade aluminum). I connected cold plates for both components in series using a liquid cooling system using a propylene glycol and water mixture

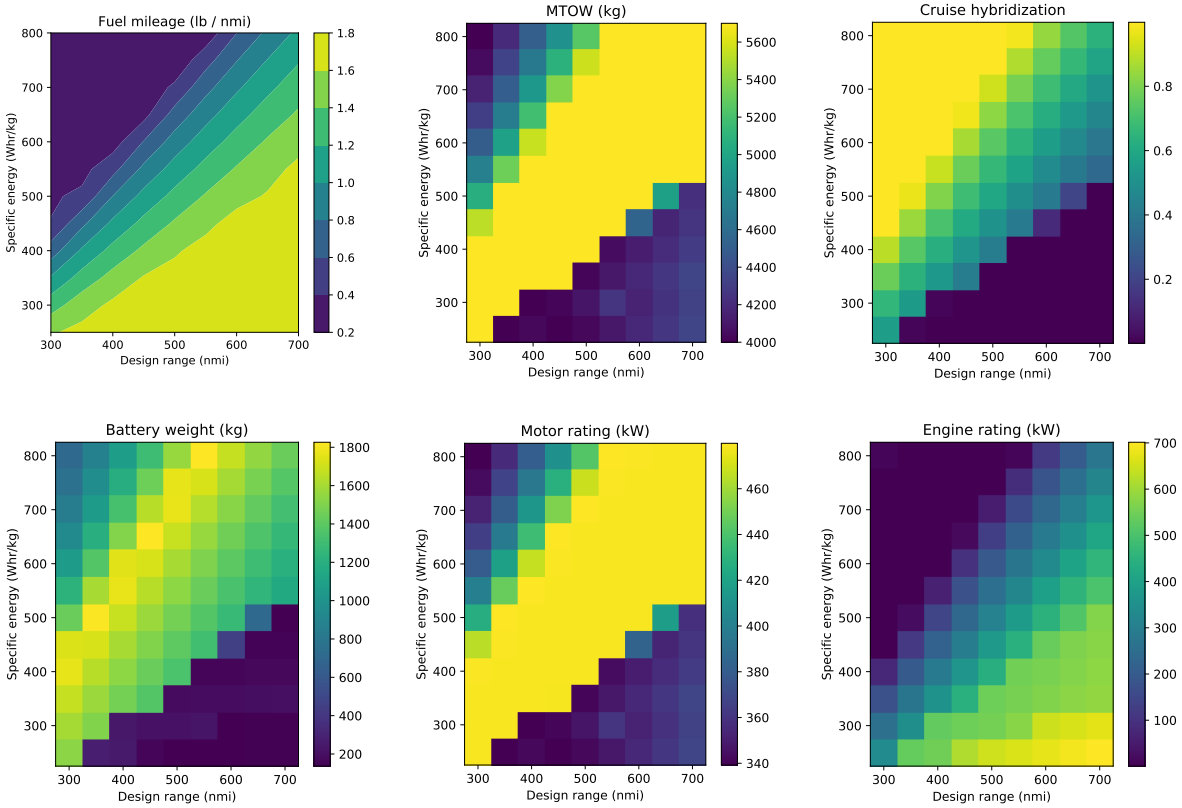


Figure 5.5: Minimum fuel burn MDO results without thermal constraints.

with a specific heat of 3801 J/kg/K [196]. The coolant loop rejects heat via a ducted heat exchanger. As this is a low-speed aircraft with modest thermal loads, I neglected the drag-offsetting effect of heat addition and used the simpler incompressible duct to model the air mass flow and drag. I set OpenConcept's default geometric parameters for the offset strip fin heat exchanger to reasonable numbers based on measurements from several automotive radiators and used those parameters for this study as well. Finally, I included a liquid coolant reservoir upstream of the heat exchanger. I include the weight of the coolant and heat exchanger in the empty weight of the airplane, and include the drag contribution of the duct and heat exchanger with the overall aircraft drag when computing the steady flight throttle setting. Figure 5.6 shows profiles of mission parameters for a single point in the grid of optimization runs (at 250 Wh/kg and 400 nmi range). The figure highlights the importance of time-accurate thermal analysis. During takeoff and low-altitude climb, heating is at its maximum and convective heat transfer capability is at a minimum (due to higher atmospheric temperature and lower coolant duct mass flow). A quasi-steady thermal analysis would predict very high temperatures during this part of the mission. However, because the thermal components have considerable thermal mass, the maximum temperature is not reached until the top of the climb phase. Sizing the thermal management system to a quasi-steady analysis at the most critical condition (early in the takeoff roll) would result in an oversized heat exchanger and unnecessarily high drag and weight penalty.

I also added several design variables and constraints to the previous problem. The optimizer sizes the heat exchanger width and the area of the duct nozzle, thus allowing it to trade off weight and drag (a larger, heavier heat exchanger will have less pressure drop for a given heat transfer rate). The optimizer also sizes the coolant reservoir. I constrain the time-accurate temperatures of the motor and battery pack to stay within operating limits (90° C for the motor and 50° C for the battery). The full MDO problem is as follows:

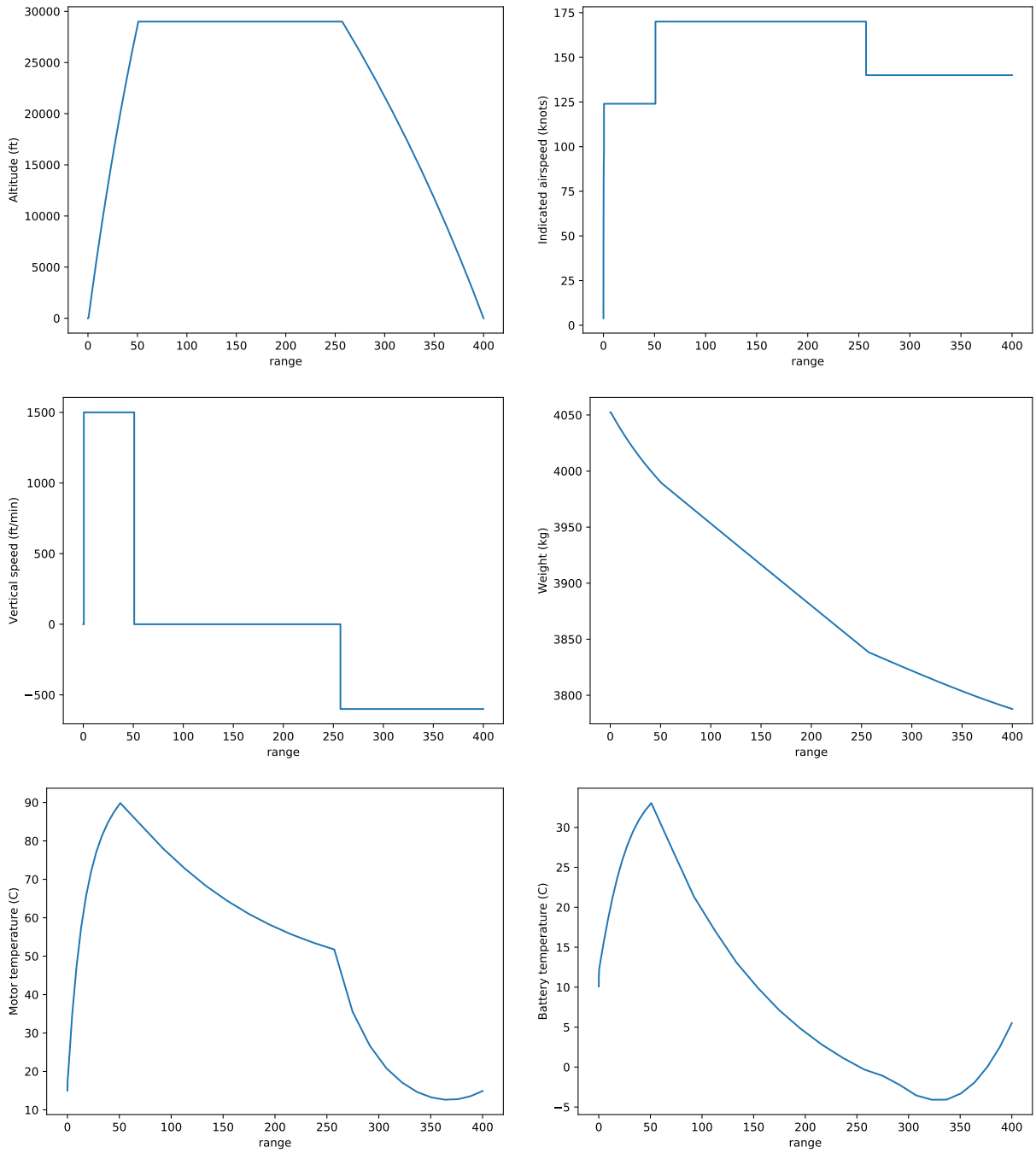


Figure 5.6: Mission trajectories for a 400 nmi mission ($e_b = 250$)

minimize: fuel burn $+ 0.01MTOW$

by varying:

MTOW

S_{ref}

d_{prop}

m_b

$P_{m,\text{rated}}$

$P_{\text{ts,rated}}$

$P_{g,\text{rated}}$

H_E (degree of hybridization w.r.t energy)

A_{nozzle} (cooling duct outlet cross-sectional area)

n_{wide} (number of heat exchanger cells wide)

m_{coolant} (coolant reservoir mass)

subject to scalar constraints:

$$\mathcal{R}_{\text{TOW}} = m_{\text{TO}} - m_f - m_{\text{empty}} - m_{\text{payload}} - m_b \geq 0$$

$$\mathcal{R}_b = E_{\text{b,max}} - E_{\text{b,used}} \geq 0$$

$$\mathcal{R}_{\text{vol}} = m_{f,\text{max}} - m_f \geq 0$$

$$\text{BFL} \leq 4452\text{ft} \quad \text{(no worse than baseline)}$$

$$\text{engine out climb gradient} \geq 2\%$$

$$V_{\text{stall}} \leq 81.6\text{kt} \quad \text{(no worse than baseline)}$$

and vector constraints:

$$\boldsymbol{P}_m \leq 1.05 P_{m,\text{rated}}$$

$$\boldsymbol{P}_{\text{ts}} \leq P_{\text{ts,rated}}$$

$$\boldsymbol{P}_g \leq P_{g,\text{rated}}$$

$$\boldsymbol{P}_b \leq m_b \cdot p_b$$

$$\boldsymbol{T}_m \leq 90^\circ C$$

$$\boldsymbol{T}_b \leq 50^\circ C$$

Figure 5.7 shows the design variables and selected responses at the optimal points across the trade space. The motor temperature constraint is always active at the top of climb for all the designs in the tradespace (and so is not shown in Fig. 5.7). The optimizer varies the duct nozzle area (to vary cooling air mass flow) and motor size (to add thermal mass) such that the motor temperature reaches the limit at the top of the climb. The heat exchanger width converges to its upper bound at virtually every point in the design space, while coolant mass converges to its lower bound at every

point.

Figure 5.8 shows the difference in key variables (including fuel mileage) after accounting for thermal design and thermal constraints. While fuel mileage worsened at every point in the design space, the impact was much larger on certain combinations of specific energy and design range. At long range and low battery specific energy, and at short range and high specific energy, there was little effect. The long range design with low e_b are essentially turboelectric and benefit from light weight and low battery waste heat; there is simply less overall heat to reject, thus minimizing the associated penalty. The short range designs with high e_b use no fuel to begin with, so their fuel burn remains at zero even as they use more energy; instead, the thermal management penalty is observed as an increase in MTOW. Between these two extreme designs, the heavy hybrid airplanes generate a large amount of waste heat and burn appreciable fuel, making the impact of thermal constraints more significant.

A very interesting trend emerged in the motor sizing design variable. The optimizer greatly oversized the motors in a band in the heart of the tradespace (seen as a band of red from middle left to top right in Figure 5.7). In the rest of the tradespace, the motor is sized by power required during climb. However, in the red band, the motor is being constrained by the thermal problem. I suspect that this is a result of the sequencing of components in the thermal management system. I designed the TMS architecture to provide the coldest coolant to the battery, since it has a lower operating temperature. The consequence is that warmer coolant flows into the motor. The motor inflow temperature varies slowly even as outside temperature drops due to the thermal mass of the battery. The best solution available to the optimizer is to oversize the motors to add thermal mass and avoid overheating at the critical top of climb point. Reordering the components could result in an improvement in fuel burn in this part of the tradespace by better balancing peak temperatures between the motor and battery.

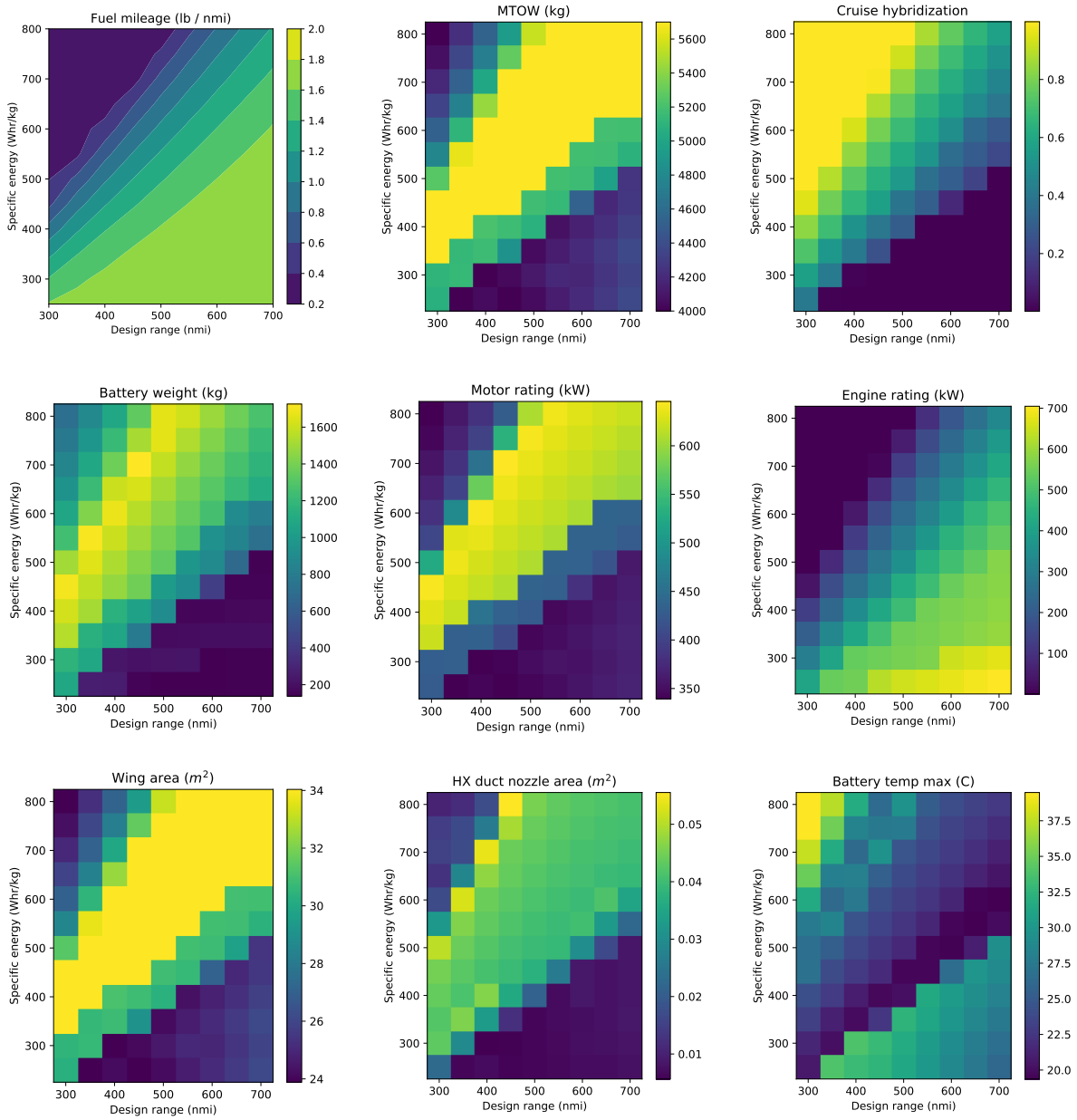


Figure 5.7: Minimum fuel burn MDO results with thermal constraints.

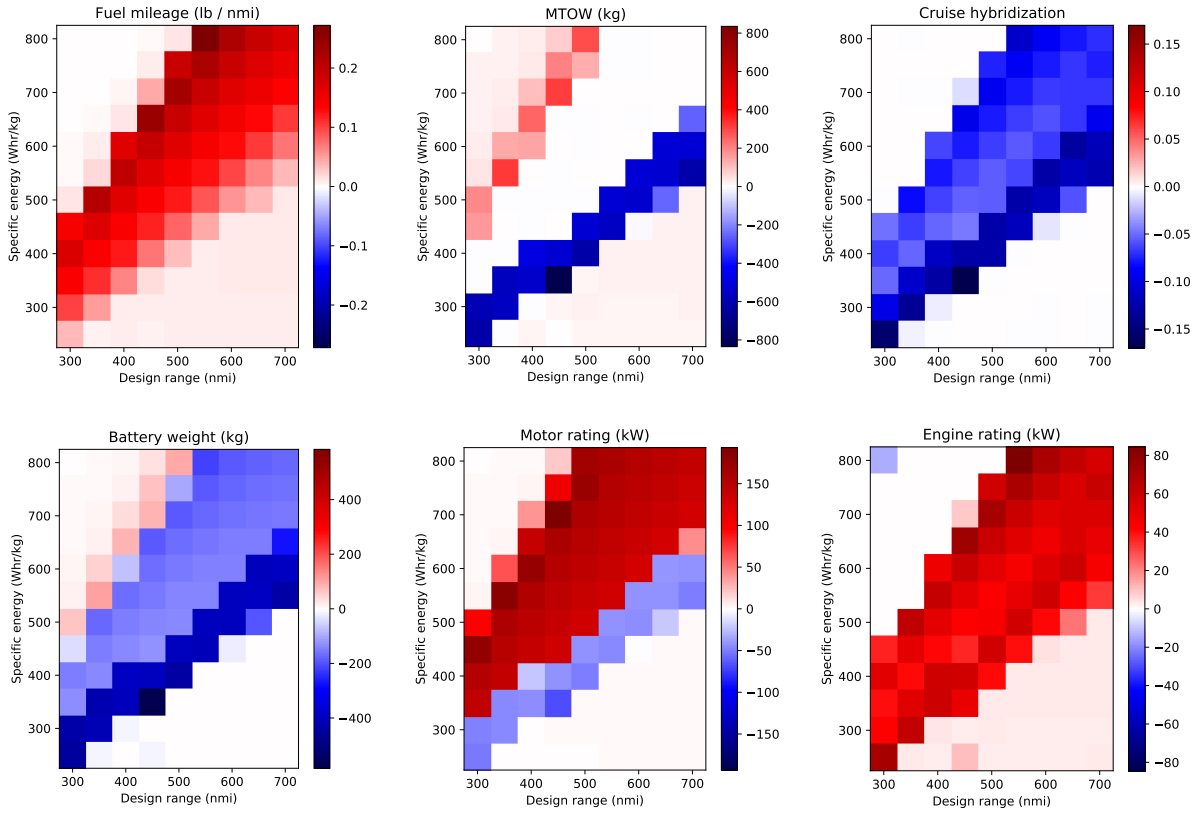


Figure 5.8: Difference in optimal designs due to thermal constraints (positive = thermally-constrained higher)

5.4 Concluding Remarks

Thermal constraints are understudied compared to other disciplines in aircraft conceptual design, and there are few publicly-available resources available for the research community to incorporate thermal constraints into electric aircraft studies. To fill this gap, I introduced thermal analysis and design capabilities to the OpenConcept Python package. I demonstrated that thermal mass effects are significant when analyzing aircraft thermal trajectories, particularly early in the mission when power is high and speeds and altitudes are low. Therefore, quasi-steady thermal models are not sufficient for the design of aircraft thermal management systems, because they can lead to dramatic over-sizing. I incorporated time-accurate thermal models into the mission analysis and used them to formulate constraints in the aircraft design optimization problem. The time-accurate thermal analyses and derivatives were computed by the OpenConcept package to enable efficient gradient-based design optimization.

I showed that thermal constraints appreciably affect the fuel burn and energy usage achievable in a series hybrid architecture, but not uniformly throughout the tradespace. The non-uniform effects make the impact of thermal constraints on aircraft design somewhat non-intuitive and underscore the importance of including them early in the design process. Electric aircraft architectures with a large percentage of battery power will be impacted by TMS penalties, but because they burn little or no fuel, the penalty is seen as an MTOW and total energy increase, not a fuel burn penalty. Conversely, turboelectric aircraft experience a modest TMS penalty due to lighter weight and lack of battery heating. Hybrid-electric aircraft see the largest fuel burn penalty since they are heavier than turboelectric aircraft (thus producing more motor waste heat) and use significant quantities of batteries (producing yet more waste heat). I also observed that the optimizer can find creative ways to satisfy the thermal constraints (such as oversizing a motor to add thermal mass and avoid a transient over-temperature condition).

The material in this chapter was previously published [186] and has been edited and expanded.

CHAPTER 6

Improved Thermal Subsystem Models for Electric Aircraft

The previous chapter described an initial set of basic thermal management models sufficient to solve the MDO problem for a small airplane, as well as the numerical approach to solving the model equations in OpenMDAO. This chapter introduces a much broader set of models designed to capture more airplane-level effects of the thermal management subsystem. These models are necessary in order to perform the MDO trade study for a more complex parallel hybrid transport aircraft described in Chapter 7. I also validate the a subset of the thermal models by replicating temperature profiles from two experimental test flights of the Pipistrel Velis Electro (the first electric aircraft to earn a type certificate from EASA or the FAA).

6.1 Improved Semi-Empirical Heat Sink Models

While the generic cold plate described in Section 5.2.2 describes one valid means of extracting waste heat from an electrical component, it does not represent the state of the art in battery cooling architectures. It also does not automatically scale with the sizing of the electrical component per current engineering practice. The purpose of this section is to describe new battery and electric machine heat sink models developed based on current best practices.

6.1.1 Battery Heat Sink Modeling

Arguably the state of the art in thermal management of large batteries is the Tesla Model 3. The Model 3 battery pack consists of hundreds of 21mm by 70mm lithium ion cells arranged in a rectangular array. The pack is manufactured by thermally and mechanically bonding two rows of cells to each ribbon (one row of cells on either side of the ribbon), forming a "bandolier" assembly [197]. The general arrangement is shown in Figure 6.1 The bandoliers are then stacked in rows to form the pack. Coolant is fed to each ribbon in parallel from an upstream manifold, an arrangement which has been estimated to double the heat rejection capability of the Model 3 compared to the earlier Model S [198].

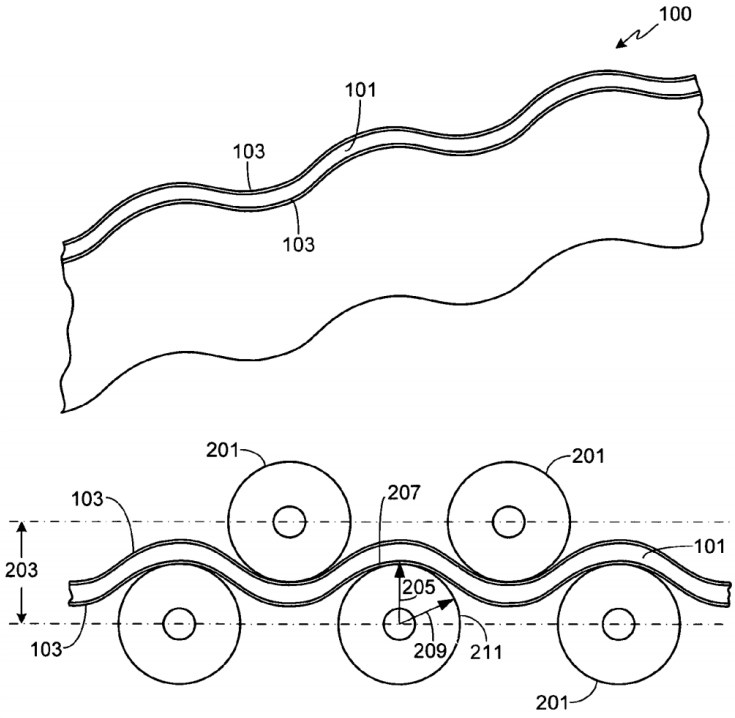


Figure 6.1: Example of a battery thermal management arrangement (from Tesla patent [199])

I developed a battery-specific heat sink model based on Tesla's design. The goal of the model is to predict the time-varying cell temperature, $\overline{T}_{\text{cell}}(t)$, given battery pack sizing parameters and coolant flow properties. First, the temperature drop from the cell volume to the cell surface must

be computed; then the convective heat transfer to the coolant must be obtained.

Convective heat transfer to the coolant in the high-aspect ratio ribbon can reasonably be modeled by considering the theoretical heat transfer due to laminar flow between infinite parallel plates. The convective heat transfer coefficient for this configuration can be computed as

$$h_{\text{conv}} = \frac{\text{Nu}\, k}{2t_{\text{ribbon}}}, \tag{6.1}$$

where Nu is the Nusselt number (7.54 for infinite parallel plates [190]), k is the thermal conductivity of the coolant, and t_{ribbon} is the narrow dimension of the coolant channel (here, I approximate the hydraulic diameter $d_h \approx 2t_{\text{ribbon}}$). The heat transfer area A in one bandolier is:

$$A = 2h_{\text{cell}}\ell_{\text{bandolier}} \tag{6.2}$$

where h_{cell} is the height of the cell and $\ell_{\text{bandolier}}$ is the length of the bandolier in the flow direction. $\ell_{\text{bandolier}}$ can be conservatively estimated as:

$$\ell_{\text{bandolier}} = n_{\text{cell}} d_{\text{cell}} \tag{6.3}$$

where n_{cell} is the number of cells long in each bandolier, and d_{cell} is the diameter of the cell. The general arrangement and relevant dimensional parameters are illustrated in Figure 6.2.

I use the NTU-effectiveness method introduced in Chapter 5 to compute the pack-level heat transfer. The number of thermal units (NTU) is computed:

$$\text{NTU} = \frac{h_{\text{conv}} A}{\dot{m} c_{p,\text{coolant}}} \tag{6.4}$$

where $\dot{m}$ is the coolant mass flow rate and $c_{p,\text{coolant}}$ is the coolant specific heat capacity. Assuming the flow direction is alternated in time to maintain a relatively uniform pack temperature, we can

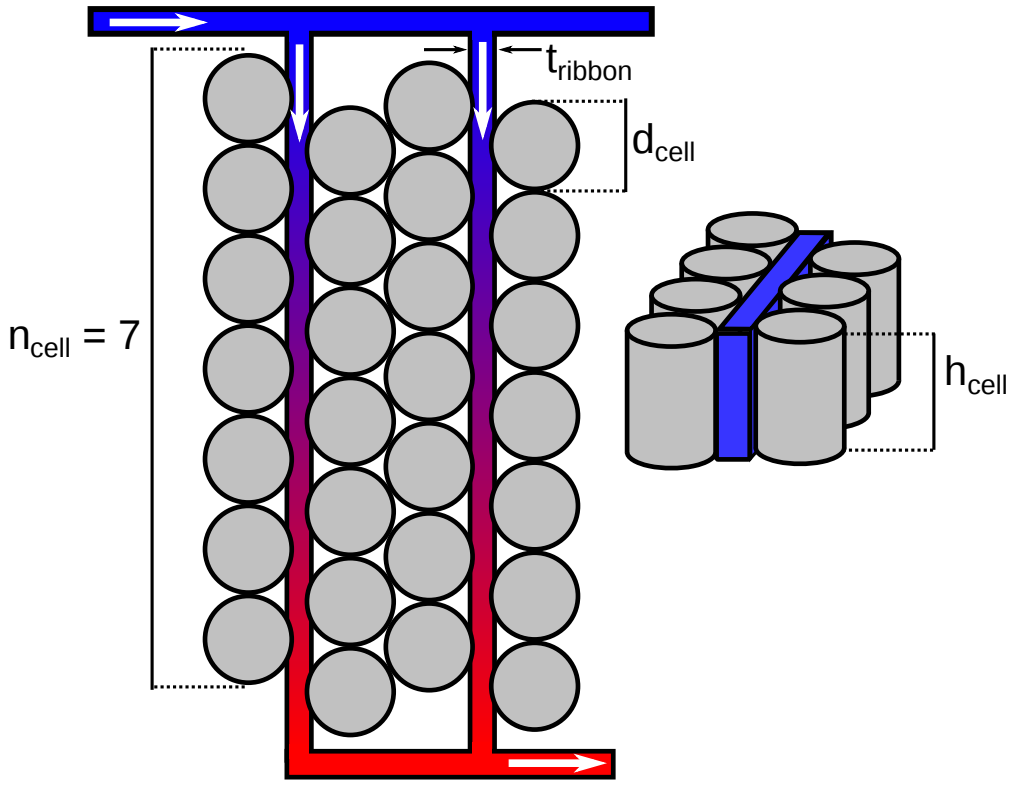


Figure 6.2: General arrangement and dimensional parameters of the battery heat sink model

use the same "infinite specific heat" assumption for the solid side of the heat transfer and compute the overall heat transfer using the expression from Chapter 5:

$$q_{\text{conv,overall}} = \dot{m} c_{p,\text{coolant}} (1 - e^{-\text{NTU}}) (T_{\text{battery,surface}} - T_{\text{coolant,in}}) \tag{6.5}$$

Averaging the heat transfer over each cell, we obtain:

$$q_{\text{conv,cell}} = \frac{q_{\text{conv,overall}}}{2n_{\text{cell}}} = \frac{\dot{m} c_{p,\text{coolant}} (1 - e^{-\text{NTU}})}{2n_{\text{cell}}} (T_{\text{cell,surface}} - T_{\text{coolant,in}}) \tag{6.6}$$

We can also pull out the thermal conductance $C_{\text{conv,cell}}$ as follows:

$$C_{\text{conv,cell}} = \frac{\dot{m} c_{p,\text{coolant}} (1 - e^{-\text{NTU}})}{2n_{\text{cell}}} \tag{6.7}$$

To compute cell surface temperature from internal cell temperature, I make a quasi-steady heat transfer assumption — that is, while the overall cell temperature $\overline{T}_{\text{cell}}(t)$ can vary with time, I assume that the radial temperature gradient in the battery instantaneously adjusts to remain consistent with the convective heat transfer. Mathematically:

$$q_{\text{conv,cell}} = q_{\text{cond,cell}} \tag{6.8}$$

where $q_{\text{cond,cell}}$ is the instantaneous radial conductive heat transfer rate from a reference point at radius R_T inside the battery to the cell surface at $R_{\text{cell}} = D_{\text{cell}}/2$. The steady-state conductive heat transfer in a hollow cylinder with inner radius R_T and outer radius R_{cell} is:

$$q_{\text{cond,cell}} = 2\pi h_{\text{cell}} k_{r,\text{cell}} \frac{\overline{T}_{\text{cell}} - T_{\text{cell,surface}}}{\ln(R_{\text{cell}}/R_T)} \tag{6.9}$$

where $k_{r,\text{cell}}$ is the material thermal conductivity of the cell in the radial direction (the material properties of cells vary immensely between the radial and axial directions) [200]. For this study, I arbitrarily chose a reference location at about 75% of the cell radius and assumed $k_{r,\text{cell}} = 0.3$ (a conservative knockdown from published estimates [200]). The radial thermal conductance of the cell is:

$$C_{\text{cond,cell}} = \frac{2\pi h_{\text{cell}} k_{r,\text{cell}}}{\ln(R_{\text{cell}}/R_T)} \tag{6.10}$$

Finally, to compute overall heat transfer as a function of $\overline{T}_{\text{cell}} - T_{\text{coolant,in}}$, I compute the series thermal conductance across conduction and convection and obtain:

$$q_{\text{cell,out}} = \frac{C_{\text{cond,cell}} C_{\text{conv,cell}}}{C_{\text{cond,cell}} + C_{\text{conv,cell}}} (\overline{T}_{\text{cell}} - T_{\text{coolant,in}}) \tag{6.11}$$

The unsteady temperatures can be calculated using the ODE:

$$\frac{d\overline{T}}{dt} = \frac{q_{\text{cell,heating}} - q_{\text{cell,out}}}{m_{\text{cell}} c_{p,\text{cell}}} \tag{6.12}$$

where m_{cell} is the mass of one cell and $c_{p,\text{cell}}$ is the bulk specific heat capacity of the cell. This relation produces an implicit system of equations in time and can be solved using the numerical integration and Newton solver approach previously described in Chapter 5.

There are two notable improvements in this approach versus the generic cold plate approach in the previous chapter. First, this model is reflective of current design practice specific to transportation lithium ion batteries. Second, the model reduces to the individual cell level and becomes independent of the sizing of the pack, reducing the number of design variables for the top-level optimization problem.

6.1.2 Motor Heat Sink Modeling

Aviation motors over 100 kW are seemingly converging on liquid-cooled architectures (including the Siemens SP200D and SP260D [201] and magniX Magni250 and Magni500 [202]). However, the cooling channel design inside the motors has not been publicly divulged (at least, not in any degree of detail). It is unclear whether the physics-based cold plate model introduced in Chapter 5 is applicable to state-of-the-art aerospace motors. It is also desirable to have a motor heat sink model which automatically scales reasonably with the motor as the optimizer changes motor power, which reduces the size of the optimization problem.

Because the materials used to create the motor (including the housing) are generally highly thermally conductive, it is reasonable to treat the motor as uniform in temperature, while the coolant spatially varies in temperature. Therefore, the heat transfer can be computed using the NTU-effectiveness method, as in Equation 6.4. However, computing NTU requires the overall heat transfer parameter $h_{\text{conv}}A$ where A is the heat transfer surface area. Without knowing even

rough parameters of the coolant channels, a bottom-up physics-based estimate of h_{conv} (or indeed, even the geometric parameter A) is ill-advised.

Based on an informal survey of liquid-cooled electric motors, most of them either use coolant channels machined into the motor casing outer diameter (indirect liquid cooling), or direct liquid cooling on the windings inside the motor case. In either event, the area available for heat transfer is roughly proportional to the area of the outer ring of the motor case (A_{case}, the yellow area in Figure 6.3). Ideally, the model would automatically scale A_{case} with rated power, as waste heat is proportional to the rated power. Using rough photogrammetry, I estimated A_{case} for the magniX motors and the Siemens SP200D and obtain values in a narrow band between 630 and 800 kW/m^2 case area.

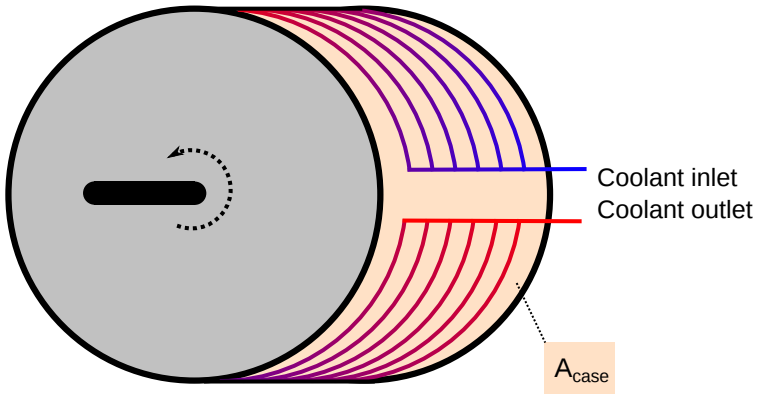


Figure 6.3: General arrangement of the motor heat sink

The best view I could find of an aerospace motor's detailed thermal design is a grainy image of a thermal finite element model of the Siemens SP200D motor in a corporate presentation [201]. Based on the color scale, my best (imprecise) estimate is that the maximum winding temperature is approximately 108° C, the cooling inlet temperature is about 85° C, and the maximum coolant temperature is somewhere around 98° C. The $c_{p,\text{coolant}}$ of Dow Syltherm heat transfer fluid is about 1950 J/kg/K [203]. If the SP200D motor is about 95% efficient and the motor is operating at full power in the thermal FEM image, it is rejecting about 10 kW waste heat, which places the mass flow rate of the coolant at about 0.4 kg/s. Working through equation Equation 6.4 with $\dot{m} = 0.4$

and temperatures as listed, we obtain an estimate of h_{conv}: 2000 W/m^2/K.

The final expression for the heat transfer out of the motor is:

$$q_{\text{cell,out}} = \dot{m} c_{p,\text{coolant}} (1 - e^{-h_{\text{conv}} A_{\text{case}} / \dot{m} / c_{p,\text{coolant}}})(T_m - T_{\text{coolant,in}}) \tag{6.13}$$

where $A_{\text{case}} \approx P_m / 650000$ and h_{conv} is estimated at 2000 W/m^2/K. The time-varying motor temperature can be computed using Equation 6.12.

6.2 Expanded Thermal Management Models

The previous two subsections describe improvements to OpenConcept's existing heat sink models. The following two sections expand OpenConcept's thermal models into two new areas: specifically, active cooling via a vapor cycle machine, as well as weight and power contributions of the hoses and circulation pumps of a liquid TMS.

6.2.1 Chiller Modeling

While motors and power electronics may operate at relatively warm temperatures above 80° C, batteries are limited to cooler temperatures. Depending on the outside air temperature, it may be difficult or even impossible to operate the battery at high power levels without active refrigeration. The Tesla Model 3 uses a vapor cycle to chill the battery coolant in warm conditions [204]. Electric aircraft may benefit from trading drag associated with air-cooled heat exchangers for additional weight and power consumption of a chiller.

A real vapor refrigeration cycle can be plausibly modeled semi-empirically by applying corrections to the theoretical Carnot cycle [205, 206]. In a vapor cycle, refrigerant is evaporated on the cold side (absorbing heat), compressed, condensed under pressure at the hot side (releasing heat), and expanded to low pressure to close the circuit. The arrangement is illustrated schematically in

Figure 6.4 The cycle moves heat q_C from a cold reservoir at temperature T_C to a hot reservoir at temperature T_H while consuming power $\dot{W}$.

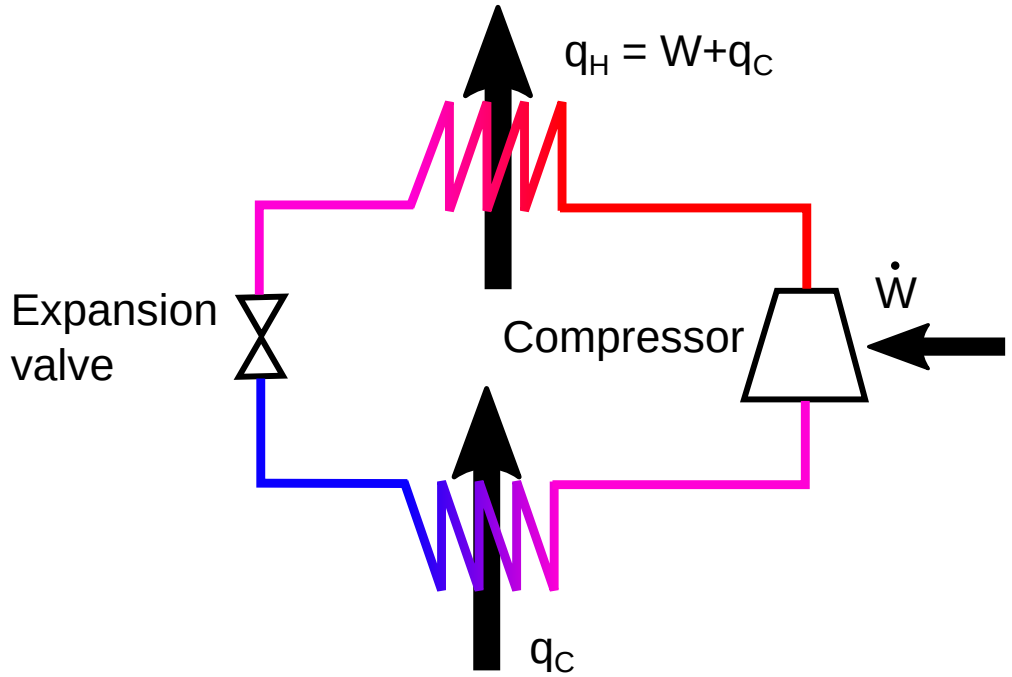


Figure 6.4: Vapor compression refrigeration cycle

The amount of heat moved in a Carnot cycle is proportional to a metric called coefficient of performance (COP) as follows:

$$q_C = \text{COP}\,\dot{W} \tag{6.14}$$

For a perfect Carnot cycle, COP is exactly:

$$\text{COP}_{\text{theo}} = \frac{T_C}{T_H - T_C} \tag{6.15}$$

However, because of unavoidable irreversible losses in a real vapor cycle, we can approximate the heat transfer as

$$q_C = \text{COP}'\,\dot{W} \tag{6.16}$$

$$\text{COP}' = \text{COP}_{\text{theo}}\eta_{\text{cycle}} \tag{6.17}$$

where η_{cycle} is the deviation from the theoretically optimal Carnot cycle, generally on the order of 50 to 70 percent for industrial equipment [206]. I assume $\eta_{\text{cycle}} = 0.4$ because aerospace chillers will likely trade some efficiency for weight. The hot side receives q_C plus the lost shaft power, or:

$$q_H = (\text{COP}' + 1)\dot{W} \tag{6.18}$$

Unfortunately, COP_{theo} has some undesirable numerical properties which I identified through experimentation in OpenMDAO. If the "hot" side becomes cooler than the "cold" side, coefficient of performance takes on a negative sign, implying that the heat transfer is reversed (i.e. *to* the "cold" side instead of away as intended). A more useful notion might be to label the temperatures T_1 and T_2 where the intended heat flow is from side 1 to side 2. COP is also undefined if the two sides equilibrate to the same temperature. Both of these features are undesirable in the context of a system of thermal equations solved iteratively (where T_1 and T_2 take on intermediate values set by a solver). For a thermal management system designed to cool a battery, the heat transfer problem only becomes *easier* if the outside air temperature becomes much colder than the battery (notionally, $T_2 < T_1$) whereas the theoretical COP expression actually makes the heat transfer problem harder. Finally, the asymptotic behavior of the theoretical COP implies that a virtually unlimited amount of heat can be removed using a vapor cycle using only modest power, as long as the temperature gap $T_2 - T_1$ remains tiny but positive. In reality, irreversible flow and pumping losses would quickly overwhelm the useful heat transfer. The COP should intuitively asymptote to a finite positive number for $T_2 - T_1 < |\epsilon|$.

To fix these numeric issues, I alter the theoretical expression as follows:

$$\text{COP}'' = \text{COP}_{\text{theo}}\tanh(\Delta T)\eta_{\text{cycle}} + \text{COP}_{\text{asymptote}}\frac{1 + \tanh(-(\Delta T + \epsilon_{\Delta T}))}{2} \tag{6.19}$$

$$\text{COP}' = \frac{\text{COP}'' e^{\alpha \text{COP}''} + \text{COP}_{\text{asymptote}} e^{\alpha \text{COP}_{\text{asymptote}}}}{e^{\alpha \text{COP}''} + e^{\alpha \text{COP}_{\text{asymptote}}}} \tag{6.20}$$

where ΔT is the temperature gap $(T_2 - T_1)$, $\text{COP}_{\text{asymptote}}$ is the value to which COP asymptotes due to irreversible losses at low ΔT, and $\epsilon_{\Delta T}$ and α are tunable parameters. Equation 6.19 (an application of the discrete induced exponential (DIE) [207, 208]) takes the theoretical COP, applies an efficiency factor, but uses a sigmoid function to keep COP positive when the temperatures flip. The parameter $\epsilon_{\Delta T}$ controls where the sigmoid function sign flips and should be positive and modest (I chose 3° K). Equation 6.20 is a "smooth minimum" operator that essentially selects the lower of $\text{COP}_{\text{theo}}\eta_{\text{cycle}}$ or $\text{COP}_{\text{asymptote}}$ but maintains continuous smooth derivatives. The parameter α controls the smoothness of Equation 6.20 and it should be modestly negative (I chose -1.5). The numerically robust COP is compared to the theoretical result for $\eta_{\text{cycle}} = 0.4$, $T_2 = 300$, $\text{COP}_{\text{asymptote}} = 10$ in Figure 6.5.

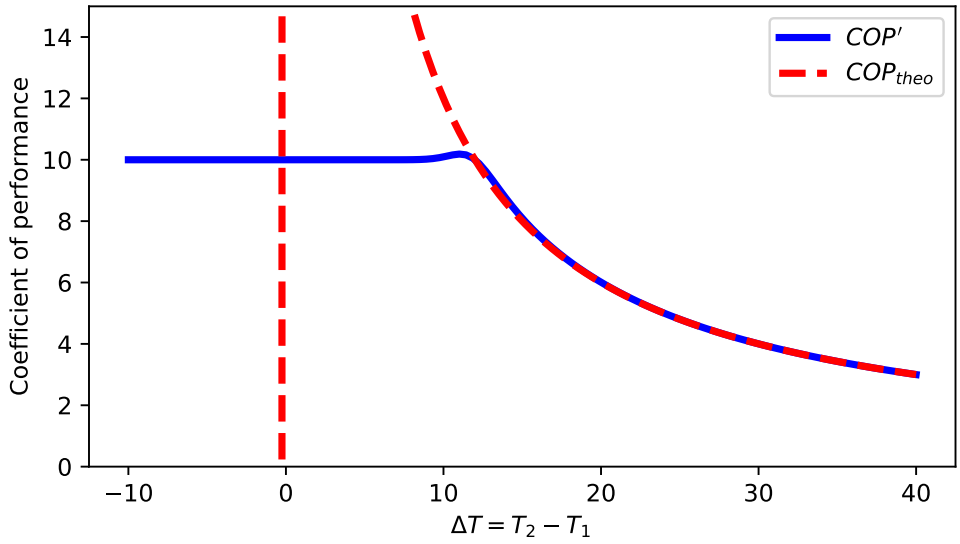


Figure 6.5: Theoretical versus robust COP

An additional complicating factor is that in some cases it may be preferable to bypass the chiller altogether and pump hot coolant from the battery directly into an air-cooled heat exchanger without undergoing any thermal lift from the chiller. For example, the Tesla Model 3 thermal management

system uses a five-way valve to bypass the chiller in cold conditions [204]. An aircraft may benefit from bypassing the chiller while flying at high altitude (where the outside air is very cold), while using the chiller during climb and descent (where the air may be even warmer than the battery). Figure 6.6 schematically illustrates the chiller subsystem in an active (non-bypassed) configuration, while Figure 6.7 shows the subsystem in direct air cooling mode. While in bypass mode, no shaft work $\dot{W}$ is required, which saves electrical load. There is also no upper limit on heat transfer q_1 in bypass mode (heat transfer is limited only by the temperature differential and heat exchanger performance), whereas in chiller mode the upper limit is $\dot{W}_{\text{rated}}\text{COP}_{\text{asymptote}}$, where $\dot{W}_{\text{rated}}$ is the maximum design power of the chiller compressor.

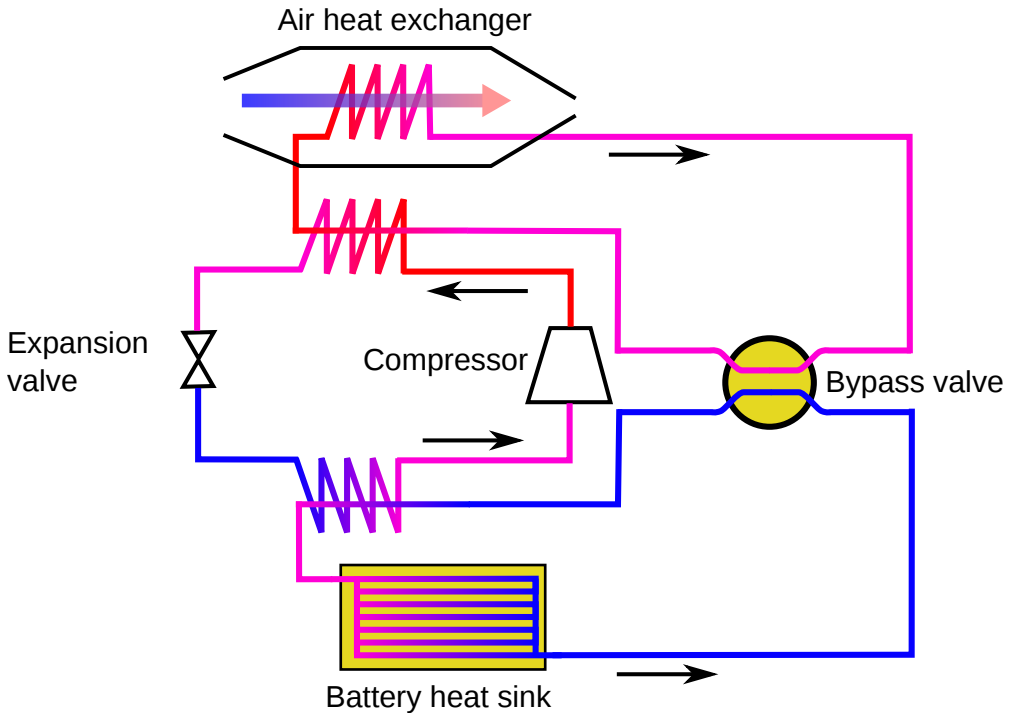


Figure 6.6: Chiller subsystem in active configuration

I introduce a bypass parameter $0 \leq \beta \leq 1$ representing the "degree" of bypass. When $\beta = 1$, the chiller is bypassed altogether. When $\beta = 0$, the chiller is completely active. There are multiple reasonable ways to implement this, but after experimenting with various approaches, the one I selected (for numerical stability) is:

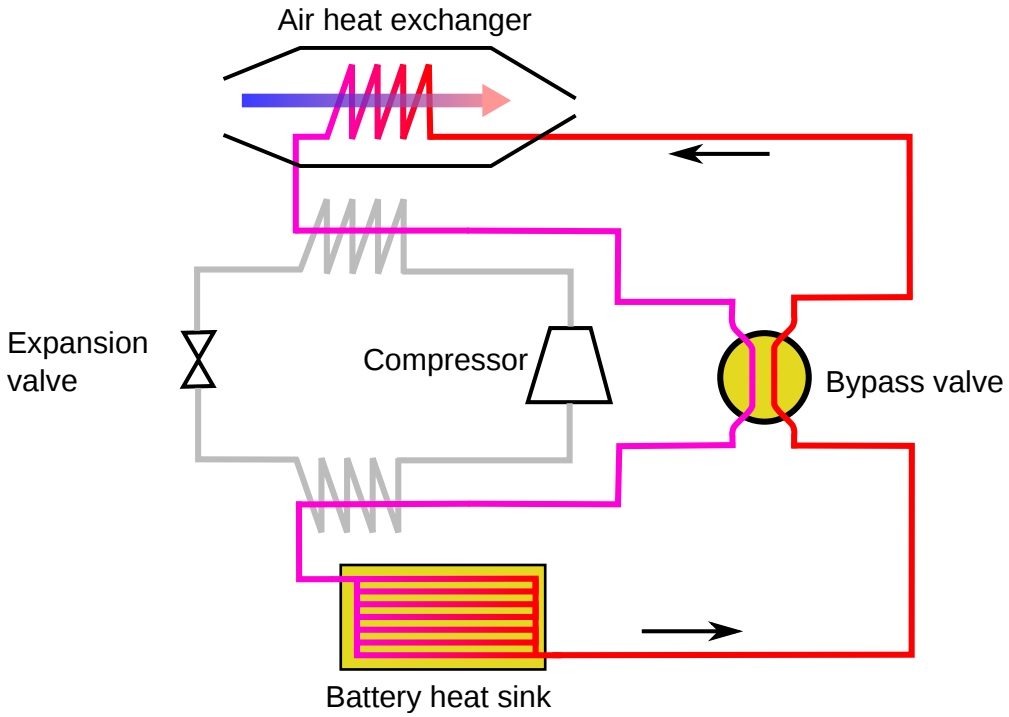


Figure 6.7: Chiller subsystem in bypassed configuration

$$T_{2,\text{out}} = \beta T_{1,\text{in}} + (1-\beta) T_{2,\text{out,chiller}} \tag{6.21}$$

$$T_{1,\text{out}} = \beta T_{2,\text{in}} + (1-\beta) T_{1,\text{out,chiller}} \tag{6.22}$$

$$\dot{W}' = (1-\beta)\dot{W} \tag{6.23}$$

where $T_{1/2,\text{out,chiller}}$ are the coolant outlet temperatures of the previously-defined chiller model, $T_{1/2,\text{in}}$ are the coolant temperatures coming into the chiller (*out from* the battery and air-side heat exchangers), and $\dot{W}'$ is the "actual" shaft power used (which will generally be consumed as an electrical load). The temperatures represent a notional mass-averaging between the bypassed and non-bypassed temperatures and are well-behaved but not strictly valid between $\beta = 0$ and $\beta = 1$ due to temperature-related COP effects. β is a good candidate for a dynamic control parameter in a trajectory optimization context; in my experience, the optimizer tends to send β to the upper and

lower bounds, which is the desired physical behavior.

Finally, the weight of the chiller system can be estimated using a linear specific power relation:

$$m_{\text{chiller}} = \frac{\dot{W}_{\text{rated}}}{p_{\text{chiller}}} \tag{6.24}$$

where p_{chiller} is the specific power of the chiller. p_{chiller} is probably somewhere in the neighborhood of 200 W/kg based on limited public data for the Honeywell Micro VCS (20kW heat removed, $T_H = 160°$ F, $T_C = 50°$ F [209], 140 lb weight [210]).

6.2.2 Pump and Hose Modeling

Since realistic coolant flow rates are relatively high (in the neighborhood of 0.5 to 1.0 kg/s), the electrical load due to coolant circulation pumps and the weight of the coolant hoses cannot be completely neglected. Furthermore, while the models in Chapter 5 considered the airplane-level drag effects of air-side pressure losses in the heat exchanger, there was no equivalent penalty on the coolant side. Put another way, if the heat exchanger is too short in the airflow direction, the liquid coolant channels become narrow and constricted, increasing pressure losses across the liquid side of the heat exchanger and resulting in higher pump power consumption and heavier hoses. There is also a tradeoff between weight and power in the fluid system design. Larger hoses have lower pressure losses (reducing pump energy consumption), but because the coolant and hose material are relatively heavy, the weight penalty will increase energy consumption at the airplane level.

Pump power consumption can be modeled as follows:

$$P_{\text{pump}} = \frac{P_{\text{fluid}}}{\eta_{\text{pump}}} \tag{6.25}$$

where P_{fluid} is the theoretical fluid power and η_{pump} is the pump efficiency (including fluid,

mechanical, and electrical losses). The fluid power is:

$$P_{\text{fluid}} = \dot{V}\Delta p \tag{6.26}$$

where $\dot{V} = \dot{m}/\rho$ is the volumetric flow rate and Δp_{pump} is the pressure *rise* across the pump. In a closed loop, the pressure rise across a circulation pump must equal the accumulated pressure losses along the loop, so for i components (such as hoses, heat sinks, etc.):

$$\Delta p_{\text{pump}} = -\sum_i \Delta p_i \tag{6.27}$$

Based on an informal survey of commercially-available fuel and hydraulic pumps on certified commercial airplanes (tabulated in Appendix D), I estimated that $\eta_{\text{pump}} = 0.35$ is an appropriate choice for coolant circulation. The weight of the pump can be estimated as:

$$m_{\text{pump}} = \frac{P_{\text{pump,rated}}}{p_{\text{pump}}} \tag{6.28}$$

where p_{pump} is the pump specific power (estimated at 450 W/kg rated electrical power) and $P_{\text{pump,rated}}$ is the pump power rating. Across all operating conditions the designer or optimizer must ensure that $P_{\text{pump}} \leq P_{\text{pump,rated}}$.

Pressure drop across a coolant hose is straightforward to compute. Assuming flow in the pipe is in the turbulent regime ($4000 < Re_{d_h} < 10^5$) the Darcy friction factor f for a smooth-walled pipe with diameter $d_{\text{hose}} = d_h$ can be computed using the well-known Blasius correlation [211] as follows:

$$f = 0.3164 Re_{d_h}^{1/4} \tag{6.29}$$

where Re_{d_h} is the hydraulic diameter Reynolds number. Then, the pressure drop across a length ℓ

of pipe can be computed using the Darcy-Weisbach equation:

$$\Delta p = f\ell\frac{\rho U^2}{2d_{\text{hose}}} \tag{6.30}$$

where U is the mean flow velocity and ρ is the fluid density. U can be computed from the mass flow rate $\dot{m}$ and cross section:

$$U = \frac{4\dot{m}}{\pi\rho d_{\text{hose}}^2} \tag{6.31}$$

Finally, the weight of the hose is comprised of the weight of the fluid in the hose plus the weight of the hose itself. The hose weight depends on the hose wall thickness, which depends on the design pressure and diameter based on the hoop stress, as follows:

$$t_{\text{hose}} = \frac{p_{\text{des}}d_{\text{hose}}}{2\sigma_{\text{des}}} \tag{6.32}$$

where p_{des} is the design fluid pressure and σ_{des} is the hose design stress. The total weight of the hose and coolant is then:

$$m_{\text{hose}} = \pi\ell\big((d_{\text{hose}} + t_{\text{hose}})t_{\text{hose}}\rho_{\text{hose}} + (d_{\text{hose}}/2)^2\rho_{\text{coolant}}\big) \tag{6.33}$$

The material weight and design stress were estimated based on a selection of low-pressure aerospace fluid hoses from Eaton Corporation [212]: $\sigma_{\text{des}} = 300$ psi at the operating point, and $\rho_{\text{hose}} = 1356$ kg/m^3. The tabulated hose data can be found in Appendix D.

6.3 Empirical Validation of OpenConcept TMS Models

While the individual component models developed in this chapter and Chapter 5 are all rooted either in empirical data or relatively simple physics, there is a paucity of empirical data with which

to validate time-varying temperature profiles of electric aircraft components. Fortunately, Pipistrel Aircraft, a Slovenian aircraft manufacturer and a leading developer of electric propulsion systems, agreed to share proprietary flight test data to aid in validating OpenConcept's thermal models. I obtained time series data from flight tests of the Pipistrel Velis Electro, the first electric aircraft to receive a type certificate from the major aviation regulators.

Figure 6.8: Pipistrel Velis Electro

(Author: Wikimedia user Andrejcheck License: CC-BY-SA)

The Velis Electro (pictured in Figure 6.8) is a two-seat, single-motor electric aircraft designed for low-cost primary flight training. It has a lightweight composite airframe and a three-blade fixed-pitch propeller [213]. The TMS consists of two independent liquid coolant loops: one for the motor and power electronics module, with a radiator near the nose, and a second loop for the batteries, with a ducted heat exchanger located in the rear of the aircraft.

The flight test data set consists of time series data for three flights. Each flight data file contains:

- Flight conditions (indicated airspeed, barometric altitude, outside air temperature)
- Control settings (motor power)

- Component temperatures (measured in the motor, both batteries, and TMS)

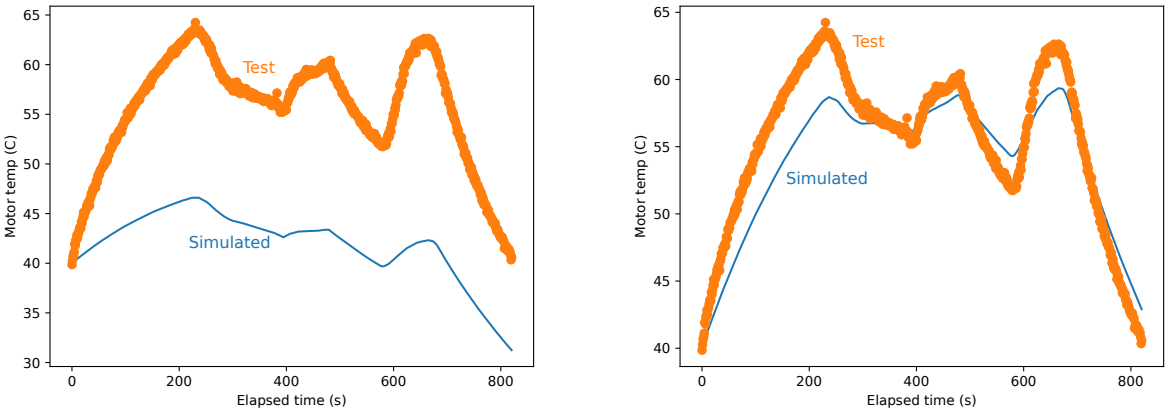
After examining the data for one flight, I decided to attempt to simulate the motor temperature profiles, in the hopes that the simulated temperatures would match favorably with the measured temperatures and provide some additional confidence in the overall thermal modeling approach: in particular, the motor heat sink and ducted heat exchanger models. Lacking any insight into the design of the heat sinks inside the proprietary battery packs, I elected not to attempt to simulate the battery TMS at this time.

In order to model the motor TMS loop, I needed to obtain a fairly significant amount of aircraft design data. Pipistrel is unusually transparent in sharing technical data publicly. In addition to the Velis Electro and E811 motor type certification data sheet (TCDS) available from EASA, the website offers additional useful technical data such as dimensioned drawings [214]. Beyond the public data, I was provided with general configuration information including system diagrams, coolant mass flow rates, and heat exchanger fin details, as well as measurements of the heat exchangers and heat exchanger ducts. The non-public design data are not reported in this dissertation.

Information that was not provided and therefore needed to be identified or estimated includes:

- Motor heat production as a function of throttle setting (i.e. the efficiency map)
- Motor heat sink architecture and dimensions
- Motor active thermal mass

Using OpenConcept, I created a coolant loop including the motor (with associated heat sink) and a louvered-fin heat exchanger (using empirical data from [191]). Because the Velis Electro flies only at very modest speeds, I used the incompressible model for the heat exchanger duct mass flow rate introduced in Chapter 5. I obtained the motor weight (22.7 kg) and maximum rated power (57.6 kW) from the TCDS for the Pipistrel E-811 electric motor [215]. Lacking any additional information about the E-811 motor's cooling channels, I elected to use the heat sink parameters



(a) No Tuning

(b) Efficiency tuned ($\eta_m = 0.92$)

Figure 6.9: Parameter identification flight test data (Flight A)

identified in Subsection 6.1.2 as-is. Instead of attempting to duplicate the test flight profiles using OpenConcept's built-in mission analysis capability, I fed the outside air temperature, airspeed, altitude, and motor throttle setting directly into the propulsion TMS model — this eliminated several sources of discrepancy between model and experiment.

To identify motor efficiency η_m, I used only the first test flight and altered η_m until the simulated and measured thermal profiles matched closely (as the heating is a strong function of η_m). The second and third test flights were held out for validation tests. Assuming $\eta_m = 0.95$ *a priori*, I simulated an initial time series (Figure 6.9a). The simulated heating was obviously too low. I iteratively adjusted the efficiency to $\eta_m = 0.92$ and obtained Figure 6.9b. This was a major improvement, but the simulated rate of temperature change was too slow. Because the motor mass consists of a rotor and a stator, it is plausible that the heating may be concentrated in the stator, reducing the apparent thermal mass m'_m of the overall motor. I adjusted the thermal mass downward by about 30% and obtained the result in Figure 6.10.

With good agreement and no additional changes to system parameters, I simulated the second and third flights (Figure 6.11). Flight B and Flight C both consist of repetitive flight test conditions

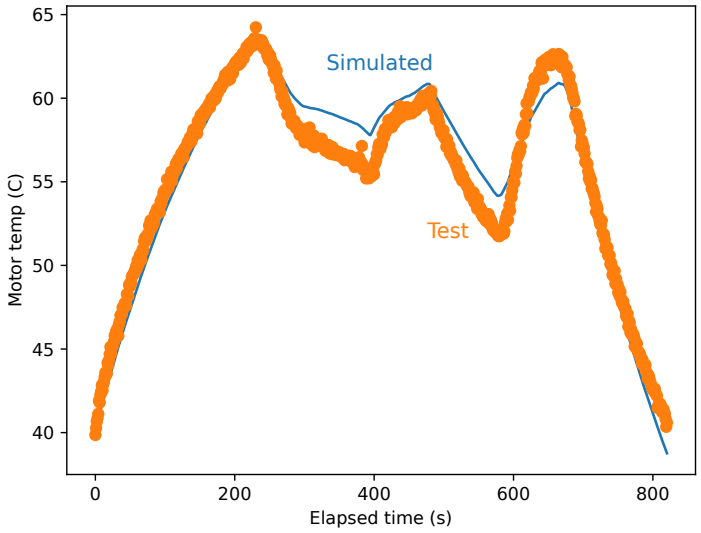
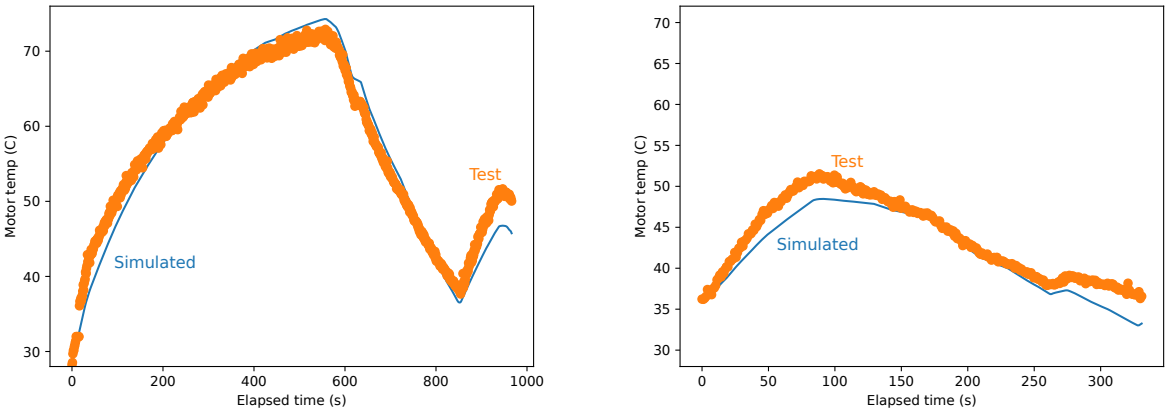


Figure 6.10: Test Flight A with final parameter tuning ($\eta_m = 0.92$, $m'_m = m_m/1.4$)

with full stop landings in between; I selected one condition arbitrarily. While I did not consciously tune parameters based on Flight B, I did at least *view* the data before re-tuning the parameters following an unrelated OpenConcept model bug fix. Flight C is a "true" blind validation case — no model changes occurred after seeing the simulated result.

Flight B (Figure 6.11a) shows outstanding agreement between simulation and experiment. Flight C (Figure 6.11b) shows reasonably good agreement. Because Flight C's profile consists of a very short hop, the motor operating temperatures remain fairly low. As resistive heating is higher at lower temperatures, it is possible that a higher-fidelity motor efficiency map would explain the small (less than 3° C) discrepancy at top-of-climb. Another possible explanation for remaining discrepancy between model and experiment is the effect of propwash on the heat exchanger duct. The Velis Electro's motor cooling duct inlet is located in the propeller slipstream. I accounted for this by adding 200 Pa total pressure boost across the duct (based on a rough estimate of propeller disc loading), but the true amount of additional forced convection attributable to the propeller is unknown, as is the thrust-dependence of this effect.



(a) Flight B

(b) Flight C

Figure 6.11: Validation flight test data

6.4 Concluding Remarks

This chapter introduced new and improved thermal models in OpenConcept. The semi-empirical battery and motor heat sink models reflect modern design thinking and scale well as the components change size during optimization. The motor heat sink and ducted heat exchanger models were empirically validated using Pipistrel flight test data, confirming that OpenConcept's numerical methods for solving the thermal equations are correct and that the component thermal models tested in the simplified validation case are reasonably predictive. This chapter is entirely new and unpublished material.

CHAPTER 7

MDO of a Parallel Hybrid Transport Aircraft

In the United States, most commercial flights are under 1000 nautical miles [216], and the large majority of these are flown using single-aisle aircraft such as Boeing 737 or Airbus A320 series airplanes. As batteries continue to improve, these modest ranges start to look interesting as applications for hybrid-electric propulsion. The 737 Max and A320neo are both sized for trans-Atlantic range, a consequence being that they take off significantly below design MTOW on shorter domestic flights. If a modular battery system was conceived that could be loaded and unloaded depending on mission demands, a parallel hybrid propulsion architecture could potentially exploit this unused payload capacity.

NASA is actively developing a flying testbed to demonstrate technologies necessary for a single aisle-sized parallel hybrid aircraft [217]. The parallel hybrid concept would consist of a 737-sized airframe with N+3 GTF engines and 1 MW of hybrid power. Optimizing the TMS architecture of this aircraft is important because of the high power levels (and therefore, waste heat) involved.

I was tasked by NASA with performing a MDO trade study and sensitivity analysis of the thermal management system of the parallel hybrid single aisle aircraft concept. The trade study is between a conventional TMS with a vapor cycle chiller, and a novel TMS with a thermoacoustic chiller. This chapter covers the modeling approach for the airframe, hybrid propulsor, and conventional TMS as well as selected trends from the optimization results. Results for the thermoacoustic TMS are the subject of a separate, upcoming paper.

7.1 System Architecture

The first step in this study was defining a baseline architecture for the TMS and the propulsion subsystem. Because the focus of the trade study is on those two subsystems only, I did not attempt to define new system architectures for other functions (such as hydraulic). The system architecture is illustrated schematically in Figure 7.1.

The propulsion system consists of a parallel hybrid variant of the NASA N+3 turbofan engine, plus supporting electrical systems. The N+3 turbofan is a NASA concept for a conventional (non-hybrid) GTF with 2030-level technology [218, 219]. I used a moderate fidelity model of the N+3 [218], added shaft power input from a parallel hybrid motor, and reoptimized the engine cycle. The details of the engine optimization procedure are discussed in Section 7.2.2.

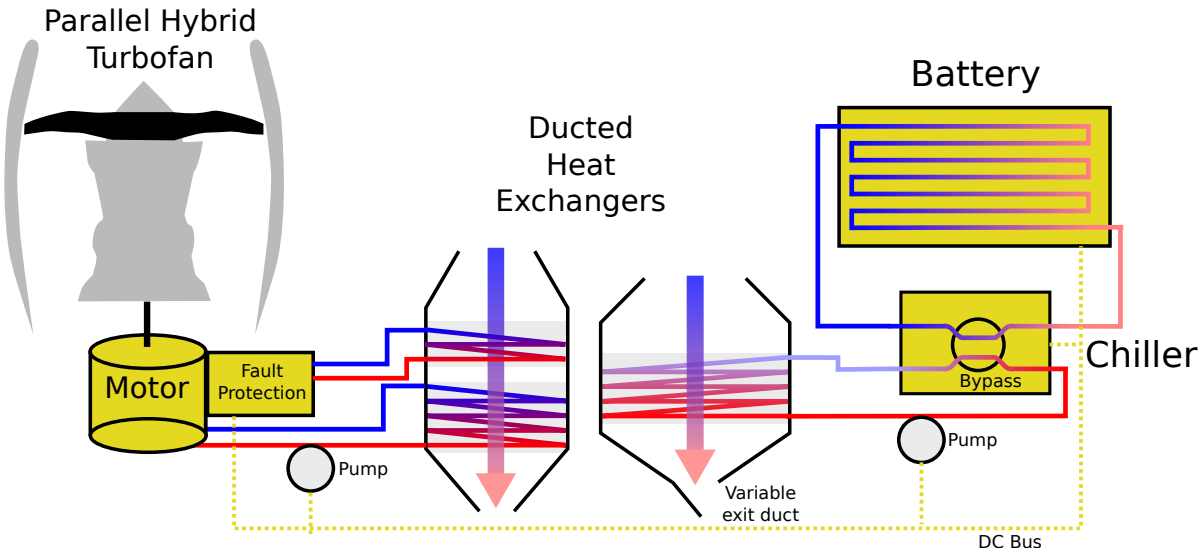


Figure 7.1: Propulsion system architecture for the parallel hybrid aircraft

The hybrid motor in the engine is connected to the DC power bus via its power electronics module and a fault current protection module, as seen in Figure 7.1. For the NASA trade study, a novel high-speed, solid-state fault current protection circuit was included. The DC bus is fed by a sizable (on the order of 3000 lb) battery bank in the fuselage of the aircraft. Because of

the substantially higher voltages and for simplicity, the electrical system for propulsion and the propulsion TMS is kept separate from the rest of the airplane electrical system.

The thermal management architecture consists of three coolant loops and two heat exchanger ducts. To avoid running hoses to the pylon from the fuselage and back, the motor and fault protection thermal management loops exchange heat with a duct co-located on the pylon. Since the motor can run hotter than the fault protection, the motor and fault protection heat sinks are plumbed to separate heat sinks in a common duct, and the motor heat exchanger lies downstream of the fault protection heat exchanger.

The battery thermal management loop includes the battery heat sink, a second ducted heat exchanger, as well as an electrically-driven chiller, all illustrated in Figure 7.1. The battery duct is located in the wing side-of-body fairing (similar to environmental control systems ducts in existing aircraft), and is equipped with a variable-area exit nozzle to control airflow rates through the duct. The chiller improves performance in warm conditions and potentially enables a smaller heat exchanger. It draws electrical power for the compressor from the propulsion DC power bus. Two electric pumps, one for the motor TMS and one for the battery, circulate coolant in the loops and are sized to overcome the pressure losses in hoses and heat exchangers. They also draw electrical power from the propulsion DC power bus.

7.2 System Modeling

7.2.1 Airframe Modeling

The baseline airplane takes the aerodynamic and weight characteristics of the 737-800. A quadratic drag polar was curve fit based on an empirical drag polar of the 737-800 [220]. Weight data for the 737-800 was assembled from public sources [221]. I assume no change in the baseline operating empty weight (OEW) due to two offsetting effects: the 737-800's aluminum airframe

could be lightened with judicious use of composite materials, but the higher- BPR N+3 engine will be heavier than the 737-800's CFM-56 turbofans. I also assume no change in the baseline drag polar, again due to offsetting effects of incremental aerodynamic improvements and higher installation drag for the higher BPR. Weight and drag increments attributable to the hybrid propulsion system are added on top of the baseline OEW and drag.

7.2.2 Hybrid Turbofan Modeling and Optimization

Parallel hybrid architectures primarily benefit from replacing fuel energy with electrical energy directly, but there may also be some benefits at the corners of the flight envelope which result in a more efficient thermodynamic cycle at the cruise condition. In order to capture this effect, I redesigned the engine cycle of the N+3 accounting for hybrid power. The base engine model consists of the pyCycle N+3 hybrid turbofan model [218][1]. I then added an additional shaft power input to the low-speed shaft to account for the hybrid electric power. e In addition to the engine model, the pyCycle repository also contains a MDP optimization problem definition. MDP signifies that the engine cycle design is constrained by performance requirements at off-design conditions. The cycle design consists of design flow rates, BPR, and turbomachinery pressure ratios, such as the low-pressure compressor (LPC) pressure ratio and the overall pressure ratio (OPR) (which, in this parameterization, essentially sets the HPC pressure ratio). In the case of the N+3 GTF, there are minimum thrust specifications for top-of-climb (TOC), cruise (CRZ), sea-level static (SLS), and rolling takeoff (RTO), as outlined in Table 7.1. The design point (where design pressure ratios and mass flows are matched) is the TOC point.

I slightly modified the original optimization problem parameterization for clarity, defining the engine design problem as listed in Table 7.2. I solved the engine design problem for both the hybrid engine and the conventional (non-hybrid) N+3 GTF, using the SNOPT optimizer and

[1]Source code available at https://github.com/OpenMDAO/pyCycle/tree/3.2.0/example_cycles/

Condition	Mach	Altitude (ft)	Thrust Req (lb)
RTO	0.25	0	22,800
SLS	0.0	0	28,621
Cruise	0.8	35,000	5,510
TOC (design)	0.8	35,000	6,200

Table 7.1: Operating points for the N+3 hybrid engine design problem

		Unit	Lower	Upper	Optimum (Hybrid)	Optimum (Conventional)
minimize	fuel burn at CRZ	lb/s			0.6627	0.6652
with respect to	fan PR_{des}		1.2	1.4	1.308	1.307
	LPC PR_{des}		2.5	4.0	**4.0**	**4.0**
	OPR		40	70	65.71	64.54
	V_{bypass}/V_{∞}		1.35	1.45	**1.35**	**1.35**
	BPR at TOC		17.0	24.5	19.66	19.53
	$\dot{m}$ at TOC	lb/s	780	850	810.92	810.92
	P_m at RTO	kW	0.0	1000.0	**1000.0**	—
	P_m at SLS	kW	0.0	1000.0	**1000.0**	—
	P_m at TOC	kW	0.0	1000.0	**0.0**	—
subject to	T_4 at RTO	R		3400	3097	3176
	T_4 at TOC	R		3230	2927	2932
	d_{fan}	in		100	**100**	**100**

Table 7.2: Optimization definition and results for N+3 engine MDP problem

OpenMDAO's pyOptSparse wrapper [155, 222, 223]. The optimized engine design parameters for both cases are listed in Table 7.2. The optimal design for the conventional engine in Table 7.2 differs slightly from the baseline cycle in the N+3 paper [219] because the paper used a manual procedure to select the final design. Despite a significant hybrid boost at the sea-level conditions which reduced T_4 markedly, the hybrid engine reduced cruise fuel burn (hybrid power *off*) by only 0.37% compared to the optimized conventional design. This very modest cycle-only improvement may reflect that the baseline engine is not temperature-limited at the RTO and SLS conditions, thus reducing the value of temperature reductions at those conditions.

Incorporating the pyCycle model of the optimized engine in OpenConcept directly is not feasible for two main reasons. First, pyCycle uses a Newton solver to converge the systems of equations

defining the model, and thus requires good guesses of the thermodynamic states at each station if reliable convergence is to be achieved. OpenConcept will likely feed dramatically different flight conditions to the engine model as it converges, which has a high likelihood of causing pyCycle itself to fail to converge. Second, pyCycle is relatively expensive to compute per flight condition compared to the rest of OpenConcept's models.

Instead of coupling pyCycle to OpenConcept directly, I used a surrogate modeling approach. I generated an "engine deck" for the conventional and hybrid N+3 engines by computing fuel flow, thrust, and surge margins at a full-factorial grid of flight conditions. The independent variables on the grid are altitude, Mach number, and throttle. The hybrid engine includes hybrid electric power setting as a fourth variable in the engine deck. Generating the deck with pyCycle is a one-time expense — generally, a few hours on a desktop computer. After generating the deck, I fit a surrogate model to the data using the kriging method. Kriging allows OpenConcept to query the engine model by interpolating between points in the engine deck. Appendix A includes details of the surrogate modeling procedure and the goodness of fit.

Figure 7.2 shows the full-throttle thrust of the optimized, conventional N+3 turbofan across a variety of flight conditions. Maximum thrust at each operating point is limited by the shaft speed of each spool as well as T_4. As expected, the thrust lapses quickly with speed and altitude. Figure 7.3 shows the thrust specific fuel consumption (TSFC) of the optimized hybrid engine at 90% (approximately cruise) throttle, with zero electrical power on. At the cruise condition ($M = 0.8$, 35,000 feet altitude), the TSFC is significantly better than the 737-800's CFM56-7 engine, which is two generations behind in technology. Adding hybrid power reduces fuel consumption significantly. Figure 7.4 shows that 10%–25% TSFC savings at a given thrust level are achieved with 1 MW hybrid power. The relative fuel burn benefit is more pronounced at lower throttle settings.

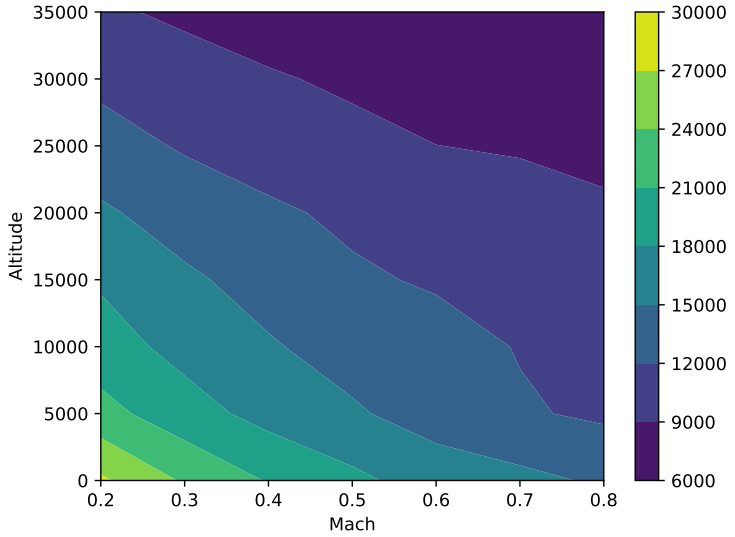


Figure 7.2: N+3 engine thrust at full throttle (lb)

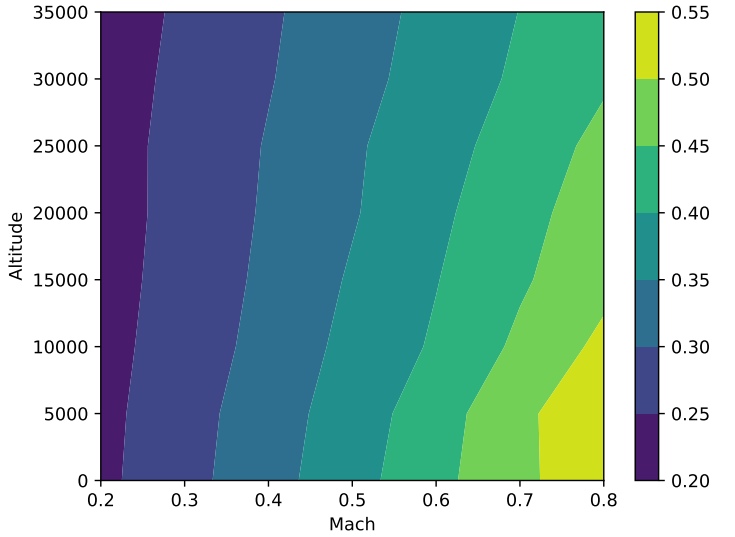


Figure 7.3: N+3 hybrid TSFC at 90% throttle (lb/hp/hr)

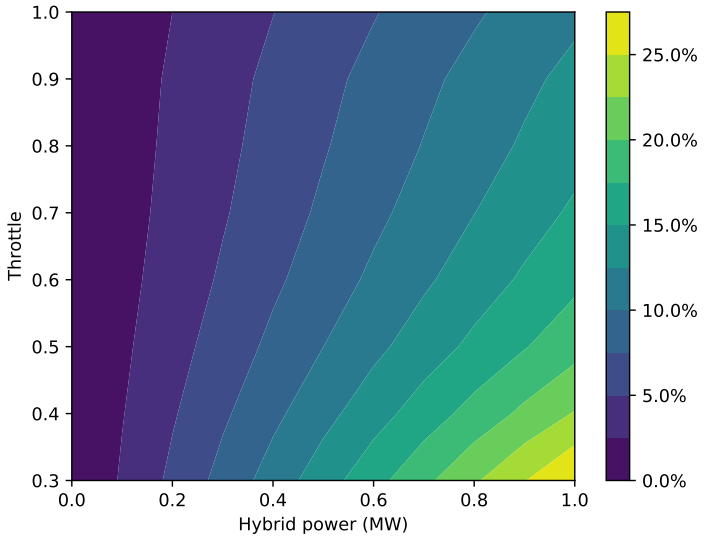


Figure 7.4: N+3 hybrid fuel savings with power on; $M = 0.8$, 35,000 feet

7.2.3 Electrical Modeling

The propulsion electrical architecture described in Section 7.1 and illustrated in Figure 7.1 was modeled using OpenConcept electrical component models. Because this is a 2030+ technology study, there are no empirical (or even detailed, simulated) performance maps or discharge curves for the motor or the battery. The hybrid propulsion motor uses OpenConcept's `SimpleMotor` class and assumes the default efficiency of 97% (including the power electronics module) with a specific power of 5 kW/kg. The battery uses the `SOCBattery` class, with a baseline specific energy of 400 Wh/kg and an efficiency of 95%. This is consistent with SOA lithium-ion batteries discharging at rates around 1C or 2C.

The fault current protection circuit, designed by a research team at the Naval Postgraduate School (NPS), is also considered. The proposed solid-state design has very fast response time, which helps prevent damage to other components in the event of a fault. However, it is "only" 99.7% efficient. The fault current limiters are therefore modeled as an inline electrical efficiency loss.

7.2.4 Thermal Modeling

The TMS architecture described in Section 7.1 and illustrated in Figure 7.1 was modeled using the OpenConcept thermal components described in the previous two chapters. I used the semi-empirical battery and motor heat sink models from Chapter 6 with default parameters, except that the effective heat transfer area of the motor was reduced by about 40% from the default. This is a conservative assumption, as my data set for aerospace motors was limited to 500kW, and there are qualitative reasons to expect that very high power motors will have higher power-to-surface area ratios. The pump and hose models are as described in Chapter 6, with the default empirical constants unchanged. Because of the high cruise speed of this transport aircraft, I used heat exchanger duct models with compressibility effects, which increases the computation cost of the problem but is necessary to account for the Meredith effect, especially at higher power levels.

The fault protection circuit is a special case. At 99.7% efficiency, it produces a significant amount of waste heat for such a small component. As the component is still in the detailed design phase, the geometry of the coolant channels is not yet firm. The NPS system specification does not quote an internal temperature limit, but does list a coolant inflow temperature requirement of 50° C or less. Because the fault protection is lightweight relative to the amount of waste heat, I make a quasi-steady assumption (fault protection waste heat is instantaneously transferred to the coolant loop), and then add an optimization constraint that the fault protection heat exchanger outlet temperature is no higher than 50° C.

7.3 Baseline Optimization

I set up an MDO problem to minimize fuel burn. The design variables, constraints, and upper/lower bounds are listed in Table 7.3. The optimizer, IPOPT [224], controls high-level mission parameters such as the cruise altitude, takeoff weight, and battery weight. The aircraft design parameters include the chiller size, physical geometry of the heat exchangers, and sizing of the

coolant hoses. Finally, the optimizer has control of *dynamic* (i.e., time-varying) parameters of importance. These include the power setting of the hybrid motor, the battery duct nozzle setting (which controls mass flow in the duct), and whether the chiller is active or bypassed. The dynamic control parameters are posed as linear ramps between a start value and an end value for each mission segment (for a total of 8 degrees of freedom for each dynamic control).

The reference mission is an 800 nautical mile passenger mission with a payload of 20,000 kg (representative of around 180 average passengers plus 40 kg cargo per person). Airspeeds, climb rates, altitudes, and throttle settings for the optimized hybrid airplane are illustrated in Figure 7.5. To capture a critically hot day (worst case for TMS design), I incremented the standard atmosphere temperatures up by 20° C.

Because the N+3 engine is generously sized for the 737-800 airframe, it flies at a relatively low throttle setting on descent. Unfortunately, adding power to the LP shaft creates compressor stall concerns at low throttle settings. To allow for reasonable hybridization on descent and avoid operability issues, the continuous descent profile is fairly shallow. Nonetheless, the hybrid motors are limited to about 30% of max power on descent.

Table 7.4 shows the optimized design variable values for the hybrid airplane. The optimizer uses significant hybrid power on this 800 nautical mile mission, which reduces its fuel burn by around 3.3% compared to a non-hybrid airplane with the N+3 engine flying the same mission. The optimizer also sizes the chiller above its lower bound, indicating that an actively-chilled TMS is favorable despite the high cruise altitude and short duration of this mission. Most of the TMS design variables, including the heat exchanger flowpath lengths and hose and pump sizes, are also off their bounds and are being sized by the optimizer.

The dynamic control parameters and component temperatures are illustrated on Figure 7.5 and 7.6. As expected, the optimizer maximizes the amount of battery power that it uses (i.e., hybrid motor power is at its limit throughout the mission). The optimizer also hits the battery and motor temperature limits through a combination of controlling the duct mass flow rates and using the

chiller. The chiller is bypassed for all but the descent phase, which is attributable to thermal mass on ascent and cold temperatures at cruise. As the airplane descends through the warmer lower atmosphere with a pre-heated battery, the chiller kicks in to avoid excessive cooling drag.

Figure 7.6 also illustrates the net force through the cooling ducts. Both ducts actually develop a little bit of net thrust at certain flight conditions, and the drag contribution is relatively modest (a few pounds) at most points during the flight. The higher speed of this transport aircraft increases compression in the duct and enables the significant Meredith effect recovery.

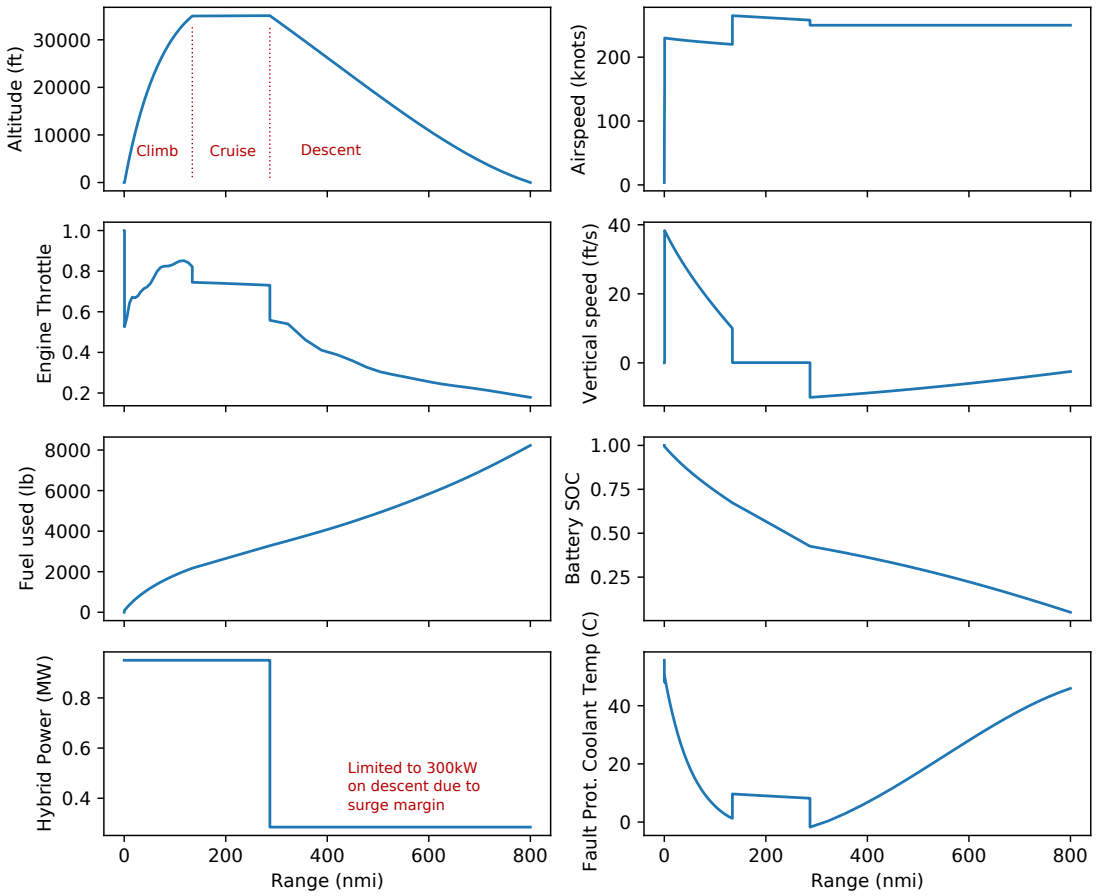


Figure 7.5: Optimized flight profile and control parameters — base case

Table 7.4 includes several comparison airplanes. The 737-800 comparison airplane uses the same empty weight and airframe drag model, but substitutes a pyCycle-generated engine deck of

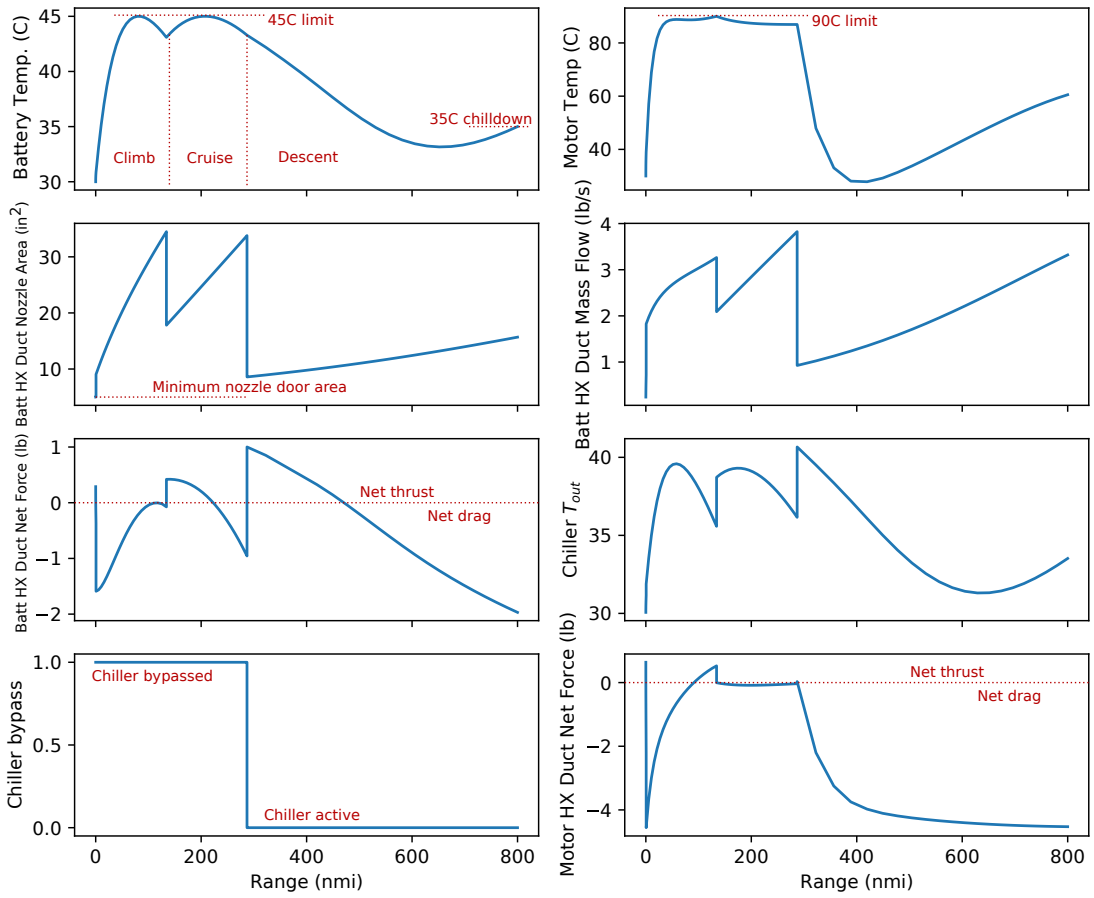


Figure 7.6: Optimized thermal trajectory and control parameters — base case

the CFM56-7 turbofan. The "N+3 GTF" comparison airplane is a non-hybrid variant with the optimized, conventional GTF engine. The modeled 737-800 uses over 30% more fuel than the N+3 conventional comparison. Finally, the "no TMS" hybrid airplane accounts for the weight and drag of the hybrid motor, power electronics, and batteries, but not the weight or drag of thermal management. Figure 7.7 shows that the parallel hybrid saves about 3.3% fuel compared to the N+3 conventional comparison airplane, after accounting for TMS weight, power, and drag penalties on this 800 mile mission. Parallel hybridization by itself saves 3.5%, whereas thermal management requirements offset that by 0.2%. This is a significantly lower penalty than found in the Chapter 5 series hybrid study. The lower thermal management penalty is partly attributable to the higher

flight speed of the transport aircraft, which enables some Meredith effect cooling drag recovery. The parallel architecture also reduces the fraction of propulsive power that is electrified, which reduces the relative size of the heat rejection problem.

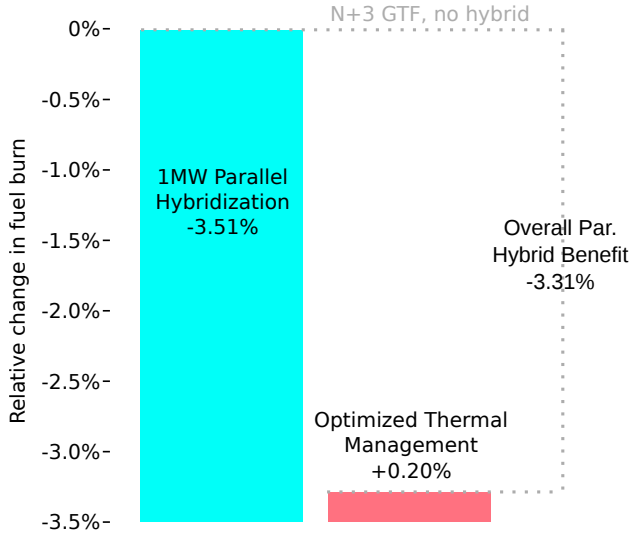


Figure 7.7: Thermal management costs only slightly offset the parallel hybrid fuel burn benefit

		Unit	Lower	Upper
minimize	Fuel burn	lb		
with respect to				
Mission parameters	Cruise altitude	ft	25000	35000
	W_{TO}	lb	110231[a]	174170[b]
	W_b	lb	2500	12500
Battery TMS parameters	Chiller power rating	kW	0.1	50
	Battery HX n_{wide}		2	1500
	Battery HX n_{long}		3	75
	Coolant pump power rating	kW	0.1	5
	Coolant hose diameter	in	0.5	2
Motor TMS parameters	Motor HX n_{wide}		50	1500
	Motor HX n_{tall}		10	25
	Motor HX n_{long}		3	75
	Fault prot. HX n_{long}		1	4
	Motor HX duct nozzle area	in^2	5	60
	Coolant pump power rating	kW	0.1	5
	Coolant hose diameter	in	0.5	2
Control parameters	Battery HX duct nozzle area	in^2	$\mathbf{5}$	$\mathbf{150}$
	Hybrid motor power	kW	$\mathbf{0.0}$	$\mathbf{1000.0}$[c]
	Chiller bypass (β)		$\mathbf{0}$	$\mathbf{1}$
subject to				
Design constraints	Battery HX cross-sectional area	in^2		300
	Motor HX width	in		23.6[a]
	Motor HX height	in		11.8[a]
	Motor HX area	in^2		300
Mission constraints	$\text{SOC}_{\text{final}}$		0.05	
	Payload weight	lb	44092[a]	
	$\boldsymbol{T_m}$	° C		90
	$\boldsymbol{T_b}$	° C		45[d]
	$\boldsymbol{T_{b,\text{final}}}$	° C		35[d]
	Fault prot. coolant inlet temp.	° C		50
	Battery coolant pump P/P_{rated}			1.0
	Motor coolant pump P/P_{rated}			1.0

[a]Non-round number because constraint specified in non-SI units
[b]MTOW for 737-800
[c]Limited to 300kW on descent due to compressor surge margin
[d]During temperature sensitivity studies these values may vary

Table 7.3: Parallel hybrid optimization problem specification

		B738 Baseline	N+3 GTF	N+3 Hybrid (No TMS)	N+3 Hybrid (optimized, w/TMS)
Range	nmi	800	800	800	800
Battery spec energy	Wh/kg			400	400
Chiller spec power	W/kg				200
Design Variables					
Cruise altitude	ft	35000	35000	35000	35000
Takeoff weight	lbm	148232	144914	159299	159643
Battery weight	lbm			6901	6969
Battery TMS					
Chiller power rating	kW				2.21
Battery HX cells wide					204.81
Battery HX cells long					13.64
Coolant pump power	kW				0.77
Hose diameter	inch				1.29
Motor TMS					
Motor HX cells wide					413.22
Motor HX cells long					16.32
Motor HX cells tall					17.04
Fault prot. HX cells long					3.72
Nozzle area	in^2				15.61
Coolant pump power	kW				0.13
Hose diameter	inch				0.73
Objective					
Fuel burn	lbm	11830.15	8511.33	8212.12	8229.13
Constraints					
Final SOC				0.05	0.05
Battery HX width	inch				11.71
Battery HX XS area	in^2				300.00
Motor HX width	inch				23.62
Motor HX height	inch				11.81
Motor HX XS area	in^2				206.30
Payload	lbm	44092	44092	44092	44092
Outputs					
Chiller weight	lbm				24.41
Battery HX weight	lbm				15.80
Batt. TMS pump weight	lbm				3.78
Batt. TMS hose weight	lbm				30.47
Motor HX weight	lbm				13.21
Motor TMS pump weight	lbm				0.65
Motor TMS hose weight	lbm				4.92

Table 7.4: Optimized design and performance of the hybrid single aisle transport, plus comparison airplanes

7.4 Sensitivity Studies

The following subsections explore the effect of several important parameters on TMS design and airplane performance, namely:

- Range
- Specific power of the chiller
- Specific energy of the battery
- Temperature limits of the battery

Each sensitivity study includes two types of figures. *Sweep* plots show the effect of the parameter in question on scalar design and performance quantities, with the sensitivity variable on the x axis. *Trajectory* plots illustrate the effect of the parameter on dynamic parameters like component temperatures and control parameters across the duration of the mission, with distance travelled on the x axis.

7.4.1 Effect of range

Figure 7.8 illustrates how the optimal parallel hybrid airplane design changes as the mission lengthens. Fuel burn increases more or less linearly with the mission length, as expected. However, we see jumps in some of the design parameters between 720 and 740 nautical miles. At this range, the optimum cruise altitude jumps from around 30,000 feet to the upper bound at 35,000 feet. The higher altitude reduces cruise drag, so the aircraft can reduce the size of the battery and reduce takeoff weight. The colder temperatures cause a jump downward in the size of chiller and the size/weight of the battery heat exchanger.

We also see a slight, generally linear trend in coolant pump size and hose diameter. This is a very subtle tradeoff. As the mission gets longer, the penalty to carry used batteries gets larger, which penalizes anything that uses electrical power. At short ranges, a larger pump that can

overcome the pressure losses in smaller, lighter hoses is favored. At long range, the electricity consumption of the larger pump is penalized and a larger hose with less pressure drop is favored.

Figures 7.9 and 7.10 show the trajectories of state variables and control parameters. There are no major discontinuities in the trajectories as range increases, but subtle differences in the duct nozzle control scheduling can be observed.

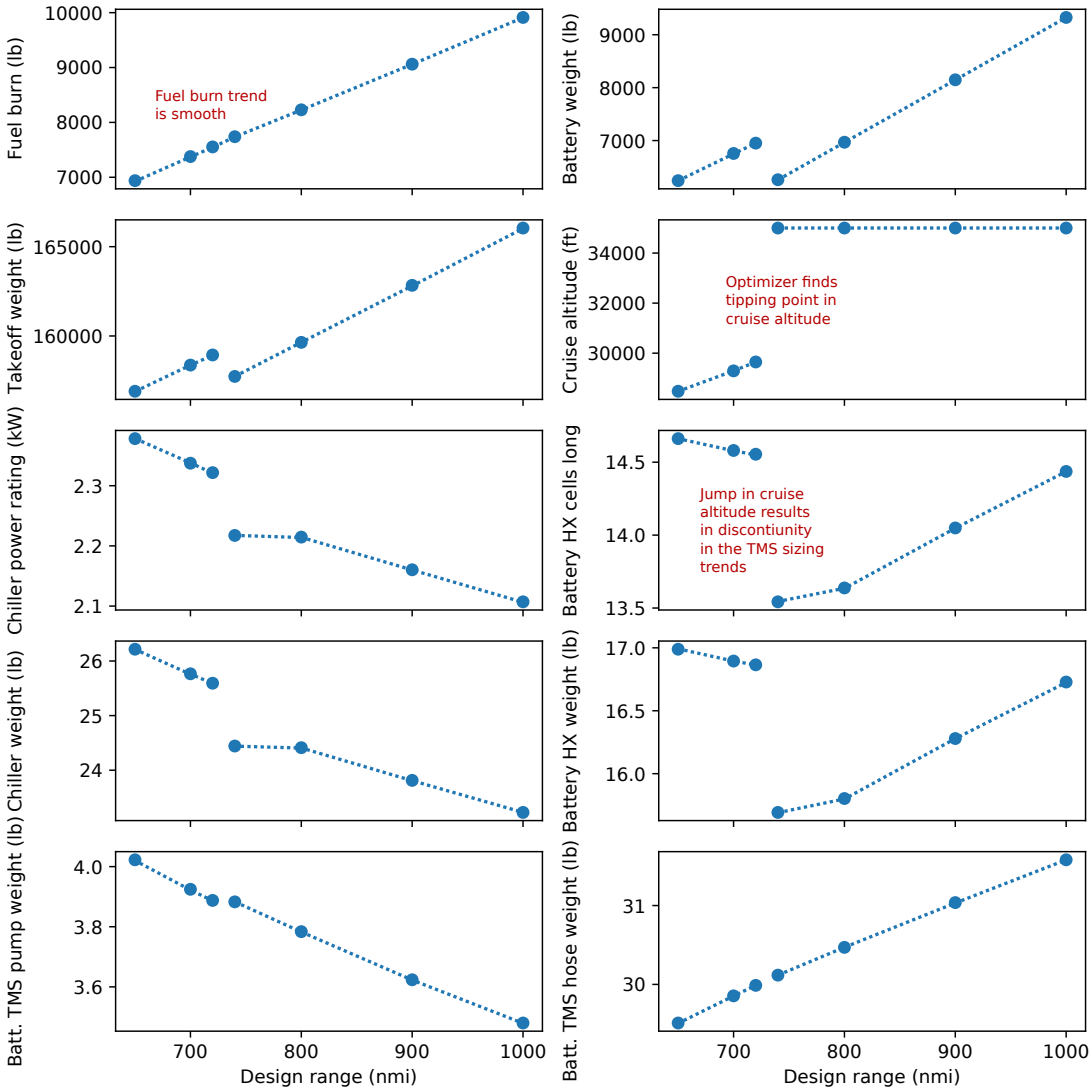


Figure 7.8: Effect of mission range on optimal design

Range	nmi	650	700	720	800	900	1000
Battery spec energy	Wh/kg	400	400	400	400	400	400
Chiller spec power	W/kg	200	200	200	200	200	200
Design Variables							
Cruise altitude	ft	28479	29293	29649	35000	35000	35000
Takeoff weight	lbm	156906	158371	158937	159643	162834	166043
Battery weight	lbm	6242.8	6756.3	6951.5	6968.8	8148.2	9326.4
Battery TMS							
Chiller power rating	kW	2.38	2.34	2.32	2.21	2.16	2.11
Battery HX cells wide		204.81	204.81	204.81	204.81	204.81	204.81
Battery HX cells long		14.66	14.58	14.55	13.64	14.05	14.44
Coolant pump power	kW	0.82	0.80	0.79	0.77	0.74	0.71
Hose diameter	inch	1.27	1.27	1.28	1.29	1.30	1.31
Motor TMS							
Motor HX cells wide		413.22	413.22	413.22	413.22	413.22	413.22
Motor HX cells long		17.27	17.14	17.08	16.32	16.34	16.35
Motor HX cells tall		17.04	17.04	17.04	17.04	17.04	17.04
Fault prot. HX cells long		3.78	3.78	3.78	3.72	3.72	3.72
Nozzle area	in^2	15.21	15.24	15.26	15.61	15.61	15.61
Coolant pump power	kW	0.14	0.14	0.14	0.13	0.13	0.12
Hose diameter	inch	0.72	0.72	0.73	0.73	0.74	0.74
Objective							
Fuel burn	lbm	6938.18	7377.20	7553.04	8229.13	9061.01	9912.62
Constraints							
Final SOC		0.05	0.05	0.05	0.05	0.05	0.05
Battery HX width	inch	11.71	11.71	11.71	11.71	11.71	11.71
Battery HX XS area	in^2	300.00	300.00	300.00	300.00	300.00	300.00
Motor HX width	inch	23.62	23.62	23.62	23.62	23.62	23.62
Motor HX height	inch	11.81	11.81	11.81	11.81	11.81	11.81
Motor HX XS area	in^2	206.30	206.30	206.30	206.30	206.30	206.30
Payload	lbm	44092.45	44092.45	44092.45	44092.45	44092.45	44092.45
Outputs							
Chiller weight	lbm	26.22	25.77	25.59	24.41	23.81	23.23
Battery HX weight	lbm	16.99	16.89	16.86	15.80	16.28	16.73
Batt. TMS pump weight	lbm	4.02	3.92	3.89	3.78	3.62	3.48
Batt. TMS hose weight	lbm	29.51	29.85	29.99	30.47	31.04	31.58
Motor HX weight	lbm	13.98	13.88	13.83	13.21	13.23	13.24
Motor TMS pump weight	lbm	0.69	0.67	0.67	0.65	0.63	0.61
Motor TMS hose weight	lbm	4.77	4.82	4.85	4.92	5.02	5.10

Table 7.5: Sensitivity of design and performance with respect to range

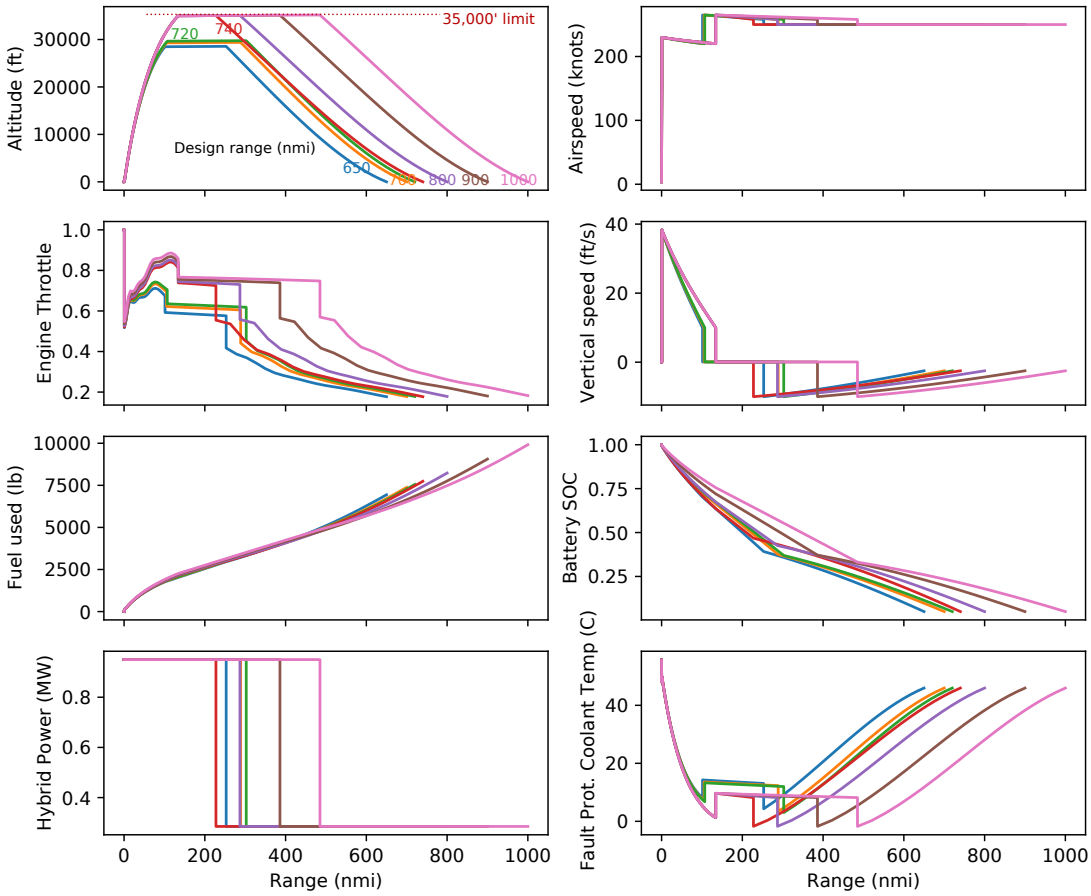


Figure 7.9: Optimized flight profile and control parameters — various ranges

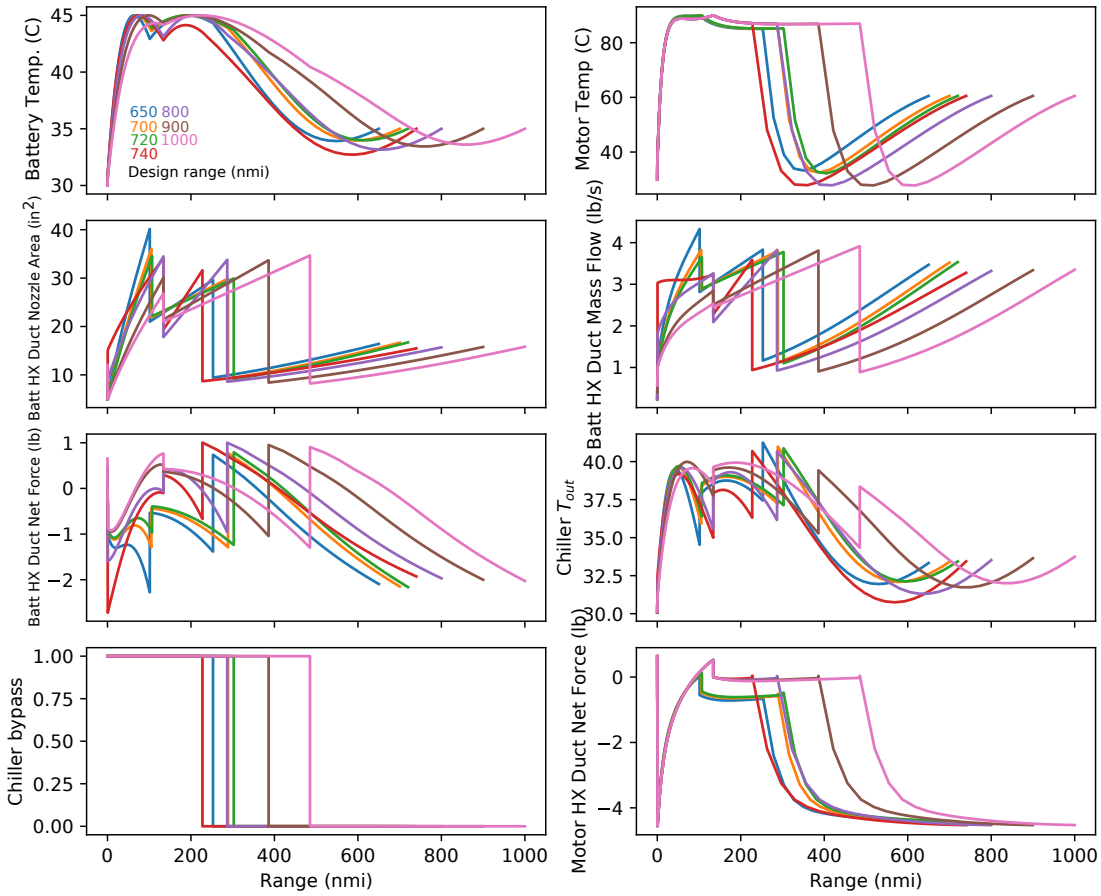


Figure 7.10: Optimized thermal trajectory and control parameters — various ranges

7.4.2 Effect of chiller specific power

For this TMS trade study, a key technological parameter is the specific power of the chiller (in other words, how heavy it is). Lower specific power will favor using a smaller chiller and larger heat exchanger. Figure 7.11 shows the effect of specific power on the TMS design for two mission profiles. The "chilldown" scenario requires the airplane to land with a battery temperature of 35° C or less (to enable the airplane to take off again quickly after unloading and loading). The "no chilldown" requirement imposes a flat 45° C limit on battery temperature across the whole mission. For both scenarios, fuel burn decreases sharply as specific power increases to about 100 W/kg. Beyond 200 W/kg (which is my best estimate of the state of the art), the returns diminish greatly. Therefore, it is likely that dramatic improvements in chiller weights are not on the critical path to hybrid propulsion for this type of mission. However, for an electric aircraft at lower altitude (e.g. eVTOL), this parameter may be more significant.

For all specific powers, the "no chilldown" scenario burns less fuel and uses a smaller chiller than for the "chilldown" scenario. The higher the chiller specific power (the lighter the chiller), the lower the fuel burn penalty attributable to a chilldown requirement. Figures 7.12 and 7.13 show trajectories and control parameters for the chilldown and no-chilldown scenarios, respectively. As the specific power increases, the TMS manages to keep the battery temperature more consistent across the mission. At low specific power, the optimizer does not provide very much chiller capacity, so the optimizer has to cool down the battery at altitude in order to "bank" enough cold thermal mass for the trip down.

Range	nmi	800	800	800	800	800	800
Battery spec energy	Wh/kg	400	400	400	400	400	400
Chiller spec power	W/kg	10	20	50	100	200	400
Design Variables							
Cruise altitude	ft	35000	35000	35000	35000	35000	35000
Takeoff weight	lbm	160149	159913	159748	159681	159643	159621
Battery weight	lbm	6958.5	6960.6	6964.0	6966.5	6968.8	6970.3
Battery TMS							
Chiller power rating	kW	1.21	1.41	1.74	1.98	2.21	2.37
Battery HX cells wide		204.81	204.81	204.81	204.81	204.81	204.81
Battery HX cells long		21.86	19.02	16.04	14.57	13.64	13.24
Coolant pump power	kW	0.71	0.73	0.75	0.76	0.77	0.78
Hose diameter	inch	1.30	1.29	1.29	1.29	1.29	1.29
Motor TMS							
Motor HX cells wide		413.22	413.22	413.22	413.22	413.22	413.22
Motor HX cells long		16.31	16.32	16.32	16.32	16.32	16.32
Motor HX cells tall		17.04	17.04	17.04	17.04	17.04	17.04
Fault prot. HX cells long		3.72	3.72	3.72	3.72	3.72	3.72
Nozzle area	in^2	15.62	15.61	15.61	15.61	15.61	15.61
Coolant pump power	kW	0.13	0.13	0.13	0.13	0.13	0.13
Hose diameter	inch	0.74	0.73	0.73	0.73	0.73	0.73
Objective							
Fuel burn	lbm	8251.63	8241.23	8233.77	8230.81	8229.13	8228.22
Constraints							
Final SOC		0.05	0.05	0.05	0.05	0.05	0.05
Battery HX width	inch	11.71	11.71	11.71	11.71	11.71	11.71
Battery HX XS area	in^2	300.00	300.00	300.00	300.00	300.00	300.00
Motor HX width	inch	23.62	23.62	23.62	23.62	23.62	23.62
Motor HX height	inch	11.81	11.81	11.81	11.81	11.81	11.81
Motor HX XS area	in^2	206.30	206.30	206.30	206.30	206.30	206.30
Payload	lbm	44092.45	44092.45	44092.45	44092.45	44092.45	44092.45
Outputs							
Chiller weight	lbm	266.64	155.41	76.52	43.73	24.41	13.05
Battery HX weight	lbm	25.33	22.04	18.59	16.88	15.80	15.34
Batt. TMS pump weight	lbm	3.47	3.58	3.69	3.74	3.78	3.80
Batt. TMS hose weight	lbm	31.05	30.77	30.58	30.51	30.47	30.45
Motor HX weight	lbm	13.21	13.21	13.21	13.21	13.21	13.21
Motor TMS pump weight	lbm	0.63	0.64	0.65	0.65	0.65	0.65
Motor TMS hose weight	lbm	5.02	4.97	4.94	4.93	4.92	4.92

Table 7.6: Sensitivity of design and performance with respect to chiller spec. power — 35 C chilldown

Range	nmi	800	800	800	800	800	800
Battery spec energy	Wh/kg	400	400	400	400	400	400
Chiller spec power	W/kg	10	20	50	100	200	400
Design Variables							
Cruise altitude	ft	35000	35000	35000	35000	35000	35000
Takeoff weight	lbm	159804	159718	159657	159627	159614	159596
Battery weight	lbm	6951.2	6952.7	6955.9	6958.6	6963.0	6963.4
Battery TMS							
Chiller power rating	kW	0.50	0.65	0.95	1.21	1.59	1.63
Battery HX cells wide		204.81	204.81	204.81	204.81	204.81	204.81
Battery HX cells long		18.87	17.10	14.88	13.69	12.61	12.41
Coolant pump power	kW	0.72	0.74	0.76	0.77	0.78	0.78
Hose diameter	inch	1.30	1.29	1.29	1.29	1.29	1.28
Motor TMS							
Motor HX cells wide		413.22	413.22	413.22	413.22	413.22	413.22
Motor HX cells long		16.32	16.32	16.32	16.32	16.32	16.32
Motor HX cells tall		17.04	17.04	17.04	17.04	17.04	17.04
Fault prot. HX cells long		3.72	3.72	3.72	3.72	3.72	3.72
Nozzle area	in^2	15.61	15.61	15.61	15.61	15.61	15.61
Coolant pump power	kW	0.13	0.13	0.13	0.13	0.13	0.13
Hose diameter	inch	0.74	0.73	0.73	0.73	0.73	0.73
Objective							
Fuel burn	lbm	8238.95	8234.43	8230.75	8229.04	8227.66	8227.01
Constraints							
Final SOC		0.05	0.05	0.05	0.05	0.05	0.05
Battery HX width	inch	11.71	11.71	11.71	11.71	11.71	11.71
Battery HX XS area	in^2	300.00	300.00	300.00	300.00	300.00	300.00
Motor HX width	inch	23.62	23.62	23.62	23.62	23.62	23.62
Motor HX height	inch	11.81	11.81	11.81	11.81	11.81	11.81
Motor HX XS area	in^2	206.30	206.30	206.30	206.30	206.30	206.30
Payload	lbm	44092.45	44092.45	44092.45	44092.45	44092.45	44092.45
Outputs							
Chiller weight	lbm	111.14	71.36	41.99	26.71	17.52	9.01
Battery HX weight	lbm	21.86	19.82	17.24	15.86	14.62	14.38
Batt. TMS pump weight	lbm	3.52	3.62	3.72	3.78	3.83	3.84
Batt. TMS hose weight	lbm	31.02	30.75	30.55	30.48	30.43	30.41
Motor HX weight	lbm	13.21	13.21	13.21	13.21	13.21	13.21
Motor TMS pump weight	lbm	0.63	0.64	0.65	0.65	0.65	0.65
Motor TMS hose weight	lbm	5.01	4.97	4.94	4.93	4.92	4.91

Table 7.7: Sensitivity of design and performance with respect to chiller spec. power — no chill-down

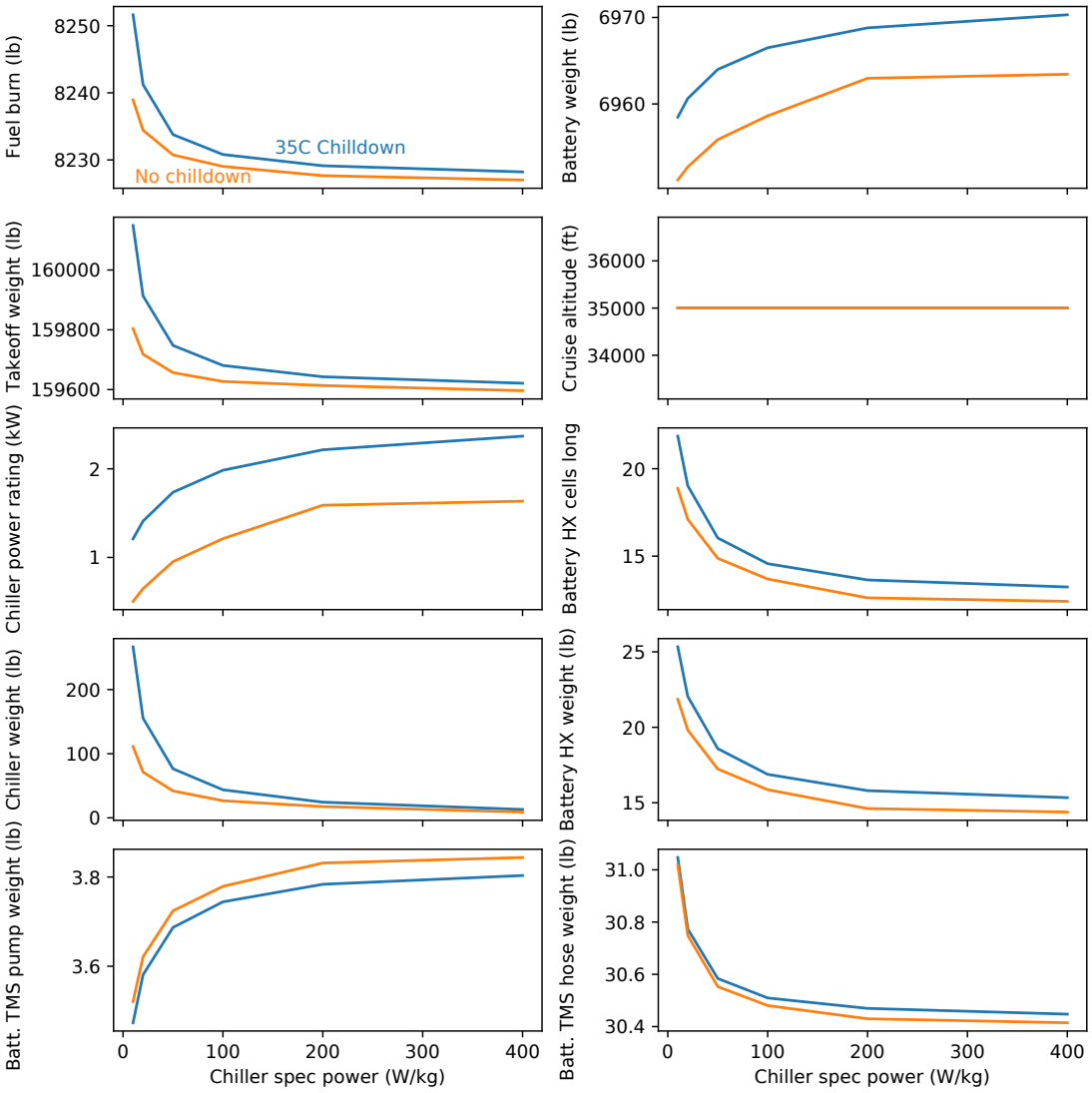


Figure 7.11: Effect of chiller spec. energy on optimal design

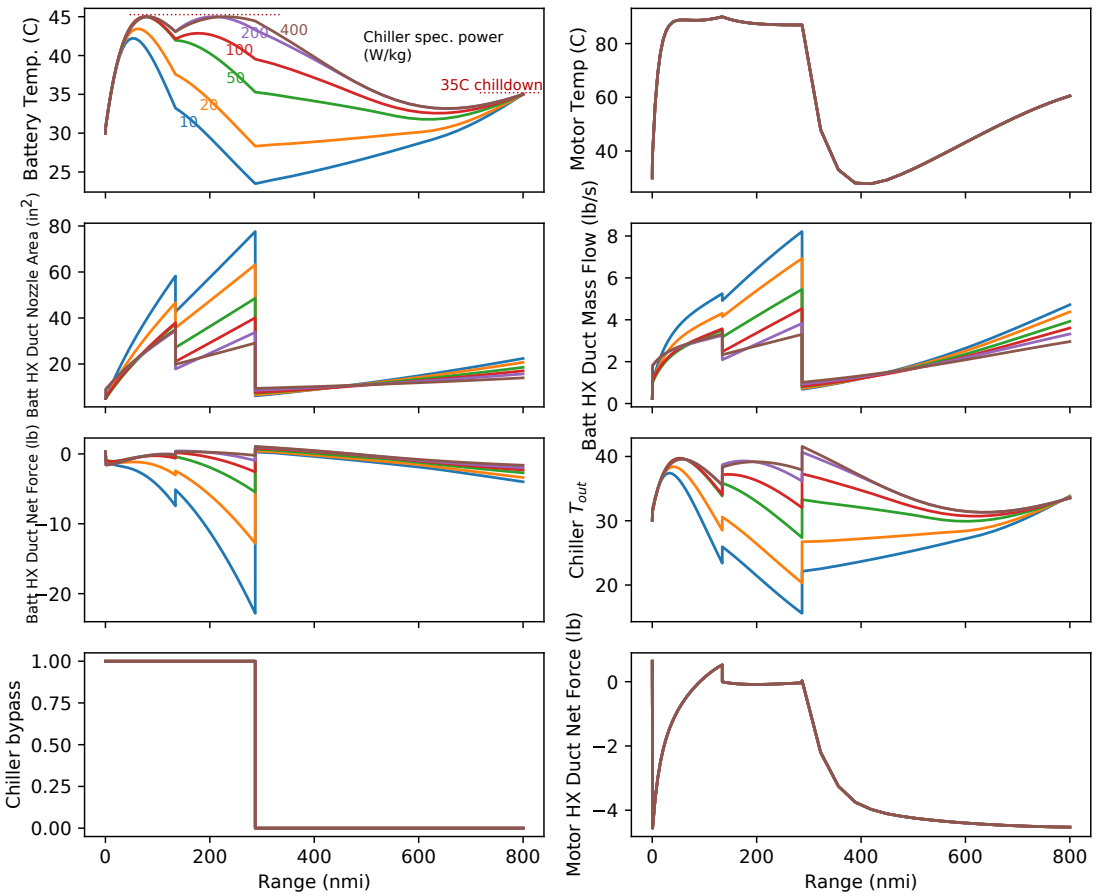


Figure 7.12: Optimized thermal trajectory and control parameters — various chiller spec. powers, with chilldown

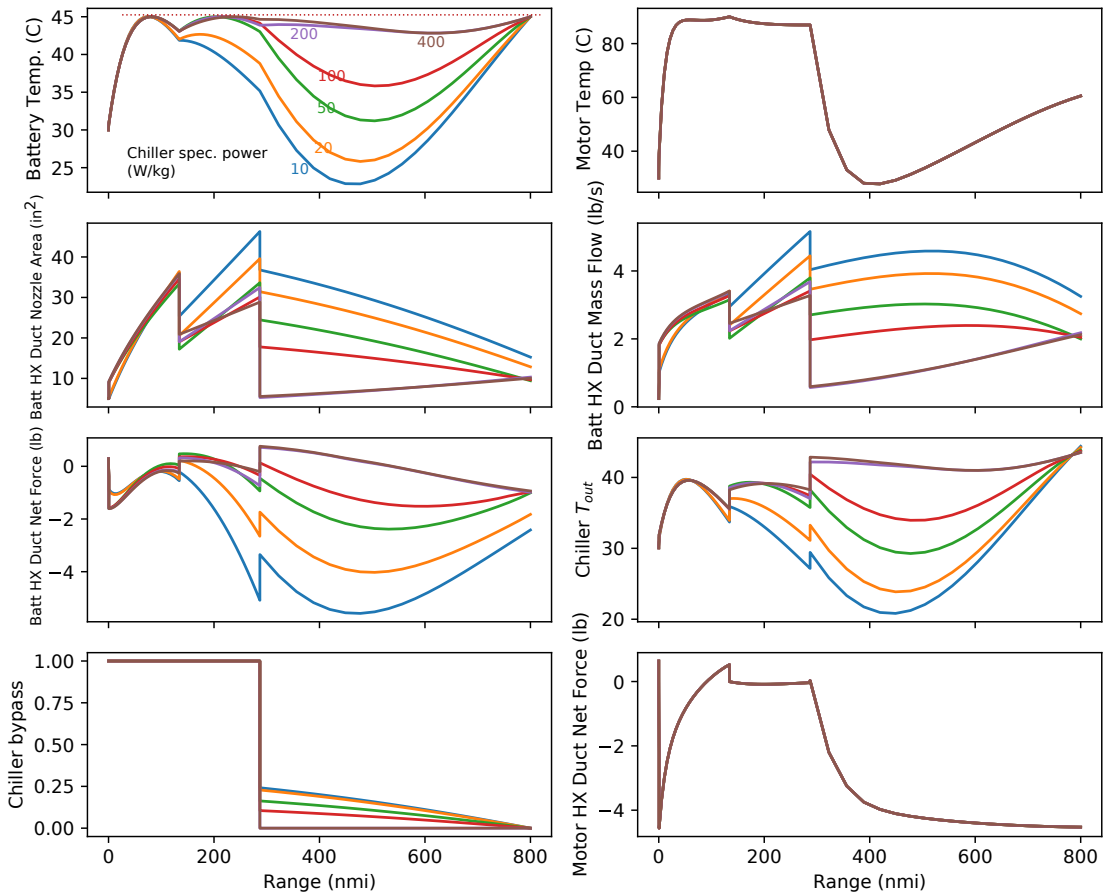


Figure 7.13: Optimized thermal trajectory and control parameters — various chiller spec. powers, without chilldown

7.4.3 Effect of battery specific energy

For this 2030+ trade study, the technological performance of the battery (in particular, specific energy) is highly uncertain. Figure 7.14 shows the effect of specific energy on aircraft design variables and performance. Because range and specific energy are so tightly linked, I plotted sweep lines for ranges between 700 and 1000 nautical miles and normalized fuel burn by range to collapse the plot. The fuel burn reduction due to battery specific energy improvement is not very large (2.5% fuel burn reduction per 100 Wh/kg improvement). This is likely due to the mild H_P/H_E of this configuration. On the other hand, 2.5% is quite significant compared to the 3.3% fuel burn improvement of the baseline hybrid.

Battery specific energy affects TMS design in diverse ways depending on the range. Figure 7.16 shows the variation in chiller sizing with range and specific energy. For example, from 700 to 900 nmi range, the chiller power is not strongly influenced by e_b, but at 1000 nmi the chiller power increases somewhat with e_b. If e_b is low and range is long, the battery pack will be heavy (and therefore, have a lot of thermal mass) which smooths out thermal transients. On the other hand, battery specific energy does strongly influence the hose/pump design subproblem in a similar way to range. The lighter the battery, the more pump power becomes favored versus hose diameter / weight and we see a linear trend similar to Subsection 7.4.1. Figure 7.15 shows that specific energy does not strongly influence control parameters or states; the family of trajectories for the 800 nmi mission look qualitatively similar.

Range	nmi	800	800	800	800	800
Battery spec energy	Wh/kg	350	375	400	425	450
Chiller spec power	W/kg	200	200	200	200	200
Design Variables						
Cruise altitude	ft	35000	35000	35000	35000	35000
Takeoff weight	lbm	161707	160606	159643	158793	158038
Battery weight	lbm	7963.9	7433.1	6968.8	6559.0	6194.7
Battery TMS						
Chiller power rating	kW	2.18	2.20	2.21	2.22	2.23
Battery HX cells wide		204.81	204.81	204.81	204.81	204.81
Battery HX cells long		13.72	13.66	13.64	13.68	13.75
Coolant pump power	kW	0.72	0.75	0.77	0.79	0.81
Hose diameter	inch	1.30	1.30	1.29	1.28	1.27
Motor TMS						
Motor HX cells wide		413.22	413.22	413.22	413.22	413.22
Motor HX cells long		16.29	16.30	16.32	16.33	16.34
Motor HX cells tall		17.04	17.04	17.04	17.04	17.04
Fault prot. HX cells long		3.72	3.72	3.72	3.72	3.72
Nozzle area	in^2	15.62	15.62	15.61	15.61	15.61
Coolant pump power	kW	0.13	0.13	0.13	0.14	0.14
Hose diameter	inch	0.74	0.74	0.73	0.73	0.72
Objective						
Fuel burn	lbm	8302.01	8262.99	8229.13	8199.47	8173.27
Constraints						
Final SOC		0.05	0.05	0.05	0.05	0.05
Battery HX width	inch	11.71	11.71	11.71	11.71	11.71
Battery HX XS area	in^2	300.00	300.00	300.00	300.00	300.00
Motor HX width	inch	23.62	23.62	23.62	23.62	23.62
Motor HX height	inch	11.81	11.81	11.81	11.81	11.81
Motor HX XS area	in^2	206.30	206.30	206.30	206.30	206.30
Payload	lbm	44092.45	44092.45	44092.45	44092.45	44092.45
Outputs						
Chiller weight	lbm	24.07	24.21	24.41	24.49	24.53
Battery HX weight	lbm	15.90	15.83	15.80	15.85	15.93
Batt. TMS pump weight	lbm	3.55	3.67	3.78	3.89	3.99
Batt. TMS hose weight	lbm	31.37	30.90	30.47	30.08	29.72
Motor HX weight	lbm	13.19	13.20	13.21	13.22	13.23
Motor TMS pump weight	lbm	0.61	0.63	0.65	0.67	0.68
Motor TMS hose weight	lbm	5.07	4.99	4.92	4.86	4.80

Table 7.8: Sensitivity of design and performance with respect to battery specific energy

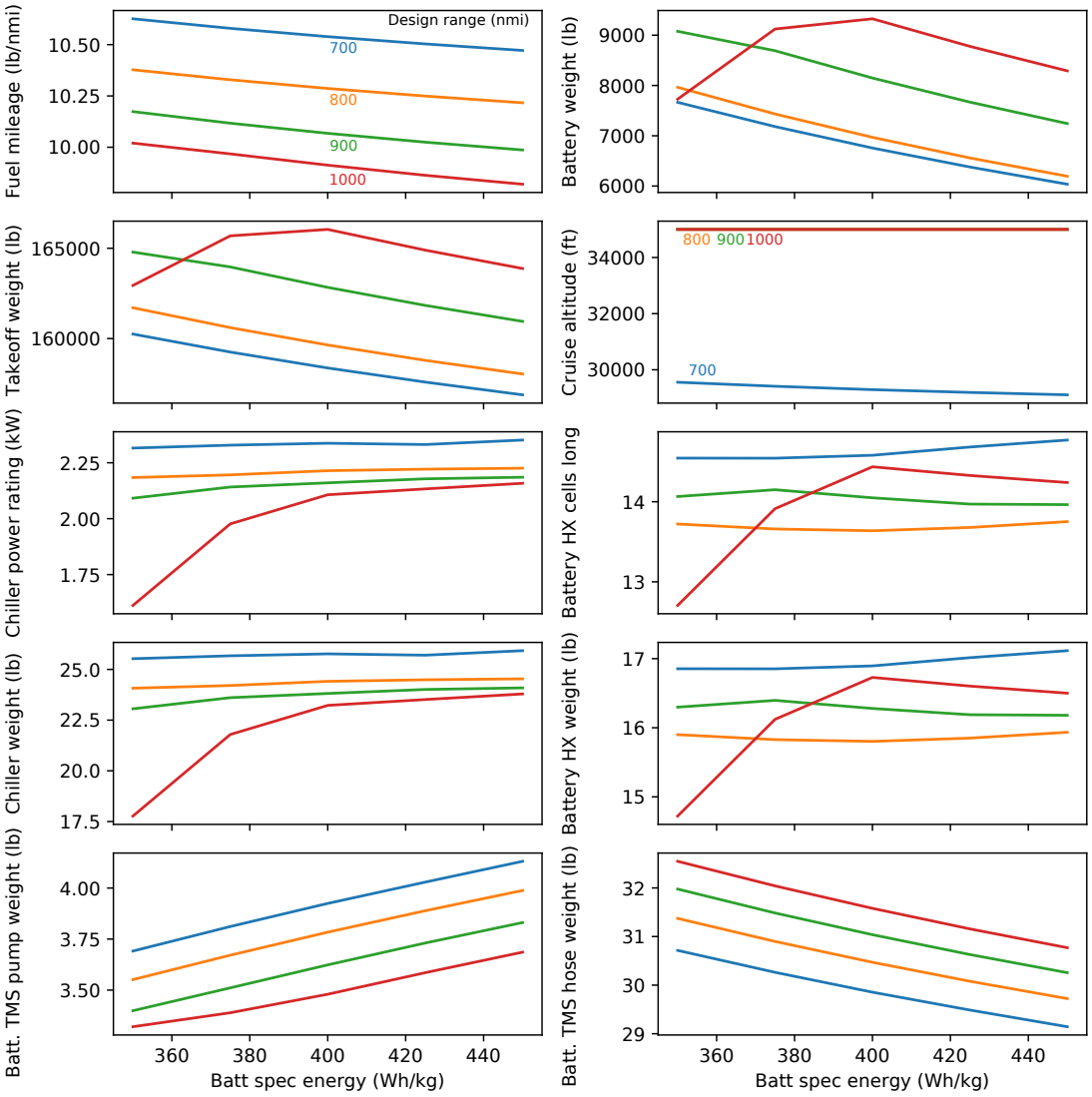


Figure 7.14: Effect of battery specific energy on optimal design at multiple flight lengths

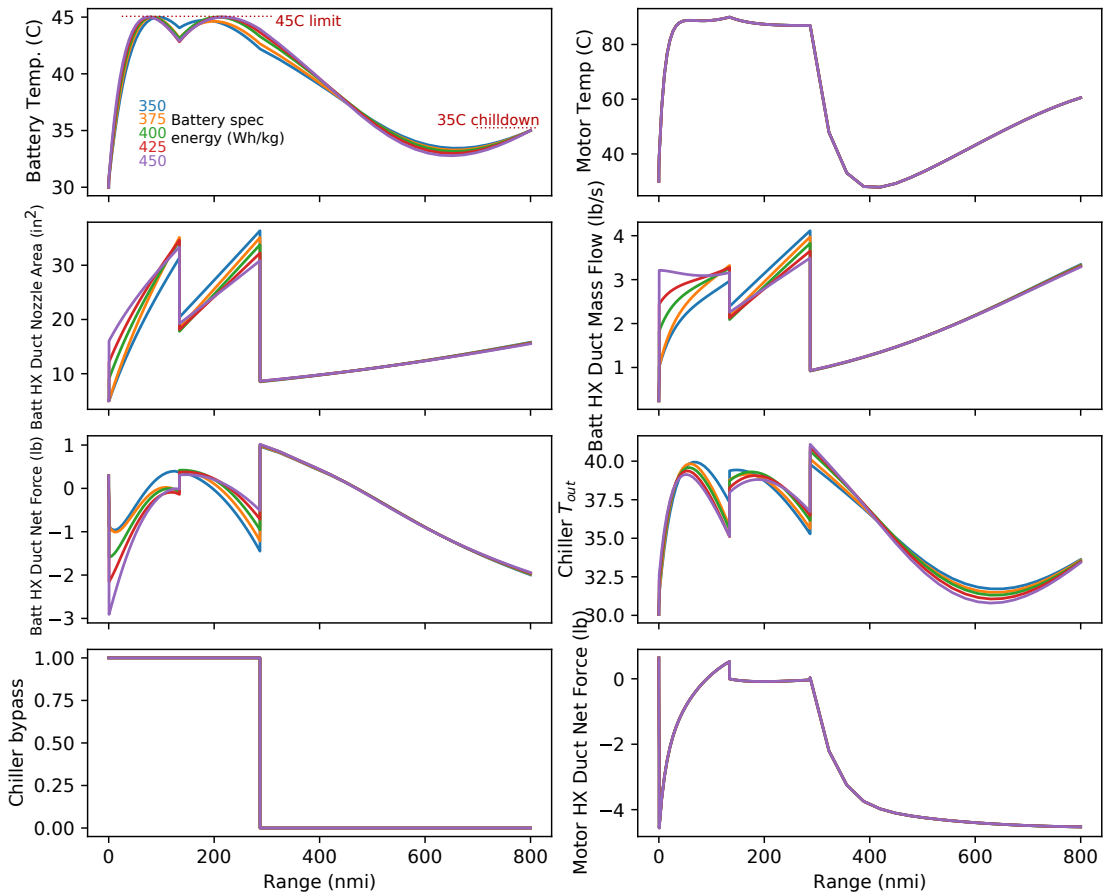


Figure 7.15: Optimized thermal trajectory and control parameters — various battery spec energies, 800 nmi flight

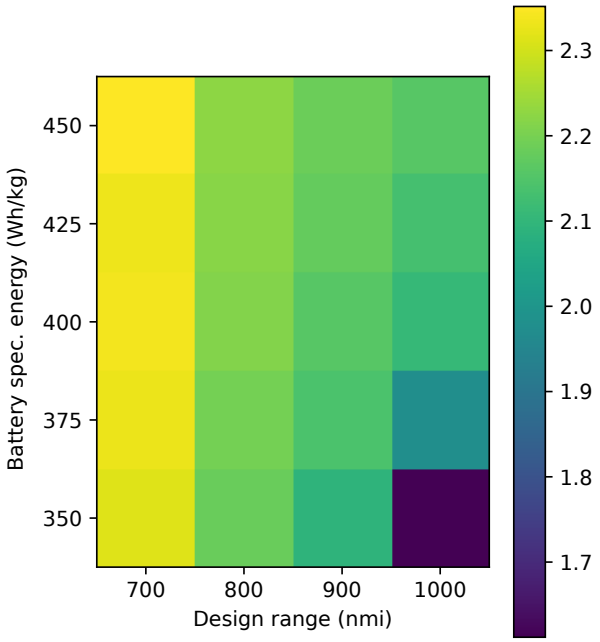


Figure 7.16: Optimal chiller size variation with flight length and battery spec. energy

7.4.4 Effect of battery thermal limitations

In principle, there is a design tradeoff between the electrical performance of a battery cell and its thermal stability (the degree to which this is actually dial-able is uncertain). Therefore, it is useful to know how much airplane-level performance improvement would result from relaxing thermal management requirements for the battery. Figure 7.17 shows the effect of the battery temperature constraint on the aircraft design and performance. Figure 7.18 shows the trajectory with states and control parameters, and illustrates the variation in the thermal constraint more tangibly.

I found that, at least in this application, fuel burn is only a very weak function of the temperature limit. Increasing the temperature limit from 40 C (cool for a battery) to 60 C (very warm) only reduces fuel burn by about 5 pounds, or 0.05%. One interesting trend is in the use of the chiller as the temperature limit is relaxed. Figure 7.17 shows that the optimizer removes the chiller altogether once the maximum temperature is 50 C or above; these are the only cases in this trade study where the optimizer did not elect to preserve at least some chiller capability. It is likely that lower-altitude aircraft that fly longer in warmer air will be more sensitive to battery operating limits than this concept.

		40 C	**45 C**	**50 C**	**55 C**	**60 C**
Range	nmi	800	800	800	800	800
Battery spec energy	Wh/kg	400	400	400	400	400
Chiller spec power	W/kg	200	200	200	200	200
Design Variables						
Cruise altitude	ft	35000	35000	35000	35000	35000
Takeoff weight	lbm	159626	159614	159549	159543	159538
Battery weight	lbm	6963.6	6963.0	6947.8	6948.0	6947.8
Battery TMS						
Chiller power rating	kW	1.68	1.59	0.10	0.10	0.10
Battery HX cells wide		204.81	204.81	204.81	192.46	162.75
Battery HX cells long		15.96	12.61	12.63	11.12	10.86
Coolant pump power	kW	0.76	0.78	0.78	0.79	0.78
Hose diameter	inch	1.29	1.29	1.28	1.28	1.28
Motor TMS						
Motor HX cells wide		413.22	413.22	413.22	413.22	413.22
Motor HX cells long		16.32	16.32	16.32	16.32	16.32
Motor HX cells tall		17.04	17.04	17.04	17.04	17.04
Fault prot. HX cells long		3.72	3.72	3.72	3.72	3.72
Nozzle area	in^2	15.61	15.61	15.61	15.61	15.61
Coolant pump power	kW	0.13	0.13	0.13	0.13	0.13
Hose diameter	inch	0.73	0.73	0.73	0.73	0.73
Objective						
Fuel burn	lbm	8229.26	8227.66	8225.79	8224.89	8224.44
Constraints						
Final SOC		0.05	0.05	0.05	0.05	0.05
Battery HX width	inch	11.71	11.71	11.71	11.00	9.30
Battery HX XS area	in^2	300.00	300.00	300.00	281.91	238.39
Motor HX width	inch	23.62	23.62	23.62	23.62	23.62
Motor HX height	inch	11.81	11.81	11.81	11.81	11.81
Motor HX XS area	in^2	206.30	206.30	206.30	206.30	206.30
Payload	lbm	44092.45	44092.45	44092.45	44092.45	44092.45
Outputs						
Chiller weight	lbm	18.54	17.52	1.10	1.10	1.10
Battery HX weight	lbm	18.49	14.62	14.64	12.19	10.28
Batt. TMS pump weight	lbm	3.73	3.83	3.83	3.88	3.83
Batt. TMS hose weight	lbm	30.44	30.43	30.42	30.38	30.37
Motor HX weight	lbm	13.21	13.21	13.21	13.21	13.21
Motor TMS pump weight	lbm	0.65	0.65	0.65	0.65	0.66
Motor TMS hose weight	lbm	4.92	4.92	4.91	4.91	4.91

Table 7.9: Sensitivity of design and performance with respect to battery temperature limit

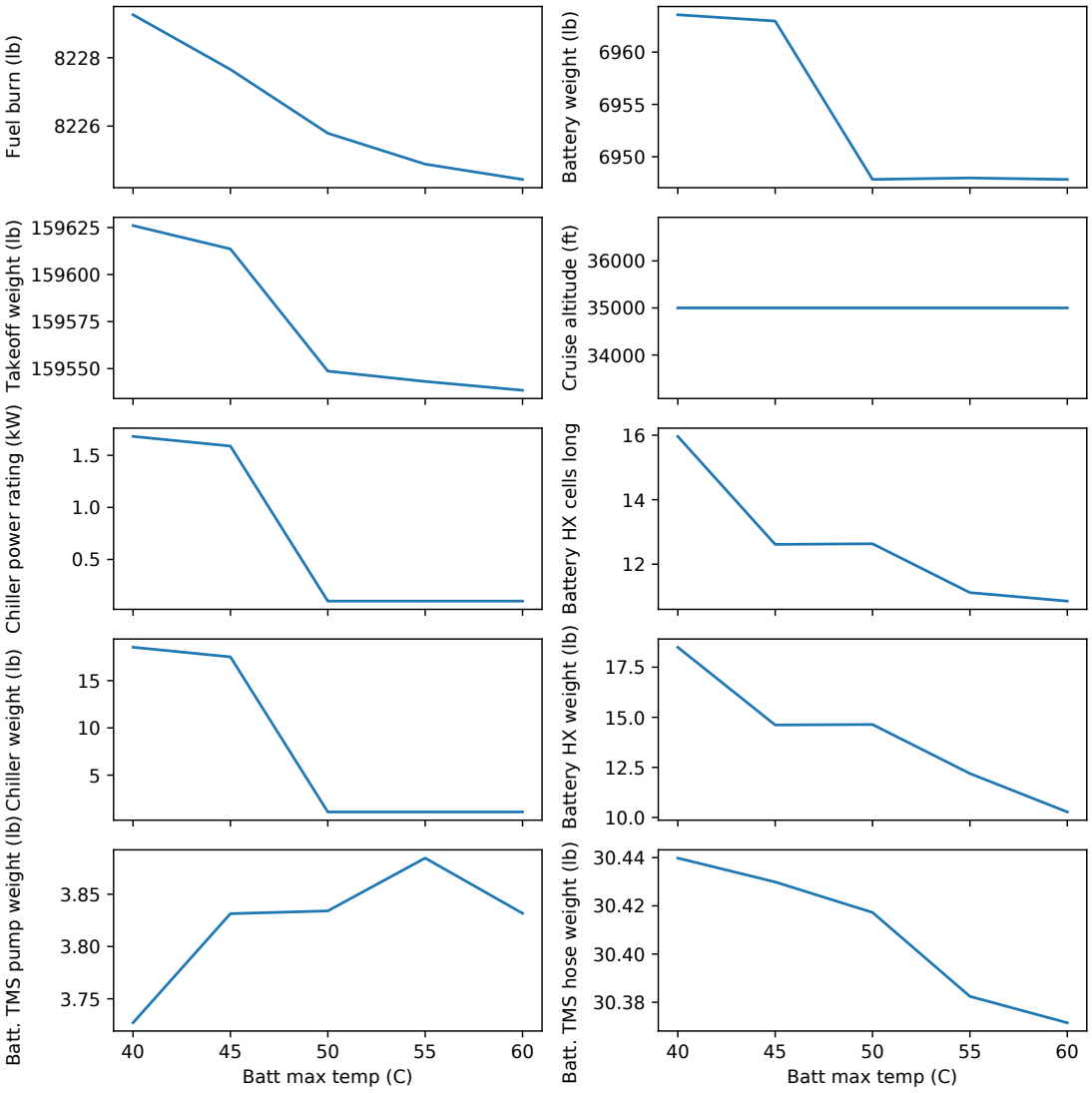


Figure 7.17: Effect of battery temperature limit on optimal design — 800 nmi flight

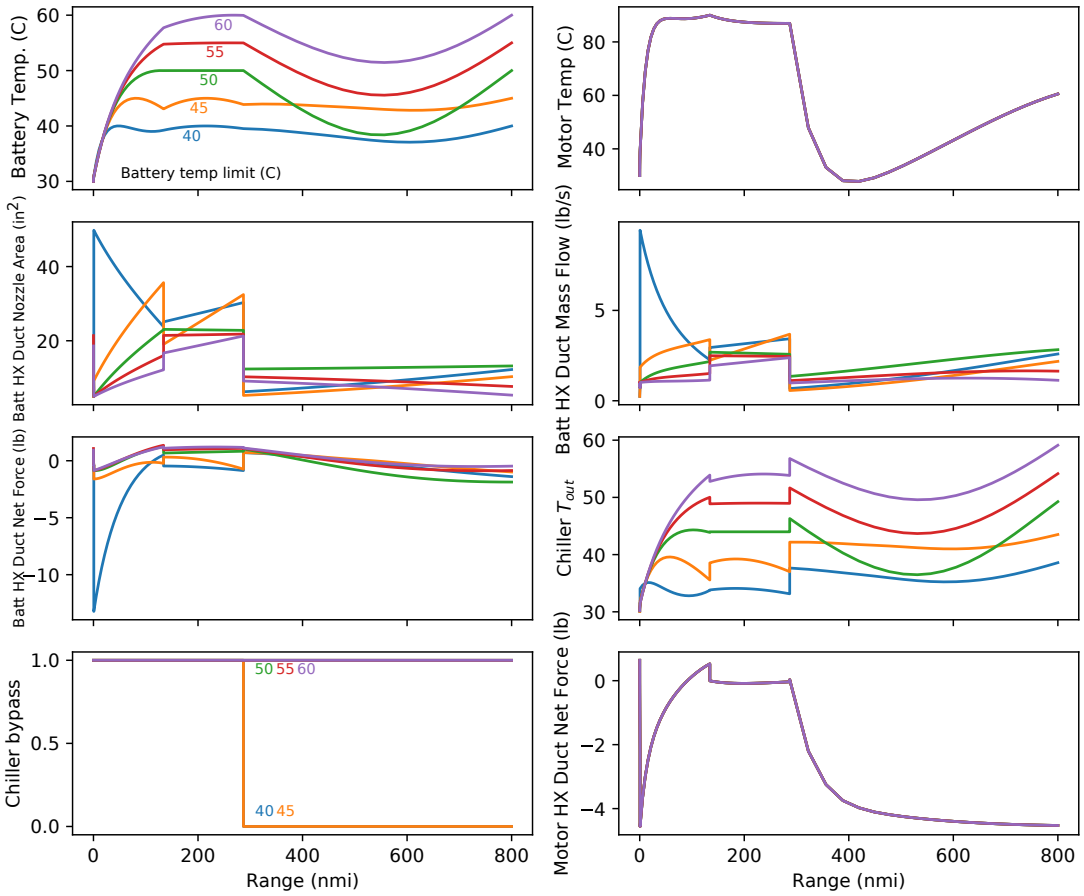


Figure 7.18: Optimized thermal trajectory and control parameters — various battery temp limits

7.4.5 Multimodality

During the process of running the optimization cases for these sensitivity studies, I observed that the IPOPT optimizer was converging cases very tightly to the minima. However, I noticed in some of the sweeps that non-physical discontinuities in fuel burn would appear. Digging deeper, I discovered that this optimization problem may have multiple local minima that are easily found. For example, I noted a false minimum while running the 40 C battery temperature limit case. Figure 7.19 shows two trajectories that meet all the design and control requirements and that IPOPT converged tightly to a local minimum. One of the optima has a significantly larger chiller than the other (4.38 kW versus 1.68 kW). The larger chiller design (detailed in Table 7.9) has an identical motor TMS design but a significantly different battery TMS, and it burns 0.04% more fuel.

During the previously-described sensitivity studies, I found a handful of other similar local minima, which I coaxed to the lower minimum by tightening design variable bounds (generally, the chiller power upper bound). While not unprecedented in my previous OpenConcept airplane studies, this was a qualitatively higher rate than I am used to seeing in conceptual design problems. My guess is that the pseudo-binary bypass behavior of the chiller introduces this multimodality to the problem. An alternative approach would be to use an ODE transcription more suited to optimal control, such as Legendre-Gauss polynomials as implemented in the Dymos software package, which seem to reduce multimodality in trajectory optimization problems.

		Larger-chiller local min	Smaller-chiller local min
Range	nmi	800	800
Battery spec energy	Wh/kg	400	400
Chiller spec power	W/kg	200	200
Design Variables			
Cruise altitude	ft	35000	35000
Takeoff weight	lbm	159756	159626
Battery weight	lbm	7000.3	6963.6
Battery TMS			
Chiller power rating	kW	4.38	1.68
Battery HX cells wide		204.81	204.81
Battery HX cells long		13.46	15.96
Coolant pump power	kW	0.78	0.76
Hose diameter	inch	1.28	1.29
Motor TMS			
Motor HX cells wide		413.22	413.22
Motor HX cells long		16.31	16.32
Motor HX cells tall		17.04	17.04
Fault prot. HX cells long		3.72	3.72
Nozzle area	in^2	15.61	15.61
Coolant pump power	kW	0.13	0.13
Hose diameter	inch	0.73	0.73
Objective			
Fuel burn	lbm	8232.38	8229.26
Constraints			
Final SOC		0.05	0.05
Battery HX width	inch	11.71	11.71
Battery HX XS area	in^2	300.00	300.00
Motor HX width	inch	23.62	23.62
Motor HX height	inch	11.81	11.81
Motor HX XS area	in^2	206.30	206.30
Payload	lbm	44092.45	44092.45
Outputs			
Chiller weight	lbm	48.27	18.54
Battery HX weight	lbm	15.59	18.49
Batt. TMS pump weight	lbm	3.81	3.73
Batt. TMS hose weight	lbm	30.41	30.44
Motor HX weight	lbm	13.21	13.21
Motor TMS pump weight	lbm	0.65	0.65
Motor TMS hose weight	lbm	4.91	4.92

Table 7.10: Comparison of two local minima — 800 nmi range, 40° C battery temp limit

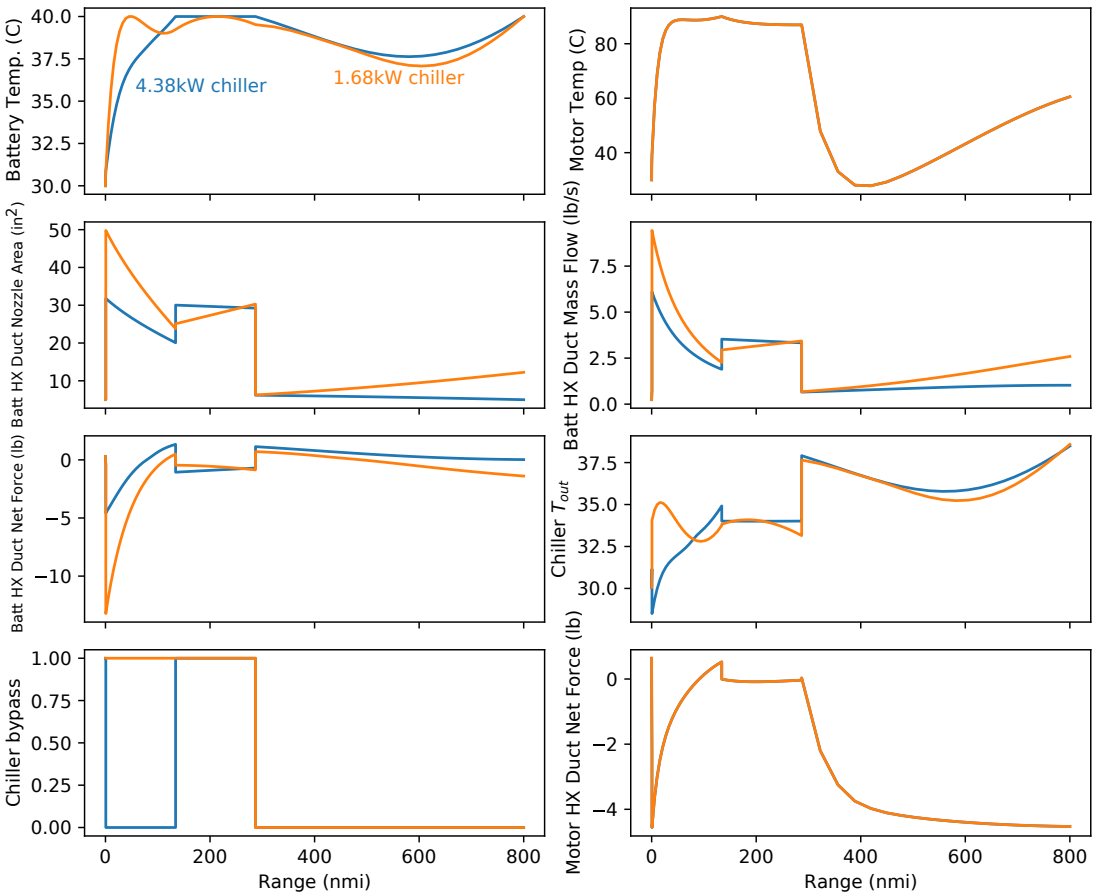


Figure 7.19: Trajectories of two local minima — 800 nmi, 40 C temperature limit

7.5 Concluding Remarks

In this section, I demonstrated a conceptual MDO trade study for a parallel hybrid single aisle transport airplane subject to thermal constraints. To my knowledge, this is the first time that a parallel hybrid transport aircraft has been studied using gradient-based optimization while also considering thermal management. Compared to a conventional (non-hybrid) comparison airplane, the hybrid propulsion architecture saves approximately 3.3% on an 800 nmi mission. Thermal management of the electrical components is responsible for about 0.2% fuel burn penalty.

I performed dozens of optimization cases to generate sensitivities with respect to key parameters. The TMS and propulsion design is a moderate function of range and chiller specific power,

but only a weak function of battery specific energy. This airplane is not as sensitive to battery specific energy as all-electric designs, but it also has much more moderate fuel burn reductions compared to all-electric architectures. SOA chiller weight is probably adequate for the purposes of this short, parallel hybrid mission, and further improvements would result in diminishing returns to fuel burn. Finally, the transport aircraft mission does not benefit very much from running the batteries a bit warmer, only saving a few pounds of fuel over a 800 mile mission. The trends illustrated here are particular to this architecture, engine cycle, and mission and are likely not applicable to other classes of hybrid and electric aircraft, especially lower-altitude, high-power eVTOL aircraft.

The combined propulsion and thermal management conceptual design problem appears to introduce subtle local minima at a higher than normal rate. Using an alternative ODE transcription or possibly using multistart optimization could be good approaches to mitigating this finding.

This chapter consists of entirely new and unpublished material.

CHAPTER 8

Development of a General Spatial Packaging Constraint for Shape Optimization

Aerodynamic shape optimization (ASO) has become an increasingly popular tool for aircraft design [225, 226]. MDO practitioners have various choices for aerodynamic models to suit their requirements, ranging from low-fidelity panel codes to RANS simulations. The structural models in aerostructural problems have also evolved from beam models to full wingbox finite element models [147, 154]. Recent interest in boundary layer ingestion has led to close coupling of aerodynamic and propulsion models in aeropropulsive design optimization [157].

While the "hard" aerosciences have been incorporated into increasingly high-fidelity MDO frameworks, one crucial aspect of aircraft design has been somewhat neglected: *spatial integration*. Spatial integration requires that all crew, passengers, payloads, systems components, and energy storage fit within the aircraft outer mold line (OML), and that objects do not overlap with each other.

Spatial integration constraints directly drive important design features found on aircraft today; from wing-to-body fairing depth on commercial aircraft, to carefully-shaped actuator "bumps" on the wings of stealth military aircraft. Electric and hybrid-electric propulsion concepts force hard decisions about where to locate hundreds or thousands of pounds of batteries that take much more volume than conventional fuel. For example, the battery of Wisk's Cora eVTOL flight test

demonstrator is located behind the cabin within the fuselage upsweep, while Uber Technologies' eCRM-003 uses large booms reminiscent of fighter external fuel tanks [227].

Since range is closely related to the quantity of batteries that can be carried on board, trade studies for electric aircraft need to rigorously account for the aircraft volume growth necessary to accommodate battery packs of varying sizes. This is a closely-coupled geometry-aerodynamic-structural-performance cycle, which can be analyzed through MDO. Safety considerations may introduce geometric requirements beyond non-interference. For example, transport aircraft are required to physically separate critical systems components so that no single failure can result in the loss of redundant functions. These kinds of trade studies are not easily accomplished using state-of-the-art MDO frameworks and present a barrier to the industrial adoption of MDO in practical vehicle design.

This chapter starts with a review of prior geometric integration (or "packing") optimization approaches, both from aerospace MDO and from outside the aerospace literature. Next, I develop a mathematical basis for an MDO geometric constraint, followed by a description of the computational implementation. Finally, I demonstrate aerodynamic shape optimization subject to spatial integration on three test cases of increasing complexity.

8.1 A Brief Review of Spatial Constraints in Shape Optimization Problems

A simple aerodynamic shape optimization problem can be posed as follows:

$$\begin{aligned} \text{minimize:} \quad & C_D \quad \text{(drag coefficient computed by CFD)} \\ \text{by varying:} \quad & \mathbf{x} \quad \text{(outer mold line shape design variables)} \\ \text{subject to:} \quad & \mathbf{g} \geq 0 \quad \text{(geometric constraints feasible)} \end{aligned}$$

Several simple methods for imposing geometric constraints on shape optimization problems

have already been developed. Kenway et al. [228] developed pyGeo, a software tool and geometry engine for high-fidelity MDO, which poses the following geometric constraints (illustrated schematically in Figure 8.1) as nonlinear inequality constraints:

Point thickness constraints: the simplest means of preventing the optimizer from reducing thickness too much in one location.

1D thickness constraints: enforced using a line of point thickness constraints suitable for preserving thickness along a line; for example, spar depth.

2D thickness constraints: enforced using an array of point thickness constraints suitable for preserving thickness over an area; for example, wing box thickness.

Volume constraints: useful for ensuring that a wing design has sufficient space for fuel.

pyGeo's constraint formulations have proven sufficient for a variety of wing and aircraft design optimization problems [147, 229, 230].

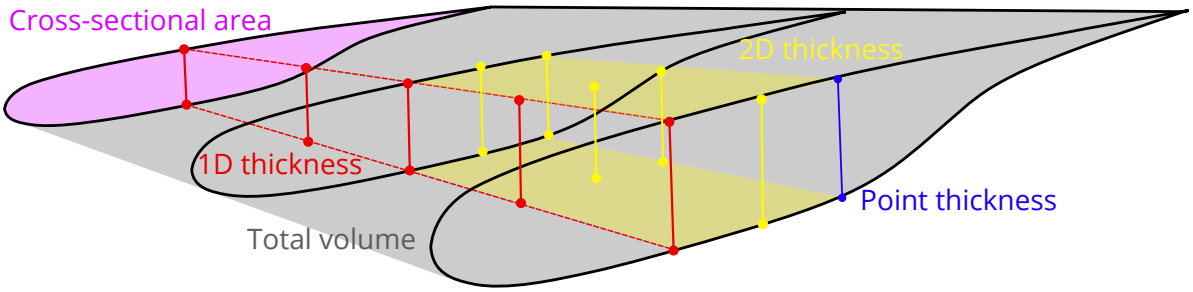


Figure 8.1: Existing geometric constraints for aerodynamic shape optimization

8.1.1 Packing Optimization Approaches in Aerodynamic Shape Optimization

The American Institute of Aeronautics and Astronautics (AIAA) Aerodynamic Design Optimization Discussion Group (ADODG) developed canonical benchmark cases which include min-

imum thickness and cross-sectional area constraints in 2D and 3D, as well as minimum volume constraints in 3D. Published solutions of the 2D cases used two general approaches to impose thickness constraints. Carrier et al. [231], Anderson et al. [232] and Fabiano and Mavriplis [233] chose 2D airfoil parameterizations with thickness control and set lower bounds on shape control design variables. Other authors [234–238] computed thickness at discrete points and imposed linear or nonlinear optimization constraints. Except Carrier et al., who used a constant-area parameterization, all authors imposed cross-sectional area as a nonlinear constraint.

All attempts to optimize the common research model (CRM) wing benchmark case imposed the minimum volume constraint as a computed nonlinear constraint. LeDoux et al. [236] used a volume discretization (voxel) approach to compute the constraint, while Carrier et al. [231] computed volume based on the triangulated surface mesh. Anderson and Aftosmis [239] imposed the additional 3D minimum thickness constraint as a 2D array of discrete thickness constraints along the vertical axis.

While these geometric constraint approaches are sufficient for these relatively simple research cases, they do not provide sufficient freedom to accurately reflect the complex geometries often found in industrial practice. In theory, relatively complex shapes can be represented by imposing a large number of point thickness constraints, but there are important practical limitations. In the absence of an automated tool, one would need to convert a 3D model of the object-to-fit into point thickness constraints by hand, which is error-prone and laborious. If the shape optimization involves both planform and thickness variables, the constrained shape stretches and deforms in unusual ways compared to the original intent.

Using thickness constraints to represent complex geometry also limits design freedom. If a thickness constraint is imposed at a planform location, the object-to-fit is effectively locked into the same planform location during the entire optimization. Unless the starting location is well-chosen, the lack of positioning design variables causes the resulting shape to be suboptimal. One can also easily envision realistic design problems in which components change shape as design re-

quirements change (e.g., motor diameter with rated torque). The geometric constraint formulations available today are not capable of handling these cases.

8.1.2 Packing Optimization Problems in Other Fields

While spatial integration has not been a focus in aircraft MDO, the operations research (OR) field has developed approaches suitable to the optimal packing of objects into a specified volume. "Tetris-like" rectangular volumes may be efficiently packed into a simple container using a mixed integer programming (MIP) approach [240]. This approach allows 90-degree permutations but not arbitrary rotations. It also requires a fixed outer bounding volume shape, which is analogous to an aircraft OML. There is no existing provision for coupling together the MIP problem with physics-based optimization (e.g., for minimum drag) of the outer volume. The MIP Tetris-like-solid formulation has been applied to spacecraft design applications [240].

The *trunk packing problem* is a canonical problem in the operations research packing optimization literature [241]. Automotive industry standards exist for the type and quantity of objects (e.g., suitcases and boxes) that must be able to fit into an automobile trunk. It is advantageous for auto manufacturers to fit as many of these irregular objects as possible into the trunk's bounded volume. Many approaches have been proposed to tackle this problem. One of the most general approaches is to use a volumetric discretization of the trunk; the subdivisions are called *voxels* [242, 243] Using a genetic algorithm, irregular objects can be rotated and translated in six degrees of freedom until no voxel is occupied by more than one object [243]. This approach is general and has been extended to handle a multiobjective engine compartment optimization to explore the tradeoff between compactness and maintainability [241]. Voxelization can also handle optimization problems where the components being packed vary in size during the optimization, such as maximizing coolant tank volume while still fitting all required components in the engine compartment [241].

However, like the MIP formulation, the volume discretization depends on a fixed OML, rendering it unsuitable for aerodynamic shape optimization applications where the OML changes

during optimization. The presence-or-absence voxel test is binary and non-differentiable, which is not suitable for gradient-based methods. Gradient-based methods are a necessity for high-fidelity shape optimization because gradient-free methods scale poorly with increasing degrees of design freedom and result in intractable computational cost [244].

Gradient-based packing optimization has been explored, but not as extensively as gradient-free approaches. Stoyan et al. [245] demonstrated gradient-based packing optimization of simple 2D and 3D geometric shapes. The approach depends on calculating a form of distance metric the authors call quasi-ϕ functions. These functions can be derived analytically for some classes of simple geometric shapes (such as a sphere or cone). Using this approach is desirable because the quasi-ϕ functions can be differentiated, making them suitable for gradient-based optimization.

Unfortunately, the quasi-ϕ approach is not very generalizable because it applies only to classes of geometric shapes for which the functions can be derived. This may be suitable for some conceptual design scenarios, where hold-out volumes of general shape and volume are known. For aircraft design, using this approach would require representing the OML geometry of the aircraft as a composition of geometric surfaces with known ϕ functions, which would require a major research effort in itself. The mathematics underlying the approach is also likely to be unfamiliar to most aerospace engineering practitioners. However, the results demonstrate that gradient-based packing optimization may be promising in at least some applications.

While the OR field has made substantial progress towards solving packing optimization problems, none of the prior approaches are suitable for use with high-fidelity MDO in aerospace. A new geometric constraint formulation is required, with the following properties:

General: The method should be able to represent arbitrary surfaces (including, at a minimum, a wing surface with high fidelity). The constraint metric calculation must not depend on the convexity of each object (since aircraft wings are often locally concave).

Differentiable: The constraint metric(s) must be differentiable and at least C^0 continuous (prefer-

ably C^1).

Efficient: The constraint metric(s) must have computation time and memory requirements with acceptable scaling properties (with hundreds to tens of thousands of geometric design variables, objects, and surface polygons).

8.2 Deriving a Mathematical Definition of Spatial Integration for MDO

To derive a mathematical definition of containment, let us consider some component (with outer surface A) to be fit inside OML surface B (Figure 8.2).

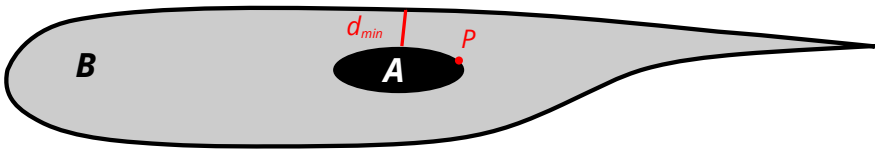


Figure 8.2: Schematic view of a wing OML and interior component

Definition. *Let A be a connected surface defined in three-dimensional real space.*

Definition. *Let B be a closed, connected, orientable surface defined in three-dimensional real space.*

Closed orientable 3D surfaces have a defined interior volume and can be thought of as 3D solids or closed 3D shells. A watertight CFD mesh always meets this definition. Formally, the volume enclosed by B and the surface B are not identical, but I call them both B for simplicity.

Definition. *Let d_{min} be the minimum distance between A and B.*

Definition. *Let ℓ be the sum of the length(s) of the intersection curve(s) between A and B, or 0 if no intersection.*

Next, an intuitive definition of geometric containment:

Definition. *A is contained within B if and only if all points P on A are inside B.*

Testing an infinite number of points P is not computationally tractable. Instead, I can replace this definition of containment with an equally intuitive one.

Definition. *A is contained within B if and only if:*

1. *any single point P on A is inside B, and*
2. *A and B do not intersect.*

A binary intersection test is not suitable for gradient-based optimization because it provides no gradient information. Therefore, I replace the second condition with a distance test with no loss of generality.

Definition. *Surface A is contained within bounding surface B if and only if:*

1. *any single point P on A is inside B, and*
2. *the minimum distance d_{min} between A and $B > 0$.*

This work focuses on Condition 2—the minimum distance tests between A and B. In an optimization context, Condition 1 (at least one enclosed point) can be satisfied by setting reasonable bounds on the geometric design variables. Alternatively, if there is a chance that the interior object might "escape" from the bounding volume during optimization, a ray-casting test can be used instead [246], as follows:

Theorem. *Point P is inside B if and only if a ray, originating from P and traveling in an arbitrary direction, intersects B an odd number of times.*

Unfortunately, the one-sided nature of Euclidean distance presents a practical difficulty for gradient-based optimizers because the value (and more importantly, the gradient) of the d_{min} goes to zero once A and B intersect. Figure 8.3 illustrates the minimum distance as a sphere translates closer to the edge of an ellipsoidal bounding volume. If the surfaces are discretized for computation (e.g., using a triangulated surface), an additional problem arises as d_{min} between the facets never reaches machine zero, even while intersected.

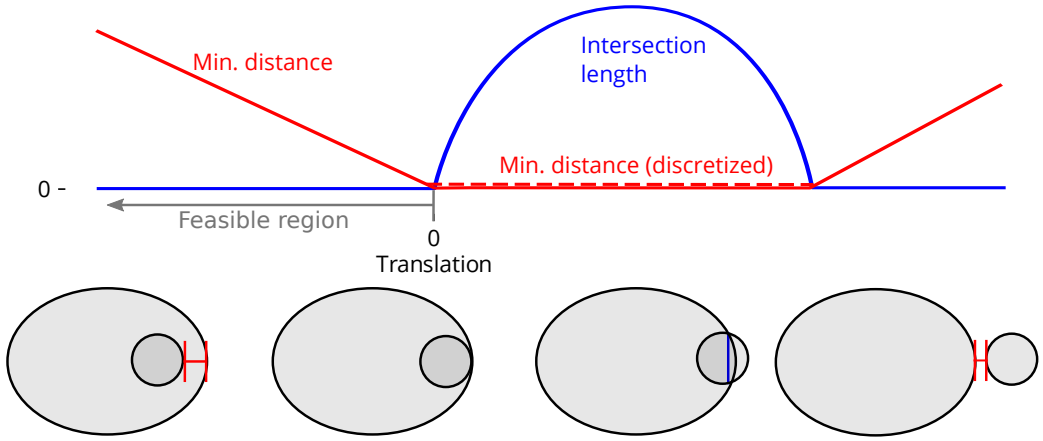


Figure 8.3: The minimum distance goes to zero post-intersection, presenting optimization difficulties.

To help the optimizer find its way back into feasible space once intersection occurs, I include the total length ℓ of the intersection curve(s) as an additional constraint. The intersection curve length provides feasibility gradient information once shapes are intersected and ensures that intersected surfaces are not falsely assessed as feasible due to discretization error (Figure 8.3).

Claim. *A gradient-based optimizer can ensure that convex component A fits inside convex bounding surface B if an arbitrary point P on A remains inside B, and the following nonlinear constraints are imposed:*

1. $d_{min} > 0$
2. $\ell = 0$

If design variable bounds and the geometric parameterization allow A to completely "escape" B (e.g., the fourth panel of Figure 8.3), the optimizer should be configured to reject iterations where P is outside B using a binary test, such as ray tracing.

A further complication occurs for nonconvex geometries. If the optimizer takes too large of a step into the intersected region, a gradient-descent or similar strategy does not allow the optimizer to return to the feasible region, as illustrated in Figure 8.4. Therefore, the optimizer must not take steps into the infeasible region that are too large.

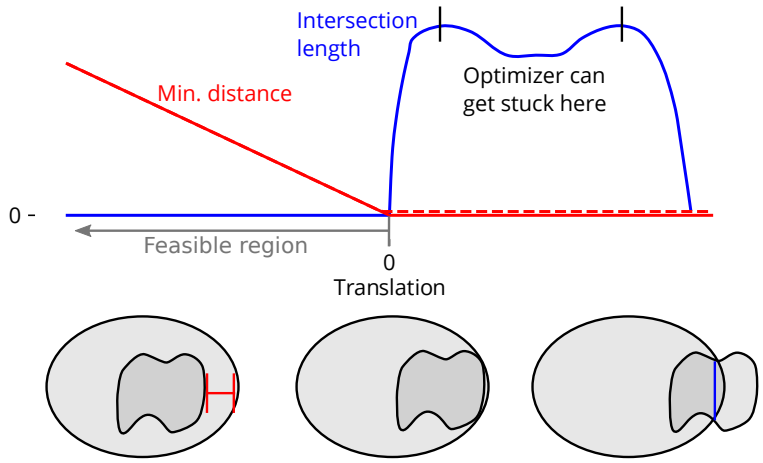
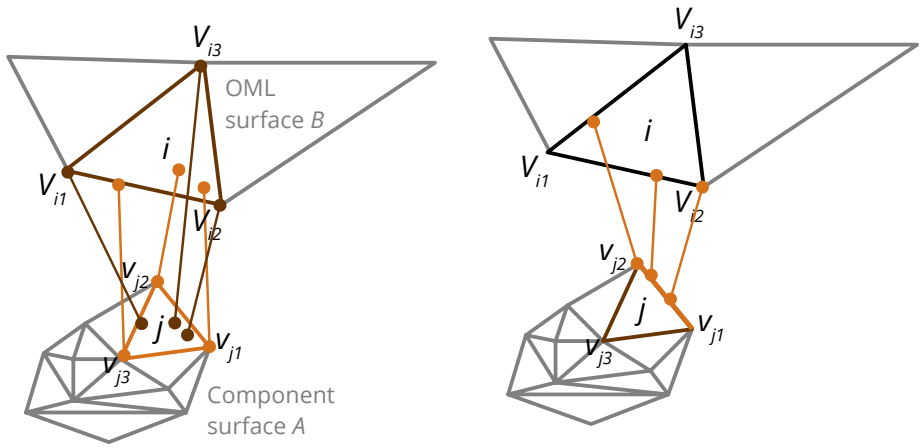


Figure 8.4: The optimizer can get stuck in the infeasible region for nonconvex geometries.

Claim. *A gradient-based optimizer can ensure that component A fits inside bounding surface B if an arbitrary point P on A remains inside B, the optimizer does not step into an intersected region, and the following nonlinear constraints are imposed:*

1. $d_{min} > 0$
2. $\ell = 0$

With this final statement, a mathematical definition of geometric feasibility is obtained that is computable, C^0 continuous, differentiable, and qualitatively well-behaved for nonlinear optimization algorithms.



(a) Point-triangle tests (6 of 6 shown)

(b) Edge-edge tests (3 of 9 shown; repeat for each edge)

Figure 8.5: Triangle distance tests

8.3 Computing Geometric Constraints

To compute the mathematical functions derived above, a data structure is needed to represent the surface geometry. While there are numerous approaches to computational geometry, I chose simple triangulated surfaces for generality and efficiency. Many unstructured CFD grids are natively triangular due to tetrahedral volumes, and structured grids using hexahedral volumes can be easily triangulated. Spline or other parametric surfaces are also easily triangulated. Furthermore, many efficient distance and intersection algorithms have already been developed for triangular geometry primitives.

The minimum distance between two triangles can be found through six point-triangle distance tests and nine edge-edge tests (Figure 8.5). I implemented the point-triangle and edge–edge distance tests of Ericson [247]. While lower-cost distance tests exist in the literature, Ericson's approach is vectorizable, allowing the use of analytic differentiation to obtain derivatives.

During optimization, the pair of triangles determining d_{min} changes for every iteration, and the

gradients of d_{min} are discontinuous. Tracking d_{min} alone also ignores useful information from the (possibly many) pairs of triangles that are *almost* the closest. Therefore, constraining d_{min} by itself is likely to produce bad results. Alternatively, all of the edge-edge and point-triangle distances could be constrained to be greater than zero. However, problem scaling makes this impractical. For example, suppose a problem has $n = 10^4$ OML mesh facets, $m = 10^3$ mesh facets per object, $k = 10^1$ objects, and $p = 10^2$ geometric design variables. There are $\mathcal{O}(nmpk) = 10^{10}$ gradient entries to store, which easily exceeds available memory even for modest problems.

Ideally, all of the almost closest facets would contribute to a single constraint. A good way of achieving this is constraint aggregation using the alternative form of the Kreisselmeier-Steinhauser (KS) function [208, 248],

$$KS\Big[\mathbf{g}\,(\mathbf{x})\Big] = g_{\text{max}}\,(\mathbf{x}) + \frac{1}{\rho}\ln\left[\sum_{j=1}^{m} e^{\rho\left(g_j\,(\mathbf{x}) - g_{\text{max}}\,(\mathbf{x})\right)}\right], \tag{8.1}$$

where $\mathbf{g(x)} \leq 0$ is the vector of inequality constraints, evaluated at the design point $\mathbf{x}$, $g_{\text{max}}(\mathbf{x})$ is the maximum constraint value at the current design point, and ρ is a constant associated with the aggregation function. For spatial integration, $\mathbf{g}\,(\mathbf{x}) = -d_j\,(\mathbf{x})$, where $d_j\,(\mathbf{x})$ is the vector of all distance test results between the facets of A and B. Using this relationship, I obtain the final form of the distance constraint for optimization:

$$KS_{\text{geom}}(\mathbf{x}) = \frac{1}{\rho}\ln\left[\sum_{j=1}^{J} e^{\rho\left(d_{\text{min}}\,(\mathbf{x}) - d_j\,(\mathbf{x})\right)}\right] - d_{\text{min}}\,(\mathbf{x}) \leq 0, \tag{8.2}$$

where $J = 15mn$ is the number of distance tests between A and B.

The KS function produces a conservative estimate of the maximum value of its inputs while preserving function smoothness and derivative information from almost-active inputs. As the minimum distance between the two objects approaches zero, the KS function returns a positive (infeasible) value to the optimizer. The level of conservativeness of the estimate can be adjusted by

changing ρ. Higher values of ρ lead to a tighter spatial fit, but the problem may be more poorly conditioned. As $\rho \to \infty$, $KS_{\text{geom}} \to -d_{\text{min}}$, but information from other close pairs of triangles is lost. Refining the resolution of the spatial discretization of either surface without increasing ρ generally makes the estimate more conservative. I explored alternative constraint aggregation approaches, such as the DIE [207, 208]; however, they do not provide a strictly conservative estimate like the KS function.

The second constraint (intersection length) is calculated following the method of Moller [249]. First, intersections between pairs of triangles are detected, if any. Then, the line segment(s) S_i of intersection are computed. Finally, total length ℓ is obtained by summing all intersection line segment lengths.

The minimum distances and intersections between two surfaces consisting of n and m triangles, respectively, can be computed by performing $\mathcal{O}(nm)$ pairwise distance tests. This operation is embarrassingly parallel, since none of the triangle-triangle comparisons depend on any of the others. Because n and m are both on the order of 10^3 to 10^5, the distance tests must be efficient.

To obtain the desired computational performance, I considered using a compiled language (such as FORTRAN) and incorporating branch logic with many early exits to avoid unnecessary computations. I would then obtain gradients for optimization via an automatic differentiation (AD) tool. An alternative approach for parallel problems is to use a graphics processing unit (GPU). GPUs are so much faster than a central processing unit (CPU) for repetitive parallel tasks that wasted computations may be acceptable and the intricate branch logic to avoid them may not be necessary.

Researchers have sometimes avoided using GPUs in the past partly because of the steep learning curve. Fortunately, the maturation of machine learning frameworks has made GPU computing substantially more accessible. TensorFlow is an open-source Python package originally developed at Google for machine learning workflows [250]. Since the majority of production machine learning tasks are performed on specialized hardware rather than CPU, TensorFlow includes a

Python application programming interface (API) providing access to fast mathematical routines on GPUs. I verified speedups of 25–100 for typical vector and tensor operations (such as addition, dot product, and min/max) compared to NumPy with Intel Math Kernel Library (MKL) on CPU. Another compelling feature of TensorFlow is the built-in gradient computation. All mathematical operations and most control flow operations have reverse-mode differentiated code built in. TensorFlow propagates reverse-mode derivatives of model outputs with respect to model inputs across the graph of operations. I implemented the mathematical approach described in Section 8.2 as a Python library using TensorFlow, under the working title `geograd`.

8.4 Aerodynamic Shape Optimization Subject to Geometric Constraints

To validate this general geometric integration constraint, I constructed three aerodynamic shape optimization test cases of increasing complexity: a 2D fairing around a box for minimum drag, a 3D fairing around a cylinder for minimum drag, and a 3D fairing around a human avatar for minimum drag.

8.4.1 Methodology

I performed all optimizations using the MACH framework [154]. MACH is a set of Python-, FORTRAN-, and C++-based tools and utilities that provide all necessary geometry, aerodynamics, and structural analyses to perform high-fidelity aircraft MDO. Individual components of MACH are described in detail in previous publications [225, 251], but I briefly review the overall methodology here.

For aerodynamic analysis, I used ADflow, an open-source code that solves the RANS equations

on structured overset meshes (though I did not use overset capability for this study) [252][1]. I used ADflow to evaluate drag and its sensitivities. The adjoint solver in ADflow computes the gradients of output quantities (e.g., drag) efficiently, even for problems with thousands of design variables [253]. Together, ADflow and the MACH framework have been used to perform transonic wing optimization for minimum drag [225, 254–256], as well as aerostructural optimization [147, 148, 229].

Unlike wing optimization cases, the test cases for this chapter often go through intermediate designs with bluff-body aerodynamics during optimization. This primarily occurs due to the cross-flow conditions included in the multipoint aerodynamic problem. Many RANS solvers have trouble converging cases with separated flow. However, ADflow is particularly robust in solving these intermediate cases.

ADflow's robustness is attributable to the approximate Newton–Krylov (ANK) nonlinear solver scheme [252]. The ANK solver uses a matrix-free approximate Jacobian formulation with the pseudo-transient continuation method. Using ADflow with ANK, cases with heavy separation in the flow-field may converge, improving the likelihood that the optimization will succeed. Intermediate steps with difficult aerodynamic characteristics can also be avoided by utilizing trust region methods, by carefully re-formulating the optimization problem into multiple subproblems, or both. However, both of these approaches require intervention by an expert user. With ANK, the user need not be as careful with problem setup. This enabled me to rapidly study the range of challenging optimization problems presented in this chapter.

Two other open-source MACH tools, pyGeo [2] and IDWarp [3], provide geometric parameterization and mesh deformation, respectively. pyGeo uses a free-form deformation (FFD) approach to modify the surface mesh with respect to the design variables [228]. The FFD approach embeds surface mesh vertices into a solid block, which deforms as geometric design variables change during

[1] https://github.com/mdolab/adflow
[2] https://github.com/mdolab/pygeo
[3] https://github.com/mdolab/idwarp

optimization. Since only the surface mesh points are directly parameterized, changes in the surface mesh must be propagated to the volume mesh. IDWarp accomplishes this mesh deformation using an inverse distance mesh warping method similar to the method developed in Luke et al [257]. The initial volume mesh itself is generated using pyHyp [251], which uses a hyperbolic volume mesh marching scheme [258]. [4]

pyGeo also contains a sub-module, DVConstraints, which computes nonlinear geometric constraints, such as thickness and volume constraints, based on the surface mesh geometry. I extended DVConstraints with `geograd` to compute the general 3D geometry constraint described in Section 8.2.

Finally, I used Sparse Nonlinear Optimizer (SNOPT) (Sparse Nonlinear OPTimizer) [259], wrapped in pyOptSparse [223], as the nonlinear optimizer to drive the problem. [5] SNOPT is especially useful for large-scale problems with many design variables, and for problems with functions that are expensive to evaluate. It has been used repeatedly for aerodynamic and aerostructural optimization problems [225, 255]. A key feature of SNOPT for this study is that is can handle objective function evaluation failures. If an analysis code fails (such as a CFD convergence failure), SNOPT reduces its step size and tries again. I used this feature to handle intersected cases. If the geometry tool detects intersection between the CFD surface mesh and the constraint object mesh, it returns a failure flag to SNOPT, which forces the optimizer to backtrack and try again with a less aggressive step. This worked qualitatively well for us, since it avoids wasting an expensive CFD evaluation.

I ran all optimization cases on a desktop computer with an Intel Core i7-6700K CPU (4 cores) and an NVIDIA GeForce GTX 970 GPU. ADflow runs in parallel on the CPU, while the geometric constraint framework runs on the GPU.

[4] https://github.com/mdolab/pyhyp
[5] https://github.com/mdolab/pyoptsparse

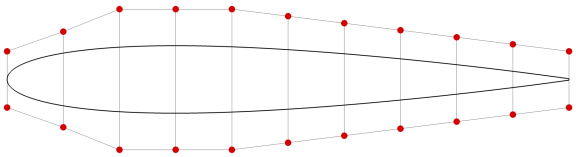


Figure 8.6: FFD control points and initial surface mesh for 2D cases

8.4.2 2D Aerodynamic Shape Optimization Around a Box

I first demonstrated the new geometry constraint using a 2D aerodynamic shape optimization problem. The objective function was to minimize multipoint average drag around a box at 0° and 10° crossflow angle. The starting surface mesh was a National Advisory Committee for Aeronautics (NACA) 0012 airfoil (Figure 8.7). The 2D parameterization consisted of 22 FFD points providing fine shape control in the y axis (Figure 8.6), one variable to translate the entire volume in the streamwise x direction, and one variable for chord in the x-axis.

I imposed symmetry in the y-axis to effectively obtain a $-10°$ crossflow case without running additional CFD cases. The optimization parameters are described in Table 8.1. To prevent negative warped mesh volumes near the trailing edge in single point cases, I also included a line of point thickness constraints at $x/c > 99\%$. Throughout this chapter, C_D is computed with respect to a fixed reference area of 1 m^2.

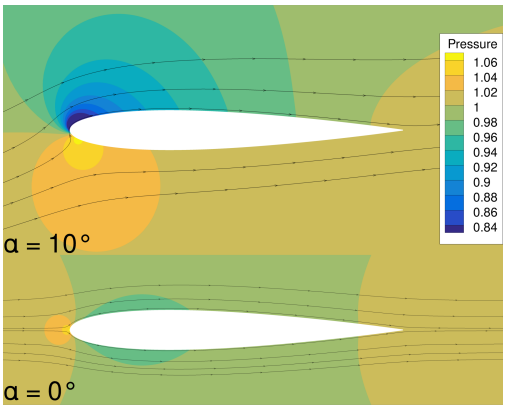


Figure 8.7: 2D multipoint optimization initial condition

	Variable	Description	Quantity
minimize	avg($C_{D,\alpha=0}$, $C_{D,\alpha=10}$)	Average drag at 0° and 10° angle of attack	
by varying:	$x_{\text{FFD}y}$	y-axis shape control variables (free-form deformation)	22
	x_{chord}	Streamwise chord length	1
	$x_{\text{translate}}$	Streamwise translation	1
subject to	$x_{\text{FFD}y,\text{lower}} = -x_{\text{FFD}y,\text{upper}}$	Symmetry across y-axis	11
	$KS_{\text{geom}} \leq 0$	Aggregated min. distance (spatial feasibility cond. 1)	1
	$\ell = 0$	Nonintersection (spatial feasibility cond. 2)	1
at condition	$M = 0.3$		
	$Re = 6 \times 10^6$		
	Sea level, standard day		

Table 8.1: 2D aerodynamic shape optimization parameters

I ran five cases with varying spatial tightness parameter ρ, as shown in Figure 8.8. Table 8.2 shows that as spatial tightness is increased, the potential drag reduction increases, with diminishing returns at large (tight) values of ρ. I also found that tightening spatial tolerances increased computational cost by about 50%. As ρ increases, the local curvature (nonlinearity) of the function increases, which reduces the accuracy of the optimizer's quasi-Newton step. A second effect of large ρ is that the KS contribution from nearby facets becomes large compared to distant segments. Floating point errors round the contributions of distant segments to zero in the constraint Jacobian, which becomes sparser as a result. Both of these effects degrade optimizer performance.

Case	Time (min)	CFD evals	C_D	ΔC_D
Baseline	–	–	0.01069	–
$\rho = 200$	130	40	0.01170	+9.4%
$\rho = 300$	131	42	0.01026	−4.0%
$\rho = 600$	123	56	0.00887	−17.0%
$\rho = 900$	185	80	0.00842	−21.2%
$\rho = 1200$	194	82	0.00820	−23.3%

Table 8.2: Drag reduction potential increases with tighter spatial integration tolerance ρ

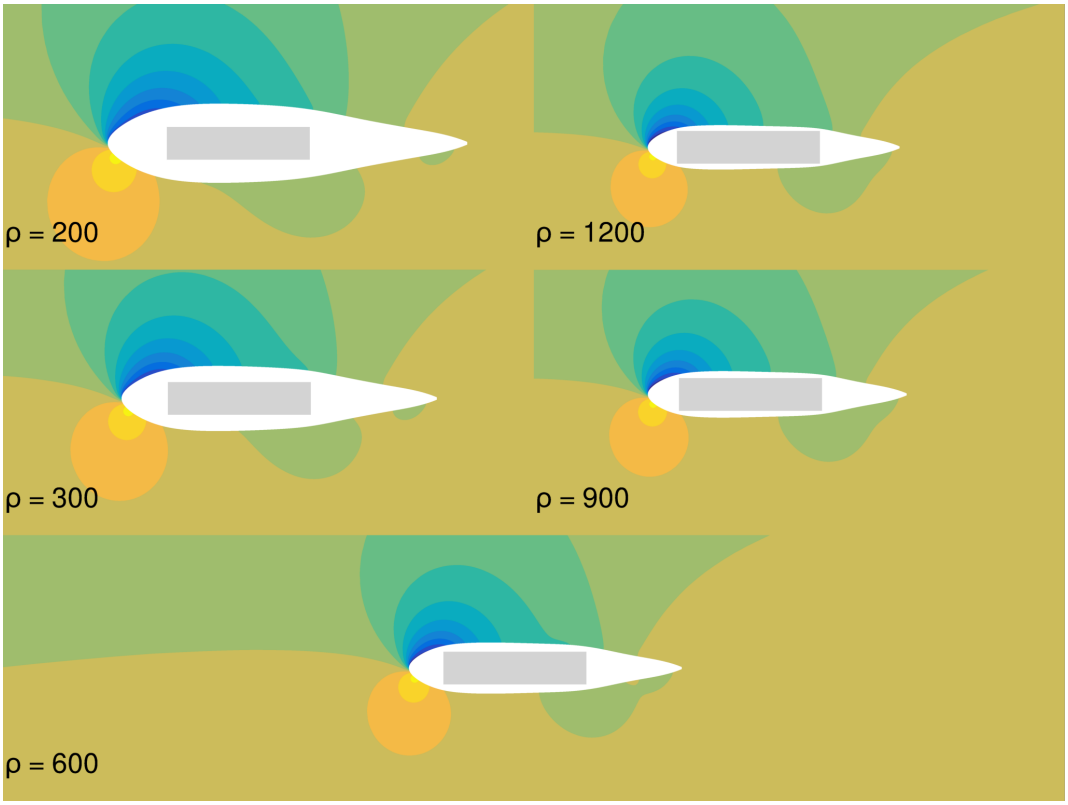
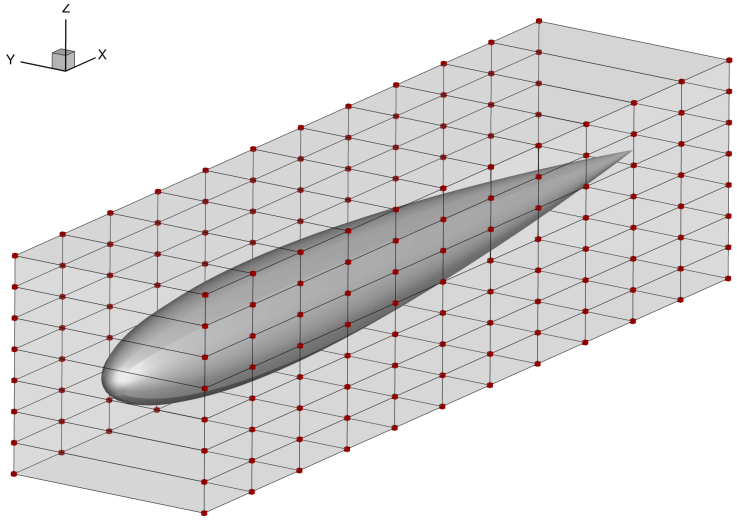


Figure 8.8: 2D multipoint minimum drag results for varying spatial tolerances (ρ)

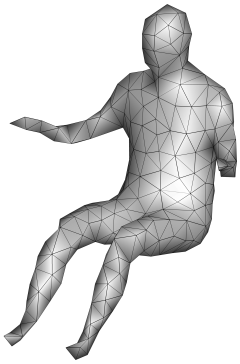
8.4.3 3D Aerodynamic Shape Optimization Around a Simple Surface of Revolution

For a simple 3D test, I defined single-point (0°) and multipoint (0° and 20°) drag minimization cases. I created a starting surface mesh consisting of a NACA 0012 airfoil revolved around the streamwise (x) axis (Figure 8.10). The structured surface and volume meshes consisted of 1802 and 237762 cells, respectively. The 3D parameterization consists of 192 FFD points which provide fine shape control along the y-axis (Figure 8.9a), and an additional 17 parameters providing degrees of freedom in the x and z axes. I imposed symmetry in the crossflow (x-y) plane to effectively obtain a −20° crossflow case without running additional CFD cases. The optimization parameters are described in Table 8.3.

Figure 8.11 shows the optimized shape for the single point case. The drag decreased 29.3%



(a) FFD control points and initial surface mesh



(b) Constraint geometry for human cases

Figure 8.9: Geometry for 3D optimization cases

	Variable	Description	Quantity
minimize	avg($C_{D,\alpha=0}$, $C_{D,\alpha=20}$)	Average drag at 0° and 20° angle of attack (multipoint), or	
	$C_{D,\alpha=0}$	Drag at 0° angle of attack (single point)	
by varying	$x_{\text{FFD}y}$	y-axis FFD shape control variables (free-form deformation)	192
	x_{chord}	Streamwise chord length of "airfoil" slices along z-axis	8
	$x_{\text{translate}}$	Streamwise translation of "airfoil" slices along z-axis	8
	x_{span}	"Span" along z-axis	1
subject to	$x_{\text{FFD}y,\text{left}} = -x_{\text{FFD}y,\text{right}}$	Symmetry across y-axis	96
	$KS_{\text{geom}} \leq 0$	Aggregated minimum distance (spatial feasibility condition 1)	1
	$\ell = 0$	Nonintersection (spatial feasibility condition 2)	1
at condition	$M = 0.3$		
	$Re = 6 \times 10^6$		
	Sea level, standard day		

Table 8.3: 3D aerodynamic shape optimization parameters

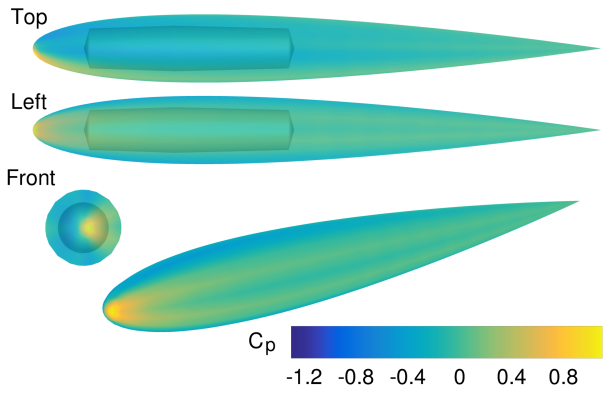


Figure 8.10: 3D ASO starting point: NACA 0012 surface of revolution enclosing a cylinder

compared to the baseline single point case. The optimized shape is a long fairing with relatively tight leading edge curvature. The tightly curved leading edge is characteristic of single point aerodynamic shape optimization, since robustness to varying flow conditions is not required. Even though the problem is parameterized in a Cartesian frame, the finished shape is almost perfectly axisymmetric, as we would expect from a rotationally symmetric spatial constraint and flow condition.

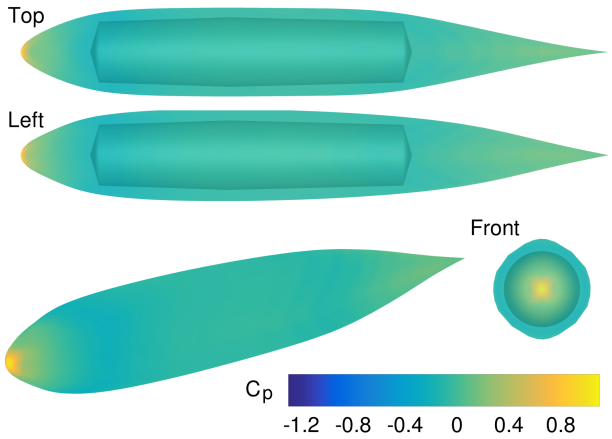


Figure 8.11: 3D single point minimum drag result for cylinder, $\rho = 1200$

Figure 8.12 shows the optimized shape for the multipoint case. Drag decreased 52.2% compared to the baseline multipoint case. The optimized shape has a gently rounded nose and broader

aft body closure angle, which visually matches the crossflow condition. Compared to the single point case, the crossflow condition has added thickness in the x-y plane and reduced length overall. The rounder nose and extra thickness-to-chord ratio both help improve resistance to flow separation at the crossflow condition. Visually, the x-y thickness is not required simply for spatial integration reasons, but the x-z plane generally tightly conforms to the cylinder. This illustrates that it is not evident *a priori* whether geometric constraints will be active for a particular problem; physics and geometry are strongly coupled.

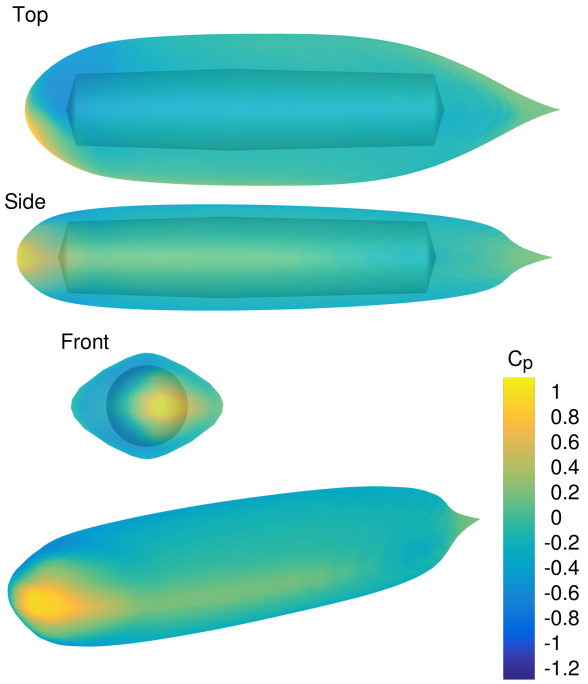


Figure 8.12: 3D multipoint minimum drag result for cylinder, $\rho = 1200$

8.4.4 3D Aerodynamic Shape Optimization Around a Human Avatar

Finally, I set up single and multipoint optimization cases where the geometry to be enveloped is that of a person in a seated position, with parameters identical to the previous case (Table 8.3). I exported a high-resolution model of an average U.S. adult in a seated driving position to a

stereolithography file (STL) using the University of Michigan Transportation Research Institute's (UMTRI) online tool[6] [260]. I then resized and reduced the complexity of the triangulated mesh using Autodesk Meshmixer[7] and imported it directly into the optimization environment; the final mesh (Figure 8.9b) had 626 triangles. Figure 8.13 shows the initial condition of the optimization.

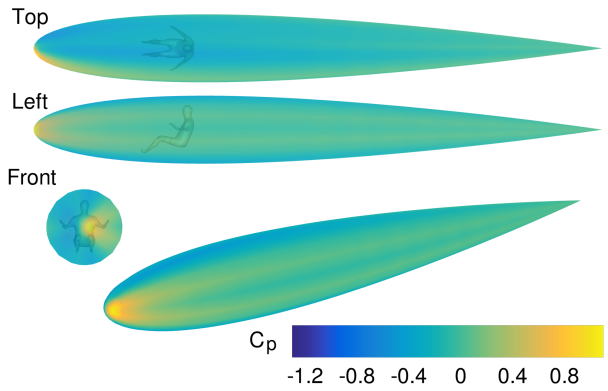


Figure 8.13: 3D ASO starting point: NACA 0012 surface of revolution enclosing human avatar

Figure 8.14 shows the optimized shape for the single point case. The drag decreased 61.3% compared to the grossly oversized baseline single point case. The optimizer generated a rounded leading edge and an elongated trailing cone with moderate closure angle. In this case, the rotational asymmetry is due to the asymmetric constraint geometry, not the flow condition.

Figure 8.15 shows the optimized shape for the multipoint case. Drag decreased 78.5% compared to the grossly oversized baseline multipoint case. Compared to the single point case, we see an even blunter leading edge and dramatically shorter overall length. The result is an even larger thickness-to-chord ratio, which reduces drag in the crossflow condition. The multipoint case benefits more from shape optimization than the single point case. This may be due to in part to the better relative performance of the initial NACA 0012 geometry in oncoming flow compared to significant crossflows.

I then tightened the spatial tolerance and re-ran the multipoint case. Figure 8.16 shows that the

[6] http://humanshape.org/
[7] http://www.meshmixer.com/

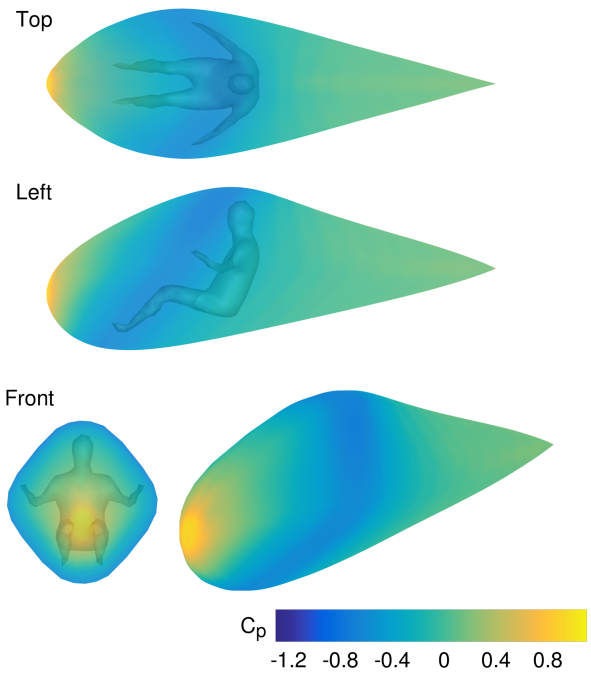


Figure 8.14: 3D single point minimum drag result for human, $\rho = 1200$

spatial fit has tightened significantly compared to Figure 8.15. There is a noticeable decrease in frontal area of the fairing. Drag decreased an additional 16% compared to the $\rho = 1200$ case. This illustrates the direct design tradeoff between spatial requirements and aerodynamics.

8.5 Concluding Remarks

Geometric (or spatial) integration constraints have been a barrier to industrial adoption of high-fidelity MDO for vehicle design applications. Existing geometric constraint techniques are limited in their ability to capture complex geometries. I reviewed the state-of-the-art in packing optimization problems and identified gaps that prevent existing methods from being directly applied to aerospace MDO. I then developed a mathematical approach for imposing spatial integration constraints based on triangulated surfaces and constraint aggregation using the KS function. This allows practitioners to specify geometric design intent, even for complex geometries, in an intu-

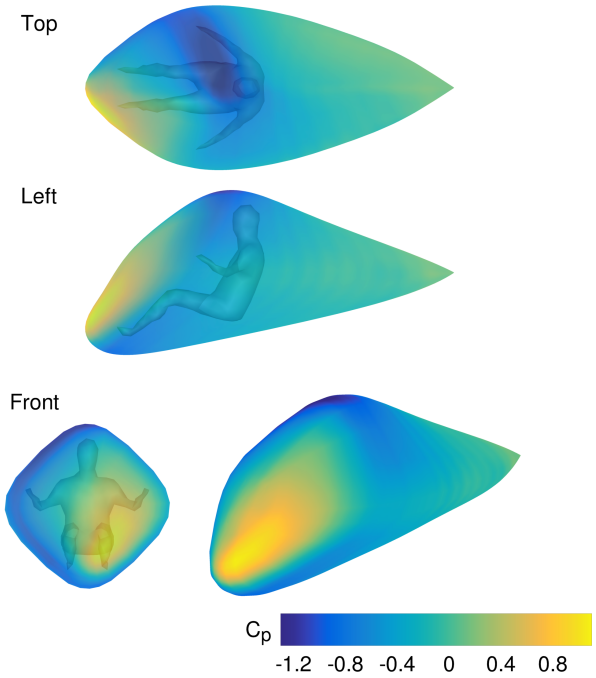


Figure 8.15: 3D multipoint minimum drag result for human, $\rho = 1200$

itive and visual way. The method is general, differentiable, and efficient, making it suitable for use with gradient-based optimization with high-fidelity simulation tools.

I demonstrated the performance of the method by minimizing drag for three test geometries of increasing complexity. Drag decreased by 30% to 80% compared to the initial designs. Multipoint aerodynamic problems with crossflows benefited more from shape optimization. Tightening spatial tolerances reduced drag further, as expected. By varying the spatial tolerance ρ, designers can perform high-fidelity trade studies to quantify the value of tighter systems or payload packaging.

All optimizations, including the multipoint 3D cases, were run on a desktop computer in less than 24 hours, placing the shape optimization capability in reach of typical MDO engineers. This new constraint formulation is a major improvement over point thickness geometric constraints and an important step towards solving aircraft systems packaging MDO problems.

This material was previously published [261] and has been lightly edited and formatted.

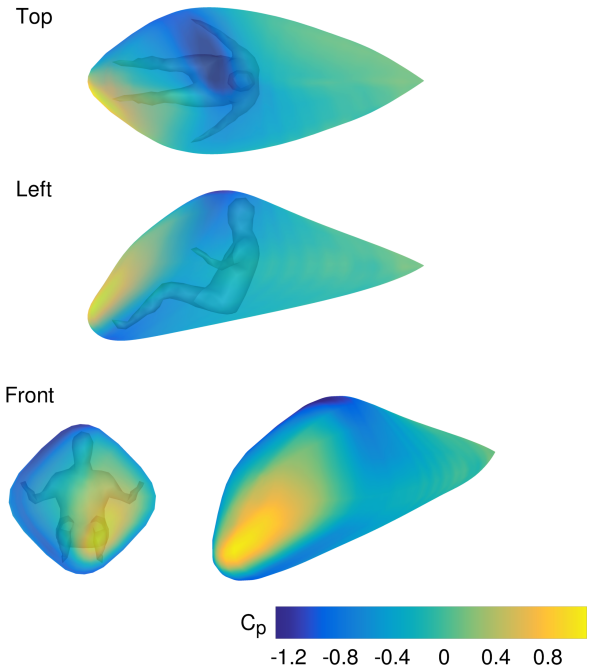


Figure 8.16: 3D multipoint minimum drag result for human, $\rho = 3000$

CHAPTER 9

Aerodynamic Shape Optimization of a Wing Considering Battery Packaging

9.1 Introduction

The previous chapter introduced a novel way of posing spatial constraints in shape optimization problems, and demonstrated aerodynamic shape optimization subject to *fixed*, arbitrary geometric constraints. I demonstrated through the test cases that the constraint formulation can successfully handle high-fidelity aerodynamic shape optimization cases with simple or complex geometry. However, in Chapter 8, I only varied the shape of the outer aerodynamic surface during optimization, not the shape of the inner object. Varying both the outer surface and inner component(s) is necessary for fully coupling aerodynamics and systems components through spatial integration. In this chapter, I demonstrate an MDO problem coupling both systems design and aerodynamic shape. The case study involves designing the wing and battery pack of a notional electric aircraft for maximum range.

9.2 A Brief Review of High-Fidelity Design Optimization of Electric Aircraft

Electric propulsion introduces new components, such as batteries and power electronics, and opens many new questions about how airplanes ought to be configured. Are batteries best installed inside the wing box, or in the fuselage? What is the optimal tradeoff between battery volume and aerodynamic efficiency? What is the optimal motor diameter for distributed propulsion along a wing? Each of these questions can be answered with a high-fidelity MDO trade study involving both outer mold line shape, and inner component layout and sizing. However, because of the limited flexibility of spatial constraints used in current MDO practice, these types of trades are difficult to do today.

Various authors have performed MDO trade studies of electric aircraft at a conceptual level of fidelity[1, 17, 27, 160]. Hwang and Ning [163] modeled and optimized a battery-powered aircraft similar to the NASA X-57 using moderate-fidelity conceptual tools implemented in NASA's OpenMDAO framework [155]. Battery sizing was considered but not the actual geometry. Gray et al. [157] optimized the STARC-ABL turboelectric concept's aft body propulsor using CFD. Other authors have optimized battery cells and battery packs using multiphysics models including electrochemical models [133], convective cooling models including CFD [262, 263] and structural finite element models [264]. However, prior to the work in this Chapter, no published work explored optimal battery layout and geometry for aircraft including high-fidelity aerodynamic coupling.

9.3 Improvements to the Constraint Implementation

Following the proof-of-concept work detailed in Chapter 8, I made substantial improvements and extensions to `geograd`. The `geograd` revision level used in Chapter 8 computed derivatives with respect to one surface mesh only. Following updates for this chapter, `geograd` computes

derivatives with respect to both surfaces simultaneously. This is required for problems where both surfaces may change in shape during optimization.

I also implemented better computational scaling on CPU-based environments. Previously, `geograd` used only GPU acceleration to provide rapid evaluation of the large number of pairwise computations. Since I anticipate using mostly CPU-based high-performance computing resources, I data-parallelized `geograd` across an arbitrary number of MPI processes when no GPU is available. `geograd` breaks up object A's mesh and distributes it across each process, performing pairwise computations locally and then reassembling aggregate values and gradients across all processes later. This MPI parallel implementation runs significantly faster than TensorFlow's multithreaded CPU implementation on a single computer.

9.4 Coupling Systems Design, Aerodynamics, and Spatial Integration: A Validation Study

Aircraft design with electric propulsion provides a timely test problem for fully-coupled systems sizing and aerodynamic design. Let us consider a transport aircraft with conventional wings and all-electric propulsion. For battery-powered aircraft, a modified Breguet equation provides an estimate of range [9, 17, 24]

$$R_b = \frac{L}{D}\eta_e\eta_{int}\eta_p\frac{e_b}{g}\frac{m_b}{m_{TO}}. \tag{9.1}$$

where η_e is the total electrical efficiency of the propulsion system, η_{int} is efficiency loss due to propulsion integration, η_p is the propulsive efficiency, e_b is the battery specific energy, g is the gravitational constant, and m_b/m_{TO} is the ratio of battery weight to takeoff gross weight.

Assuming fixed m_{TO} (i.e., trading payload for batteries) and focusing only on terms which

vary significantly during aerodynamic shape optimization, I have:

$$R_b \propto \frac{L}{D} m_b \propto \frac{L}{D} \frac{V_b}{\rho_b} \tag{9.2}$$

where V_b and ρ_b are battery volume and mass density, respectively. However, adding range is not as simple as adding battery volume, since L/D and battery volume are coupled through spatial integration. If the aircraft OML is constrained by battery shape, L/D will tend to decrease as battery volume increases. Lithium-ion batteries are much denser than jet fuel (in terms of mass per volume) and a wing enclosing them is likely to be weight-limited rather than volume-limited. However, emerging battery chemistries such as Li-S and Li-air are expected to have similar mass density to jet fuel [265, 266]. Combined with the need for higher weight fraction to achieve useful range, these alternative chemistries are much more likely to experience wing volume limitations. Furthermore, in high fidelity, the specific location of the batteries will have a large effect on range (it is generally more efficient to add volume at the root of a wing than at the tip).

I constructed a multi-point (2 flight condition) optimization problem as described in Table 9.1.

The subsections below will describe variations on this general problem: in particular, single- versus multi-point optimization and restrictions on battery geometry. The optimizations were performed using the MACH framework [154, 225]. The computational methodology was very similar to Chapter 8. Notably, I used the open-source ADflow[1] adjoint RANS solver to efficiently evaluate drag and its sensitivities with respect to geometric design variables [252, 267]. For geometric parameterization, I use pyGeo's[2] free-form deformation (FFD) approach [228]. The starting wing geometry is similar in dimensions to a DC-9 planform, albeit with a more-modern RAE 2822 transonic airfoil. The wing was parameterized with 96 independent free-form deformation control points (Figure 9.1). After a mesh refinement study, I selected a mesh with 816,392 volume

[1] https://github.com/mdolab/adflow
[2] https://github.com/mdolab/pygeo

	Variable	Description	Quantity
maximize	$\frac{L}{D}V_b$	Range, averaged from flight conditions 1 and 2	
by varying	$x_{\text{batt,geom}}$	Battery FFD geometry	varies
	$x_{\text{FFD,wing}}$	Wing airfoil section geometry	96
	$x_{\text{twist,wing}}$	Twist distribution along wing axis	7
	α	Angle of attack	2
subject to	$C_{L,\text{fc1}} = 0.5$	Lift constraint at flight cond. 1	1
	$C_{L,\text{fc2}} = 0.4$	Lift constraint at flight cond. 2	1
	$KS_{\text{geom}} \leq 0$	Battery packaging feasibility (min. distance)	1
	$\ell = 0$	Battery packaging feasibility (nonintersection)	1
	$\boldsymbol{t}_{\text{wing}} \geq 0.5$	Wingbox thickness no less than half of baseline	1
at condition	$M = 0.8$		
	$h = 10000$ m		
	standard day		

Table 9.1: Battery packaging shape optimization parameters

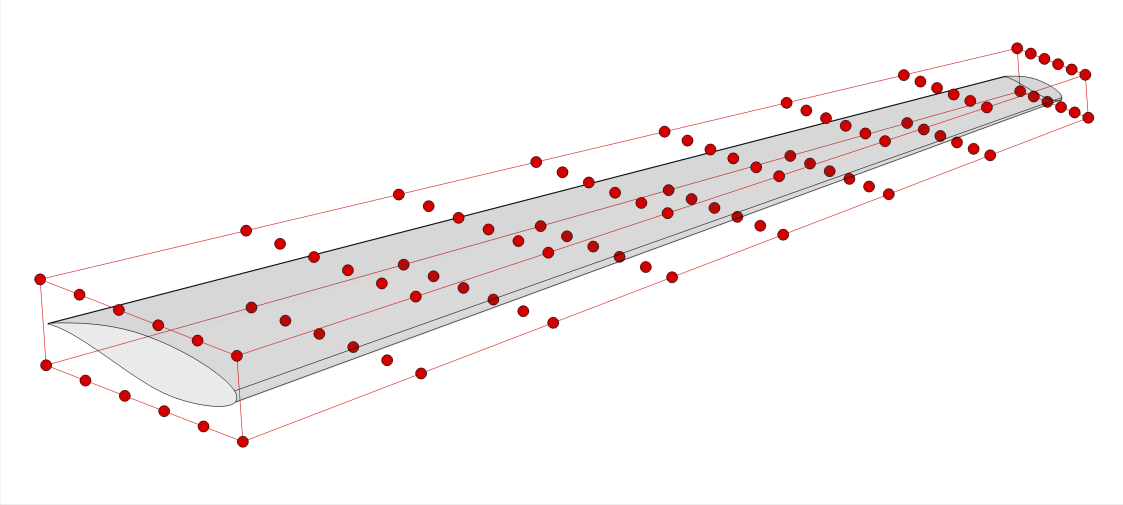


Figure 9.1: Wing geometric parameterization control points

cells. I used SNOPT (Sparse Nonlinear OPTimizer) [259], wrapped in pyOptSparse[3] [223], as the nonlinear optimizer to drive the problem.

I ran all optimization cases on single 48-core Skylake nodes on the Texas Advanced Computing Center (TACC) Stampede2 high-performance computing (HPC) cluster. Both ADflow and `geograd` (the geometric constraint package) are MPI parallelized with one process per physical

[3]https://github.com/mdolab/pyoptsparse

core.

	Baseline	Multipoint Conformal	Multipoint Rectangular	Singlepoint Conformal
Objective ($V_b L/D$)	187.82	305.59	254.97	314.05
Battery volume (m^3)	7.560	12.18	10.31	12.29
Opt. wall time (one node, h)		68.7	73.5	16.0
Flight Condition 1				
C_L	0.5000	0.5000	0.5000	0.5000
C_D	0.0214	0.0199	0.0203	0.0196
C_M	0.7593	0.7160	0.7301	0.7347
L/D	23.35	25.09	24.67	25.55
Flight Condition 2				
C_L	0.4000	0.4000	0.4000	
C_D	0.0152	0.0159	0.0161	
C_M	0.6292	0.587	0.601	
L/D	26.34	25.10	24.77	

Table 9.2: Summary of optimization results

9.4.1 Conformal Battery

For the first case, I created an initial battery geometry by trimming the leading and trailing edge of the starting wing and reducing its thickness slightly, producing a shape which conforms to the wing. I used the same set of 96 FFD control points illustrated in Figure 9.1 to parameterize the battery (although its optimizer design variable values are independent of the wing). I then optimized the wing and battery using the multi-point problem formulation listed above. The optimization was considered converged once SNOPT's optimality metric was reduced three orders of magnitude. Table 9.2 shows high-level results from the optimization run, which took 68.7 hours on one TACC Skylake node. The range quantity increased by 62.7% compared to the baseline design. Figure 9.3 shows a before-and-after comparison of pressure, lift, and twist distribution, airfoil section geometry, and battery section geometry. The optimizer greatly increased inboard wing thickness. Outboard of the battery pack end, the wing was reduced to minimum thickness.

The battery, although parameterized independently from the wing, fits tightly to the wing upper and lower surfaces with very small gaps.

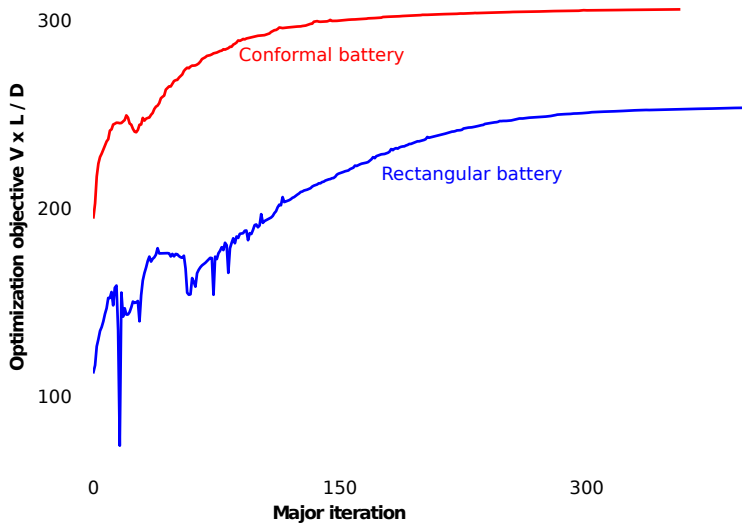


Figure 9.2: Optimization histories for conformal and rectangular battery problems

9.4.2 Rectangular Battery

Because today's batteries are generally composed of cylindrical cells of finite size, it is likely that arbitrary battery curvatures cannot be achieved. To explore the effect of geometric restrictions on battery design, a second initial battery geometry was created by flattening the upper and lower surfaces of the battery box from the previous problem, which produced a shape with no initial curvature. I then split the flattened battery into three separate sections, each with two thickness design variables (one upper, one lower) at all eight spanwise stations. This geometric parameterization effectively provides thickness and vertical shear only, eliminating chordwise curvature during optimization. The optimization converged three orders of magnitude in 73.5 hours on one node and increased the range quantity by 126% (Figure 9.4). The large amount of improvement is partly attributable to the small starting battery volume.

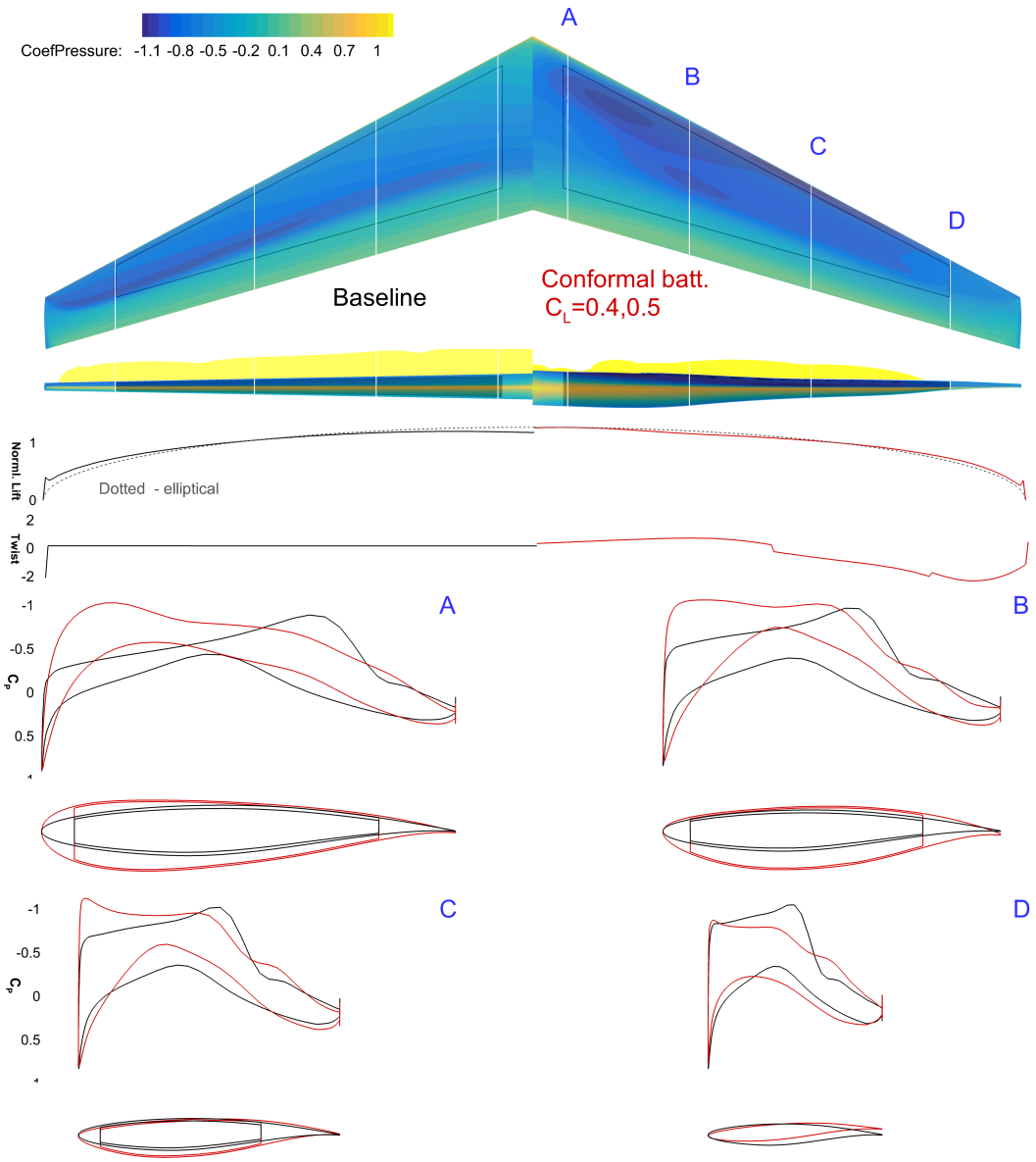


Figure 9.3: Multi-point optimized battery geometry and aerodynamic data compared to baseline

The optimizer again greatly increases inboard thickness and reduces thickness outboard of the battery end. The corners of the batteries fit tightly to the boundaries of the wing cross sections. Figure 9.5 shows the rectangular battery optimized shape compared to the conformal battery optimized shape. There are some subtle differences in wing thickness and shear, but overall the shapes are remarkably similar. Using conformal batteries resulted in a 20% improvement in the range

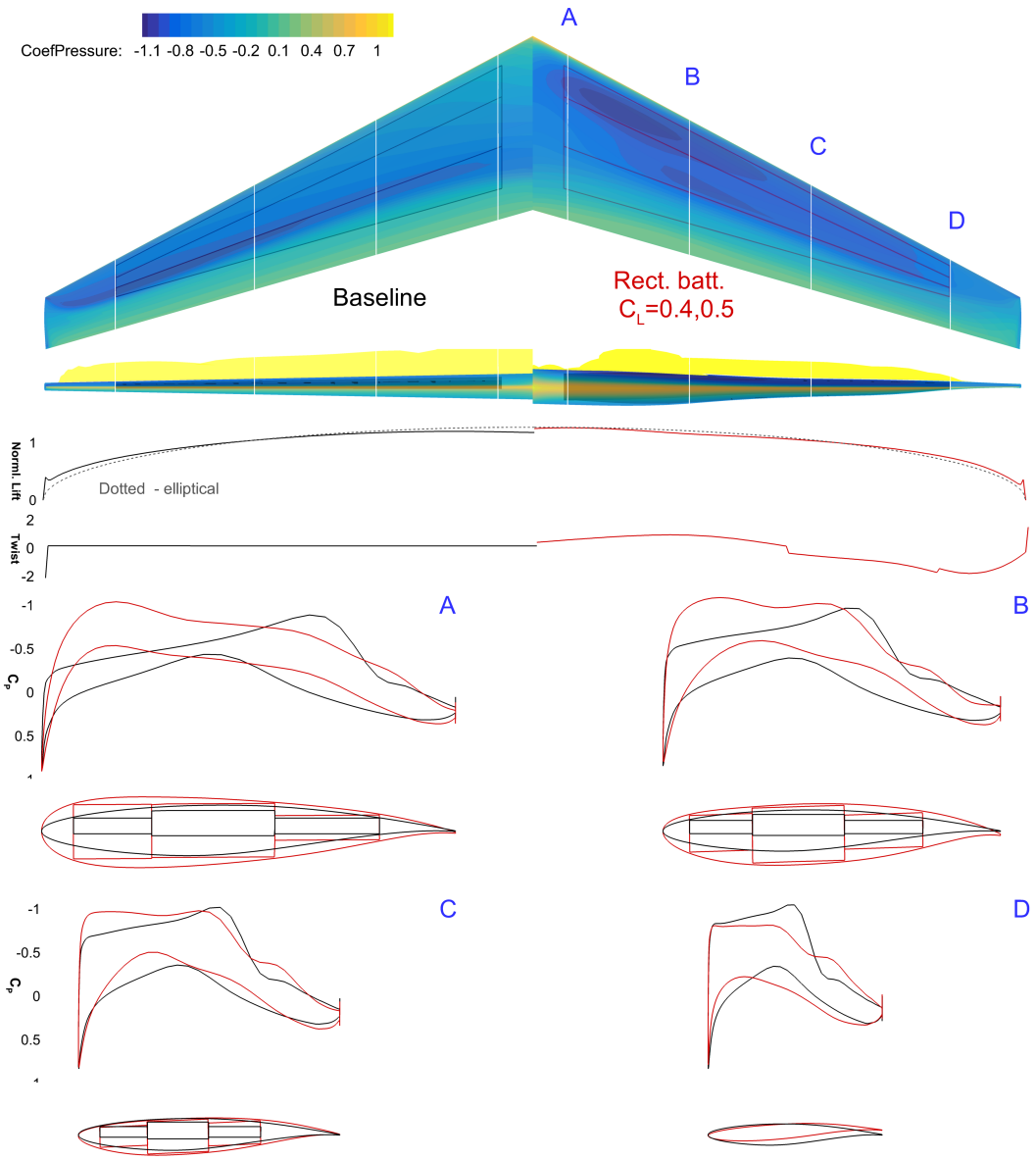


Figure 9.4: Multi-point optimized rectangular battery pack geometry and aerodynamic data compared to baseline

quantity compared to the rectangular restriction.

Figure 9.2 shows the value of the objective function for both problems as optimization progresses. The conformal battery problem converges somewhat earlier (as I saw with wall time). This is likely due to the box corners and edges in the rectangular problem, which makes the KS-

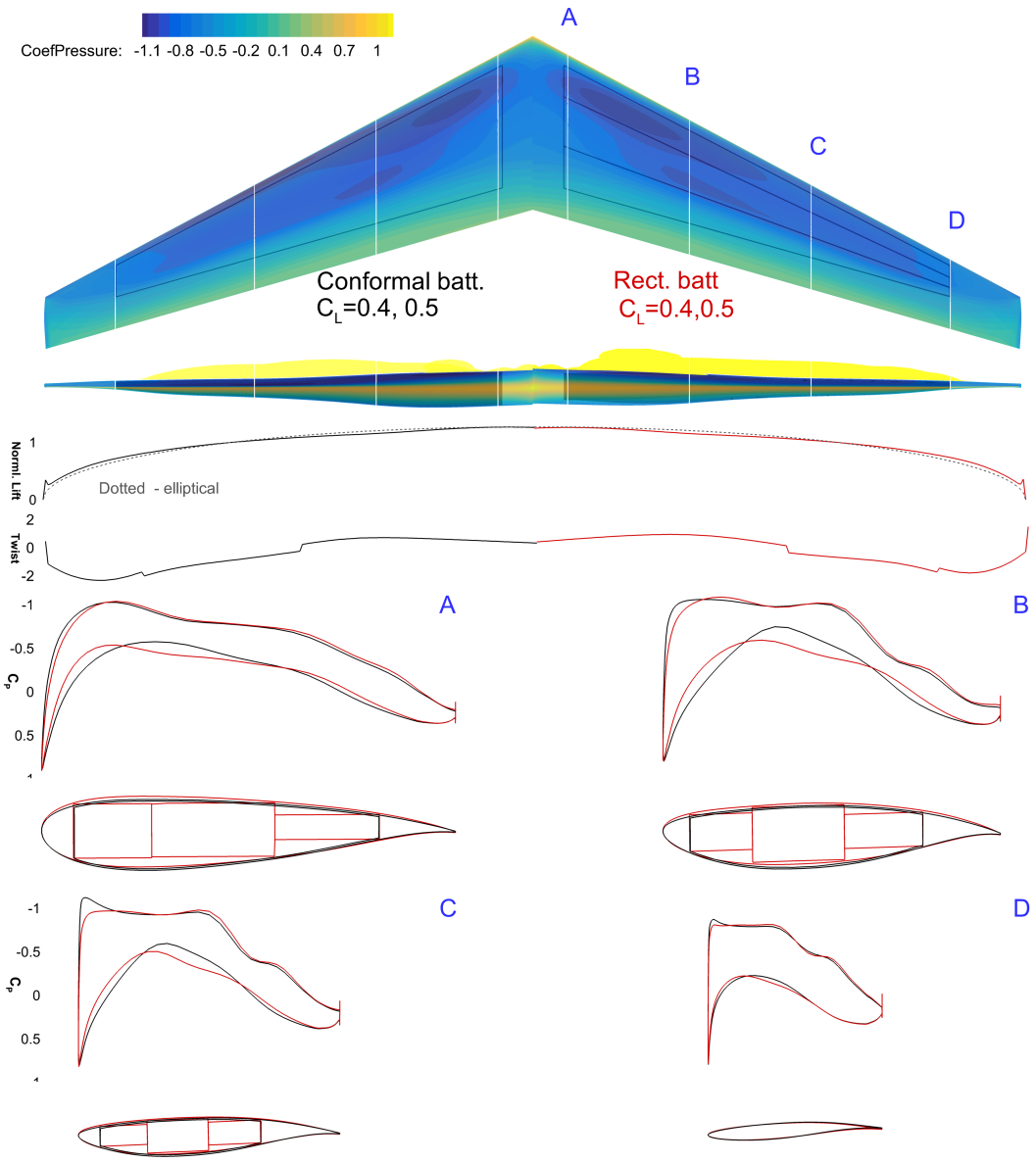


Figure 9.5: Optimized conformal battery pack compared to rectangular

aggregated distance more nonlinear in the design space near the optimal point. Both optimization histories show non-monotonically increasing objectives early on. This is attributable to SNOPT's use of a penalty to force the problem to become feasible. As the problem becomes feasible, the penalty term contributes less to the problem and objective function convergence becomes monotonic.

9.4.3 Effect of Multiple Flight Conditions

I ran an additional optimization case for a conformal battery considering a single flight condition. Some ASO problems produce non-physical results (such as sharp leading edges and shock-free designs) when optimized for a single flight condition. However, in this case, I saw only slight differences in optimized geometry between the single-point and multi-point cases (Figure 9.6). I speculate that the objective function favors preserving volume and thickness in the leading edge, preventing non-physical sharp leading edge solutions even for single-point optimization runs.

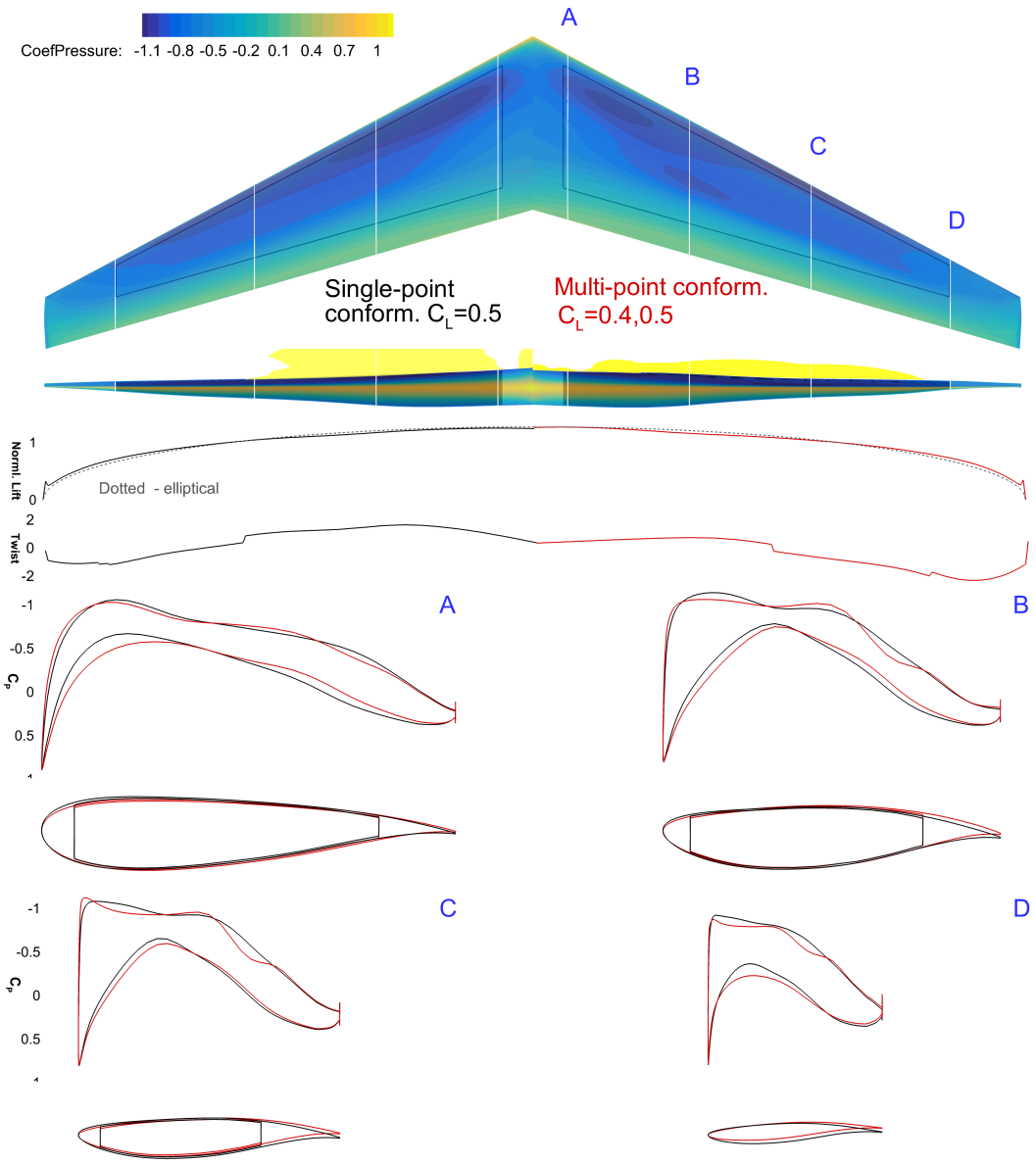


Figure 9.6: Single-point optimized conformal battery compared to multi-point

9.5 Concluding Remarks

In this chapter, I built on my earlier geometric constraints work by performing optimization with both inner and outer geometries changing at the same time. This introduces coupling between aerodynamics and propulsion systems via spatial integration. I optimized the wing of an electric

aircraft for maximum range as a validation case, and was able to find the optimum tradeoff between battery volume and aerodynamic performance. I found that, for this volume-limited transonic problem, allowing batteries to conform more closely to the curvature of the wing skins improves range by about 20% versus a purely prismatic battery pack shape. These results show that the KS-based spatial integration constraint formulation can successfully couple multiple disciplines via spatial integration in high fidelity.

The material in this chapter was previously published [268] and has been lightly edited and formatted.

CHAPTER 10

Aerostructural Optimization of a Wing for a Hydrogen Aircraft

10.1 Introduction

As demonstrated many times in this dissertation, one of the most pressing technical challenges in electric propulsion is increasing range capability. The limited specific energy of today's batteries (on the order of 200 $\mathrm{Wh\,kg^{-1}}$) [17] places unacceptable limits on design range for important applications, such as single-aisle commercial air transport. With continued progress in battery technology, hybrid-electric designs can exploit electric motors to reduce weight, fuel burn, cost, and drag [17], but they still consume fuel and remain a point source of carbon dioxide (CO_2) emissions.

Hydrogen fuel may be an attractive alternative because it can alleviate the range issues of battery-electric propulsion without producing point-source CO_2 emissions. Fuel cells consume hydrogen and produce electricity, emitting only water vapor [33]. Alternatively, hydrogen can be burned in specially-adapted engines, emitting water vapor and trace nitrogen oxides [269]. If the hydrogen is produced cleanly (e.g., through electrolysis powered by renewable energy), it can achieve CO_2-free flying. While the impact (and potential mitigation) of high-altitude water vapor emissions on climate change is not entirely clear, eliminating CO_2 emissions would be a significant

step towards greener aviation.

In this chapter, I introduce an optimization-based method for integrating compressed hydrogen fuel into a wing with minimum weight and drag penalty. It is important to remember that this is not a hydrogen aircraft design study *per se*, but rather validation that the spatial constraint formulation developed in Chapters 8 and 9 performs well on a relevant wing design problem with multiple high-fidelity analysis disciplines. I do not claim to have designed a feasible or desirable airplane at the top level (e.g., I do not consider fuselage design or propulsor design). I also do not claim that design trends observed on this test case are generalizable to other potential hydrogen aircraft configurations.

10.2 A Brief Review of Hydrogen Aircraft Propulsion

Hydrogen fuel in aviation has a surprisingly long history. Soon after liquid hydrogen was first produced for the space program, the NACA experimented with hydrogen combustion aircraft concepts. Silverstein and Hall [270] proposed using hydrogen fuel for a subsonic high-altitude bomber (Figure 10.1) in a declassified 1955 NACA research memo. Even then, it was apparent that integrating the enormous hydrogen tanks into the aircraft would be a significant challenge. From 1957 to 1959, NACA flew a B-57 Canberra bomber (Figure 10.2) converted to run one engine using liquid hydrogen fuel [271]. The airplane was able to transition from jet fuel to hydrogen and back again on numerous successful flights. The pilots noted that the hydrogen-powered engine tended to leave contrails even when the kerosene-powered engine did not. Simultaneously, Kelly Johnson's Skunk Works was seriously considering building a hydrogen-powered supersonic bomber [271]. The study airplane, known as the CL-400 Suntan, was canceled by 1958, but the government learned valuable insights on the safe handling of hydrogen fuels.

After the cancellation of the CL-400, hydrogen aviation fuel was not seriously pursued operationally again. However, the concept is experiencing a renaissance. In the last two decades,

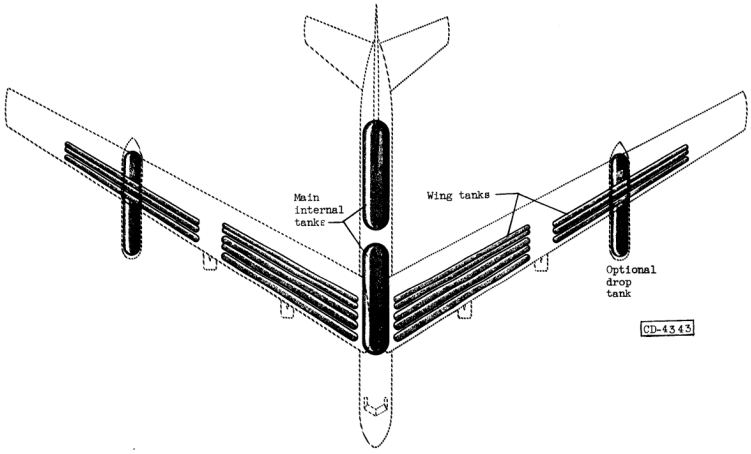


Figure 10.1: The Silverstein–Hall subsonic bomber concept, which used liquid hydrogen tanks in both the wing and fuselage [270]

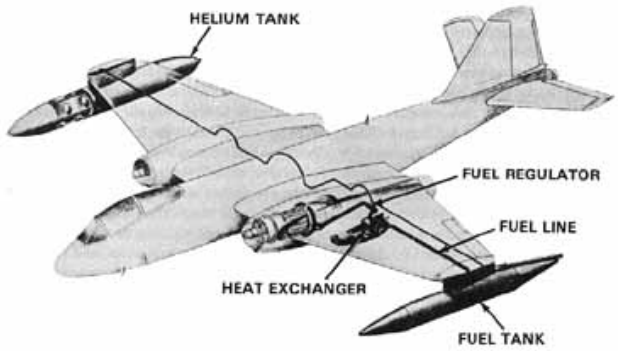


Figure 10.2: NACA's converted B-57 testbed, which used liquid hydrogen to power one engine [271]

prototype hydrogen-powered aircraft have been built and flown, including the Hy4 [272] and Boeing Fuel Cell Demonstrator [33]. NASA has also funded a university consortium to examine liquid hydrogen propulsion for transport aircraft applications [269]. Finally, the concept seems to be gaining some traction in industry, with both startups (ZeroAvia) and incumbents (Airbus) promoting hydrogen as environmentally-friendly alternatives to petroleum fuels [269].

Notwithstanding hydrogen's very high energy mass density, its volumetric density under normal conditions is much lower than jet fuel [269]. Commercial aircraft carry the vast majority

of their fuel in sealed wing tanks without significant aerodynamic penalty; hydrogen at ambient pressure is volume-limited in the same space. Therefore, while hydrogen alleviates some of the weight challenges of green aviation concepts, it creates a new *spatial integration* challenge. As demonstrated in Chapters 8 and 9, packing optimization is a promising way of incorporating spatial integration requirements into aircraft trade studies.

10.3 Problem Description

This work extends the wing packing design optimization problem described in the previous two chapters to the aerostructural domain. Neither of the previous examples involved multidisciplinary physics (e.g., aerodynamic and structural design), and the complexity of the packing problem was simple compared to problems of broader industrial interest. A packing problem with both high-fidelity multidisciplinary physics and many objects to pack has not yet been demonstrated.

Considering structure complicates the wing packing optimization problem in three primary ways. First, the structural analysis enables the airplane model to vary both weight and drag, likely producing a different optimal design than when considering aerodynamics alone. Second, structural members such as spars and ribs significantly increase the packing problem's complexity compared to considering the wing OML only. Finally, each optimization iteration's computational cost significantly increases because an iterative solution to the aerostructural problem is now required, and a coupled adjoint must also be computed.

10.3.1 Wing Description

The baseline wing design is representative of a simple, single-aisle transport aircraft and is identical to the wing featured in the University of Michigan MDO Lab aerodynamic shape optimization tutorial[1]. The wing planform is modeled after the Douglas DC-9, except with RAE2822

[1] https://github.com/mdolab/mach-aero

transonic airfoils at baseline. The wingbox front and rear spar locations generally approximate the proportions of the DC-9 wing structure (Figure 10.3). I generated a conventional wingbox structural layout with 18 evenly-spaced rib bays and 8 stringers between the spars. The stringers each run all the way out to the end of the wing box. The cruise condition is Mach 0.8 at 10,000 m

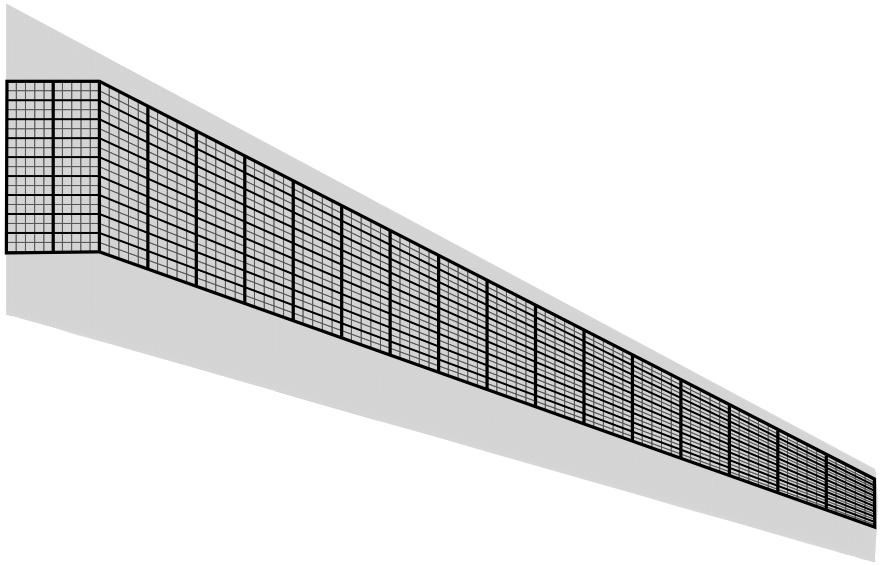


Figure 10.3: Wingbox structural mesh overlaid on OML planform

altitude. The baseline weight at cruise is 54,900 kg, which equates to approximately $C_L = 0.5$. I compute structural loads at a 2.5 g static condition, flown at Mach 0.75 and 5,000 m altitude.

10.3.2 Tank Description

Unlike the previous wing problem that featured a battery, I optimize a set of seven hydrogen tanks in each wing root (one per rib bay). The 700 bar compressed hydrogen tanks are cylindrical with spherical end caps. Each tank can vary in radius and length, with additional variables to position each tank relative to its rib bay (Figure 10.4). The tanks consist of optimized carbon fiber reinforced polymer (CFRP) laminate.

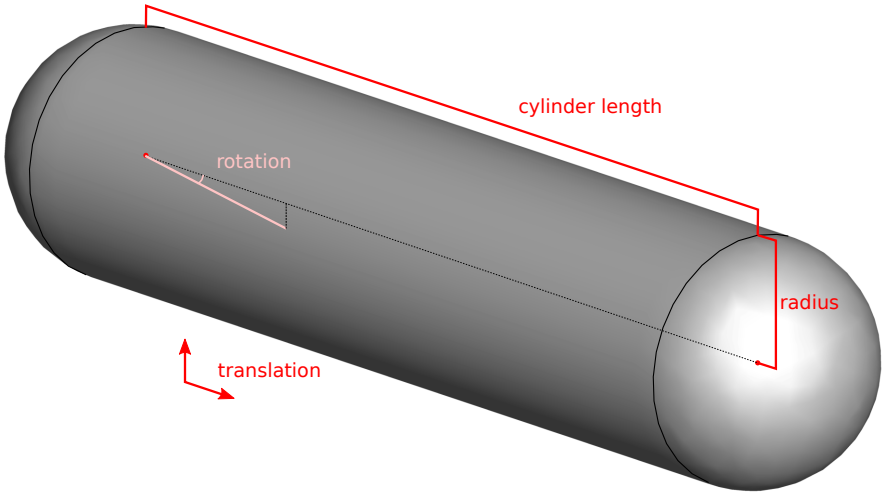


Figure 10.4: A representative compressed hydrogen tank with design variables labeled

10.4 Methodology

To perform aerostructural analysis and design, I used the MACH framework [147]. The MACH framework integrates several high-fidelity analysis tools with geometry engines while propagating design variable derivatives [154]. The subset of aerodynamic shape optimization tools is open-source and freely available.

10.4.1 Aerostructural Analysis

I used the open-source ADflow solver for aerodynamic analysis and derivatives [267]. ADflow is a structured, multiblock, overset RANS solver with discrete adjoint gradients. I use the Spalart–Allmaras turbulence model and an approximate Newton–Krylov solver for this problem [252]. The aerodynamic mesh (Figure 10.5) consists of approximately 800,000 volume cells and was generated using pyHyp [251], an open-source implementation of the hyperbolic scheme described by Chan and Steger [258]. The aerodynamic solver settings and mesh are virtually identical to those in the MACH aerodynamic shape optimization tutorial.

For structural analysis, I use the open-source finite element solver TACS [273]. TACS com-

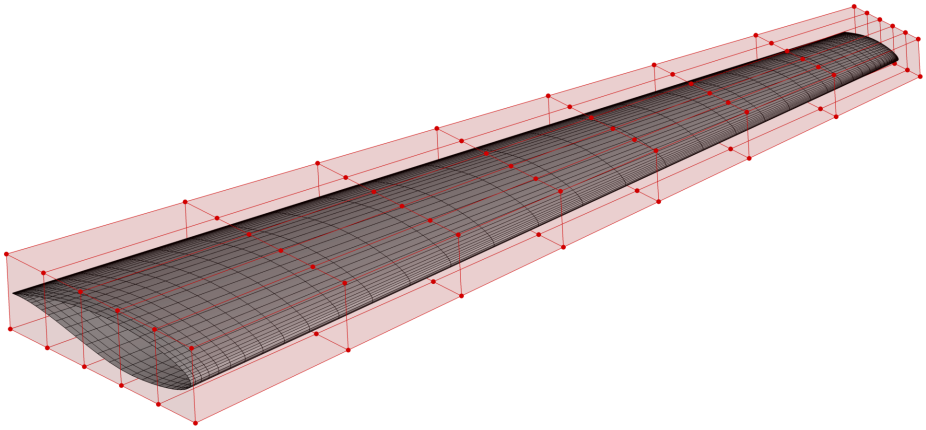


Figure 10.5: Aerodynamic surface mesh pictured with FFD control points

putes efficient adjoint derivatives with respect to the structural sizing (thickness) variables and geometry. The structural mesh consists of 7,632 CQUAD4 elements (Figure 10.3), with explicitly-modeled stringers. I only performed a linear static analysis in this scenario, though TACS supports geometric nonlinearity and buckling.

Because structural deflections affect the aerodynamic surface and vice-versa, an aerostructural solver is required. I use a block Gauss–Seidel approach to solve the aerostructural analysis and a Krylov method to solve the coupled adjoint [154].

10.4.2 Geometry

I use two different geometry engines in this problem; one for the wing and one for the hydrogen tanks. The wing was parameterized using the free-form deformation (FFD) method [274] using the open-source implementation in pyGeo [228]. The FFD volume (Figure 10.5) is identical to the one generated in the MACH aerodynamics tutorial and contains 96 design variables. Both the CFD surface mesh coordinates and the structural elements are embedded in the same FFD volume, so geometric displacements are always consistent between the two. CFD surface mesh displacements are propagated to the volume using IDwarp, an open-source implementation of the inverse-distance weighted warping scheme from [257].

I use a different approach to parameterize the tank geometry. An initial triangulated representation of each tank surface is generated using Engineering Sketch Pad (ESP) [275], an open-source computer-aided design (CAD) application. Using a Python wrapper around ESP's OpenCSM library, I map each point on the tank surface onto the CAD B-spline surface in parametric coordinates and save the result. When the geometry is perturbed, I retrieve a new set of surface mesh points using the same parametric coordinates. This way, the topology of the mesh is preserved across geometric perturbations. A more detailed account of the ESP geometry engine is included as Appendix C. The design variables are as pictured in Figure 10.4—five variables per tank, for a total of 35.

Gradients with respect to geometric design variables are computed using a parallel finite differencing approach. Since the aerostructural design optimization is done in a HPC environment with 100 or more available cores, I can perform finite differences across dozens of geometry variables without incurring much cost in terms of wall time (a few seconds). The advantages of using this open-source CAD package were readily apparent at several points. For example, I was able to edit the source to suppress certain console output, which, while useful in interactive mode, clogs the output when running dozens of instances simultaneously. Open source code also made the Python wrapper possible, which was indispensable for this project.

Like the study in Chapter 9, spatial integration constraints are computed using the software package `geograd`. The software computes two metrics for each pair of objects to be packaged: the KS aggregated distances between all the triangular facet pairs and the length of the intersection curve(s) between the two objects (if any). As detailed in Chapter 8, when adequately constrained in the optimization these metrics produce spatially-feasible packing solutions. Because minimum distance computations on geometry have poor scaling properties when done naively, careful attention to detail is required in the implementation.

There is a tuning parameter, ρ in the KS function, which controls how conservative the constraint aggregator is. Choosing a larger ρ gives a more exact result for the constraint, but it increases

the constraint curvature. Excessively-large ρ will prevent the optimizer from taking efficient steps near the active zone due to the aggressive curvature. Unless otherwise specified, I used $\rho = 250$ for all the described cases below.

The TensorFlow implementation in the previous work was well-suited to a desktop environment with a GPU; however, its CPU runtime was not scaling well. For this study, I re-implemented the TensorFlow-based `geograd` described in Chapters 8 and 9 in FORTRAN for improved performance. The new Fortran implementation enables orders of magnitude speedup via heuristics such as bounding-box testing. I compute derivatives of the outputs with respect to surface mesh inputs using the Tapenade automatic differentiation tool [276]. I wrapped the Fortran codebase in Python using the f2py utility [277]. The details of the software implementation, performance benchmarks, and scaling data are included in Appendix B.

10.4.3 Optimization

To solve the MDO problem, I use the gradient-based nonlinear optimizer SNOPT 7.7.5 [222, 278]. SNOPT is wrapped with the Python interface pyOptSparse [223]. I exploit a unique feature of SNOPT in this problem. In my experience, optimizers often take steps that are too big early on in a packing problem, before a good quasi-Newton estimate of the constraint curvature is built up in the optimizer. To prevent unreasonably large steps from consuming wall time, `geograd` passes a "fail" flag to SNOPT if any packing objects intersect by more than a moderate tolerance during a step. When this happens, I also prevent the aerostructural solution or adjoints from running. SNOPT then backtracks by a factor of ten and continues. Significant time savings can be achieved this way.

Because I am using constraint aggregation for both the structural failure constraints and spatial integration, the optimization problems have aggressive curvature near the optimum. This is a challenging scenario for the optimizer. The optimizer computes an internal optimality criterion, and it usually expects this figure to drop by six orders of magnitude for successful convergence. I

found qualitatively that the optimality metric stays artificially high when constraint aggregation is used, preventing the optimizer from a "normal" exit. Criteria for run convergence in this chapter include:

- Optimality metric on the order of 10^{-3} or less
- Feasibility metric on the order of 10^{-4} or less
- Diminished continued improvement (per-iteration improvement on the order of 0.01%)

I ran the cases using two or three Intel Skylake nodes on the Stampede2 supercomputer at the University of Texas—a total of 96 to 144 physical processors.

10.5 Results

This section contains a progression of optimization results for the wing-tank problem with increasing complexity. First, I kept the wingbox fixed and optimized the tank shape only for maximum volume. Next, I allowed the wing outer mold line to vary, but considered only packing and aerodynamics, and minimized drag subject to a fuel volume constraint. Then, I ran a series of aerostructural optimization cases with all design variables active.

10.5.1 Optimizing Tank Shape Only

As a simple test, I exercised the CAD-based tank geometry by maximizing the volume of the tanks V_{tank} within the baseline wing box. Since there was no high-fidelity analysis in the loop, I used a Linux desktop for this case. Because there were fewer design variables and constraints, I chose a slightly more aggressive $\rho = 300$. I found that the maximum achievable volume was approximately 1.718 m^3 and the optimizer converged without difficulty. Table 10.1 lists the design variables, objective, and constraints and Figure 10.6 illustrates the resulting geometry (in black). The tanks expand to efficiently fill the wingbox, as expected.

		Quantity	Lower	Upper	Scaling
maximize	V_{tank}	1			1.0
with respect to	Tank radius	7	0.14 m	3.0 m	0.1
	Tank length	7	1.0 m	2.5 m	0.1
	Tank x-y offset	14	-0.2 m	0.2 m	0.1
	Tank tilt	7	-1.0 °	4.0 °	1.0
	Total number of design variables	**35**			
subject to	Packing (aggregated distance)	7	0.0 m		100
	Intersection perimeter	7		0.0 m	1.0
	Total number of constraints	**14**			

Table 10.1: Problem formulation: tank shape optimization for maximum volume

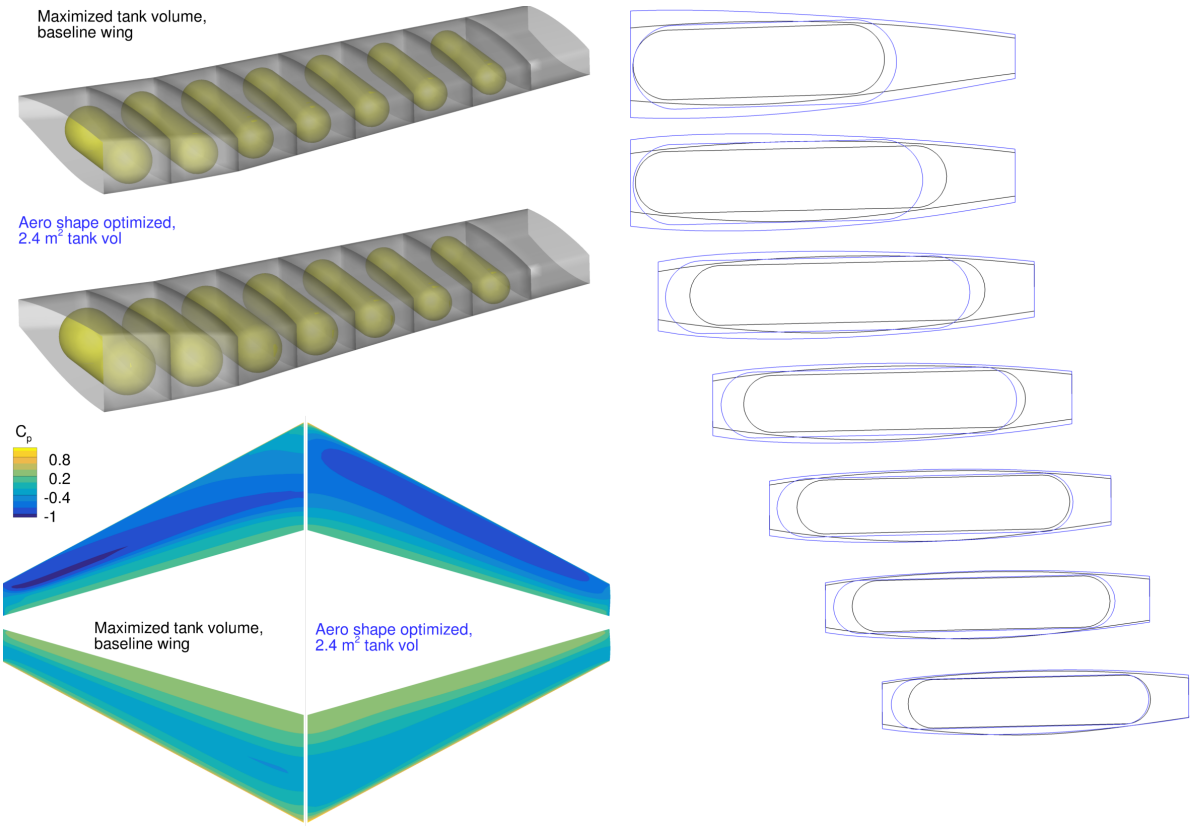


Figure 10.6: Solutions to the volume maximization and minimum drag subproblems

10.5.2 Optimizing Wing OML and Tank Shape

Next, I increased the degree of difficulty substantially by adding aerodynamic physics. I added the angle of attack and outer mold line shape variables to the problem and include a constraint

		Quantity	Lower	Upper	Scaling
minimize	D	1			1.0
with respect to	Angle of attack	1	0 °	10 °	0.1
	Sectional shape	96	-0.5	0.5	1.0
	Tank radius	7	0.14 m	3.0 m	0.1
	Tank length	7	1.0 m	2.5 m	0.1
	Tank x-y offset	14	-0.2 m	0.2 m	0.1
	Tank tilt	7	-1.0 °	4.0 °	1.0
	Total number of design variables	**132**			
subject to	$L - W$ (cruise)	1	0 N	0 N	$1/W$
	V_{tank}	1	2.4 m^3		1.0
	Wingbox thickness (vs. baseline)	100	1.0		1.0
	Packing (aggregated distance)	7	0.0 m		100
	Intersection perimeter	7		0.0 m	1.0
	Total number of constraints	**116**			

Table 10.2: Problem formulation: aerodynamic shape optimization of wing and fuel tanks for minimum drag

balancing lift L and weight W (a notional cruise condition). Because they have entirely separate geometric parameterizations, the inner tanks and OML are only coupled via the spatial integration constraints. The objective is to minimize drag D subject to a minimum hydrogen fuel volume constraint of 2.4 m^3 (arbitrary, but intended to push out the OML a reasonable amount). The KS parameter ρ remains at 300 for comparison to the volume-only case. The problem summary is in Table 10.2 and the optimization result is included in Figure 10.6 (in blue). We see that the inboard OML grows substantially to accommodate larger diameter tanks. The optimizer strategically adds thickness near the aft spar to allow tanks to expand rearward. The far outer part of the wing is constrained by minimum thickness.

10.5.3 Aerostructural Optimization at Fixed Tank Volume

This case adds structural analysis, including a 2.5 g maneuver condition (Figure 10.8), and structural design variables to the previous problem. In the absence of a whole-airplane performance model, this aerostructural optimization is a multiobjective problem between wing weight and drag.

I choose $2m_{\text{wing, struct}} + D$ as a reasonable compromise objective function. Table 10.3 describes the problem formulation.

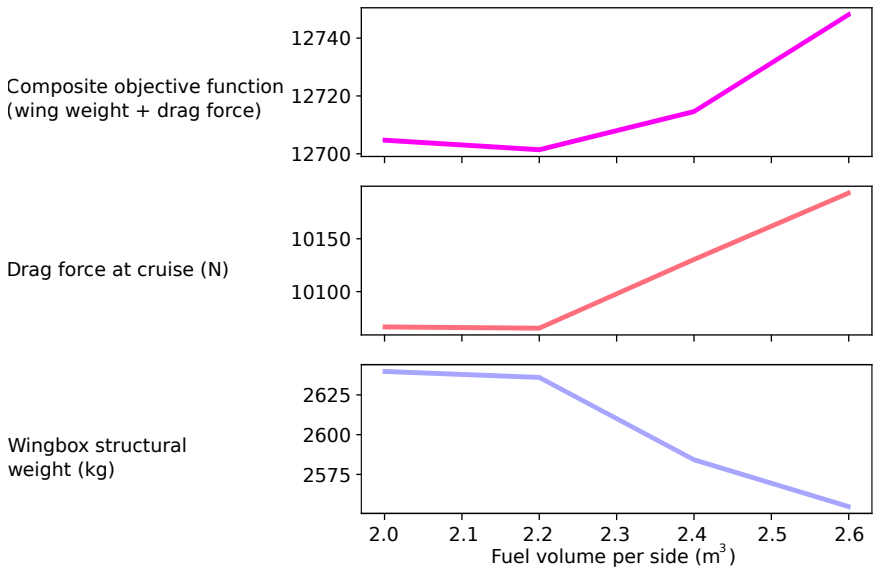


Figure 10.7: Increasing fuel storage in the wing requires an aerostructural penalty, though aerodynamics and structure move in opposite directions

Consider an aircraft design team that wishes to know how adding fuel volume in the wing might affect weight and aerodynamic performance. This is an important piece of information when deciding how to allocate fuel throughout the whole airplane. We can answer this problem using packing optimization. I ran the problem in Table 10.3 at several fuel volume requirements, from 2.0 to 2.6 m^3. Figure 10.7 shows how the aerostructural performance of the airplane changes as fuel volume is added. At 2.2 m^3 or less, the wing aerostructural optimum has enough room to accommodate the tanks without affecting the design. At 2.4 m^3 tank volume and above, the optimizer can no longer accommodate the tanks without pushing out the OML, incurring a sharp rise in drag. The drag increase is partially offset by a structural weight *decrease* due to the increased structural depth at the side-of-body.

This example illustrates why it is essential to consider multiple disciplines when evaluating spatial integration tradeoffs. Without optimization, engineers would need to perform laborious

analysis and iterate internally to achieve a good result. Using an optimization framework, each of the runs used 400 to 600 core-hours on the HPC (four to six hours wall time each). If the cases run in parallel on an in-house or cloud HPC service, it is easily conceivable that the data for this trade study could be collected and analyzed in one working day.

Figure 10.9 illustrates the solution for 2.4 m^3 fuel volume (in black).

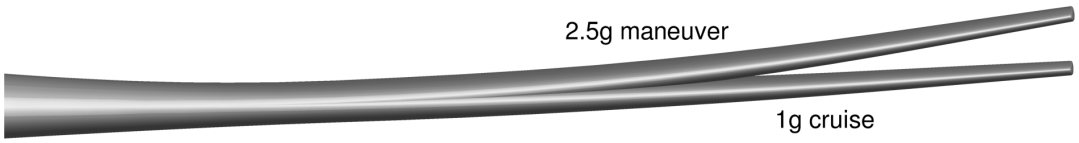


Figure 10.8: Aeroelastic solutions at the cruise and maneuver conditions

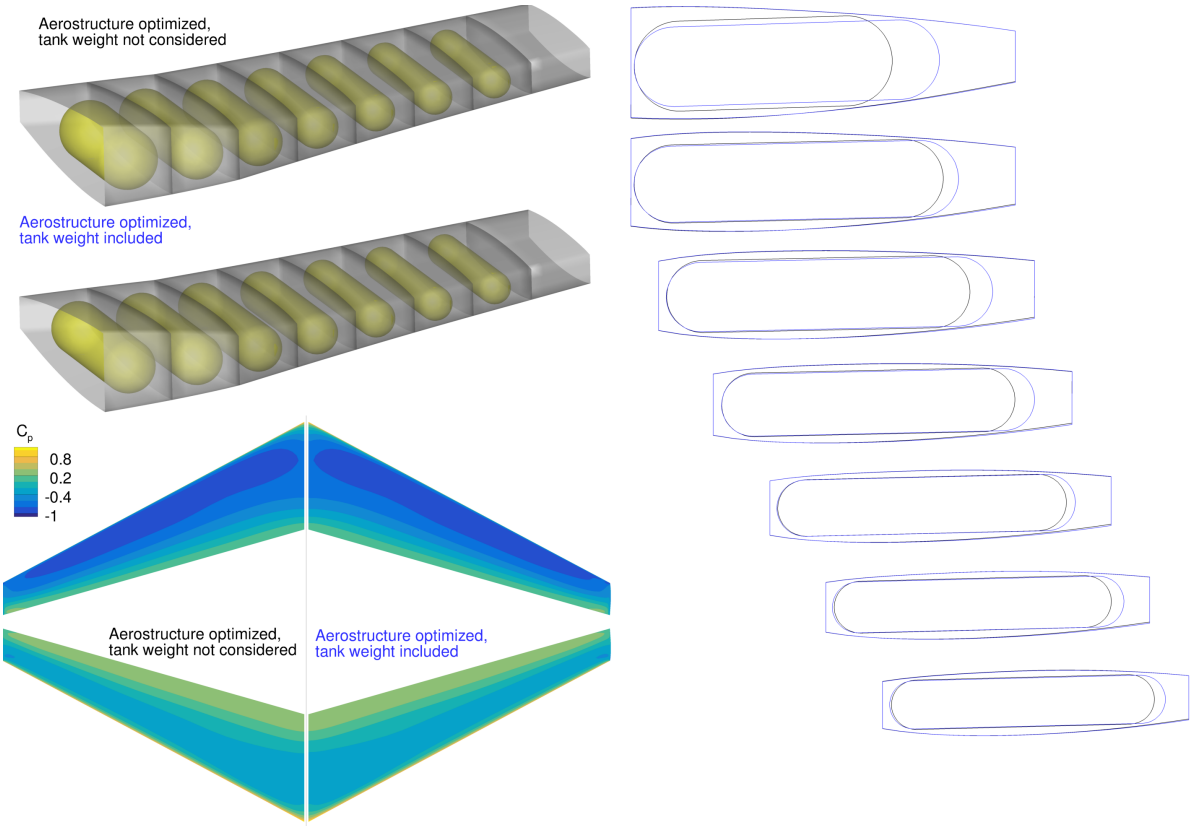


Figure 10.9: Solutions to the aerostructural problem with and without considering the weight of the tanks

		Quantity	Lower	Upper	Scaling
minimize	$2m_{\text{wing, struct}} + D$	1			1.0
with respect to	Angle of attack	2	0 °	10 °	0.1
	Stringer thickness	54	0.0016 m	0.02 m	100.0
	Spar thickness	18	0.0016 m	0.02 m	100.0
	Skin thickness	18	0.0016 m	0.02 m	100.0
	Rib thickness	18	0.0016 m	0.02 m	100.0
	Sectional shape	96	-0.5	0.5	1.0
	Tank radius	7	0.14 m	3.0 m	0.1
	Tank length	7	1.0 m	2.5 m	0.1
	Tank x-y offset	14	-0.2 m	0.2 m	0.1
	Tank tilt	7	-1.0 °	4.0 °	1.0
	Total number of design variables	**241**			
subject to	$L - W$ (cruise)	1	0 N	0 N	$1/W$
	$L - 2.5W$ (maneuver)	1	0 N	0 N	$1/W$
	V_{tank}	1	various m^3		1.0
	Structural failure at 2.5 g (aggregated)	3		1.0	1.0
	Wingbox thickness (vs. baseline)	100	1.0		1.0
	Packing (aggregated distance)	7	0.0 m		100
	Intersection perimeter	7		0.0 m	1.0
	Total number of constraints	**120**			

Table 10.3: Problem formulation: aerostructural design optimization of wing and fuel tanks with composite objective

10.5.4 Aerostructural Optimization Considering Tank Weight

While the previous subsections showed a series of successful aerostructural optimization packing cases, I have still neglected a significant effect: hydrogen tank weight. Because of the extreme pressure and low density of the compressed fuel, even a CFRP composite tank will have a hydrogen fuel fraction significantly less than ten percent [279]. The radius and length of the tank will affect its weight significantly.

Offline, I set up a structural optimization problem to minimize the weight of a tank made from a bidirectional carbon fiber laminate, considering both axial and hoop stresses [2]. The method uses a simplified classical laminate theory model and computes the required laminate thickness in the cylindrical portion of the tank subject to hoop and axial stress. I compared the optimization model to a detailed finite element study of a CFRP tank [279] and found that the structural weight estimate was within 15% of the published value [3]. At 700 bar and 2.35 burst pressure safety factor, using an optimal bidirectional laminate with Toray 1100G prepreg [280], I found that the optimal tank wall thickness is a constant 0.1315 times the tank radius. Therefore, I did not need to explicitly incorporate tank structural analysis into the optimization—only a weight calculation based on tank radius and length. The density of the CFRP material is 1,573 kg/m^3. The problem formulation is summarized in Table 10.4.

The resulting geometry is visualized in Figure 10.9 (in blue). While the OML only changes subtly at the lower trailing edge, the changes allow the tanks to become much longer and narrower, reducing hoop stress and tank weight. This is a complex tradeoff between the structural weight of a component and the structural weight and drag at the airplane level. It is a good illustration of MDO's potential to find non-obvious solutions in airplane trade studies rapidly.

Figure 10.10 shows the structural sizing variables for this case. Some of the structural zones are minimum gauged, such as the ribs and some spar web zones. Figure 10.11 shows the structural

[2] https://gist.github.com/bbrelje/b599102f2d83749df681dd5c2c0865e1
[3] https://gist.github.com/bbrelje/947ef6ff401a201812fde465518b74ff

		Quantity	Lower	Upper	Scaling
minimize	$2(m_{\text{wing, struct}} + m_{\text{tanks}}) + D$	1			1.0
with respect to	Angle of attack	2	0 °	10 °	0.1
	Stringer thickness	54	0.0016 m	0.02 m	100.0
	Spar thickness	18	0.0016 m	0.02 m	100.0
	Skin thickness	18	0.0016 m	0.02 m	100.0
	Rib thickness	18	0.0016 m	0.02 m	100.0
	Sectional shape	96	-0.5	0.5	1.0
	Tank radius	7	0.14 m	3.0 m	0.1
	Tank length	7	1.0 m	2.5 m	0.1
	Tank x-y offset	14	-0.2 m	0.2 m	0.1
	Tank tilt	7	-1.0 °	4.0 °	1.0
	Total number of design variables	**241**			
subject to	$L - W$ (cruise)	1	0 N	0 N	$1/W$
	$L - 2.5W$ (maneuver)	1	0 N	0 N	$1/W$
	V_{tank}	1	2.4 m^3		1.0
	Structural failure at 2.5 g (aggregated)	3		1.0	1.0
	Wingbox thickness (vs. baseline)	100	1.0		1.0
	Packing (aggregated distance)	7	0.0 m		100
	Intersection perimeter	7		0.0 m	1.0
	Total number of constraints	**120**			

Table 10.4: Problem formulation: aerostructural design optimization of wing and fuel tanks with composite objective, considering tank weight

failure criterion at the 2.5 g maneuver case. We can see that the optimizer has removed material almost everywhere until most of the wingbox is nearly at failure at ultimate load (2.5 g plus 1.5 safety factor).

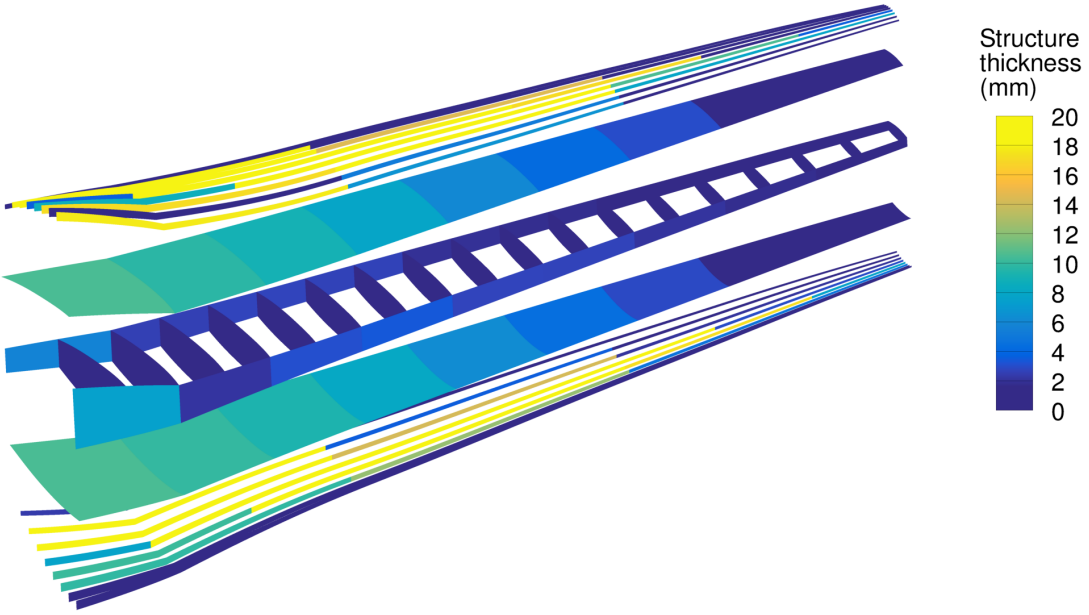


Figure 10.10: Structural sizing variables at the optimum (with tank weight, 2.4 m^3 fuel volume)

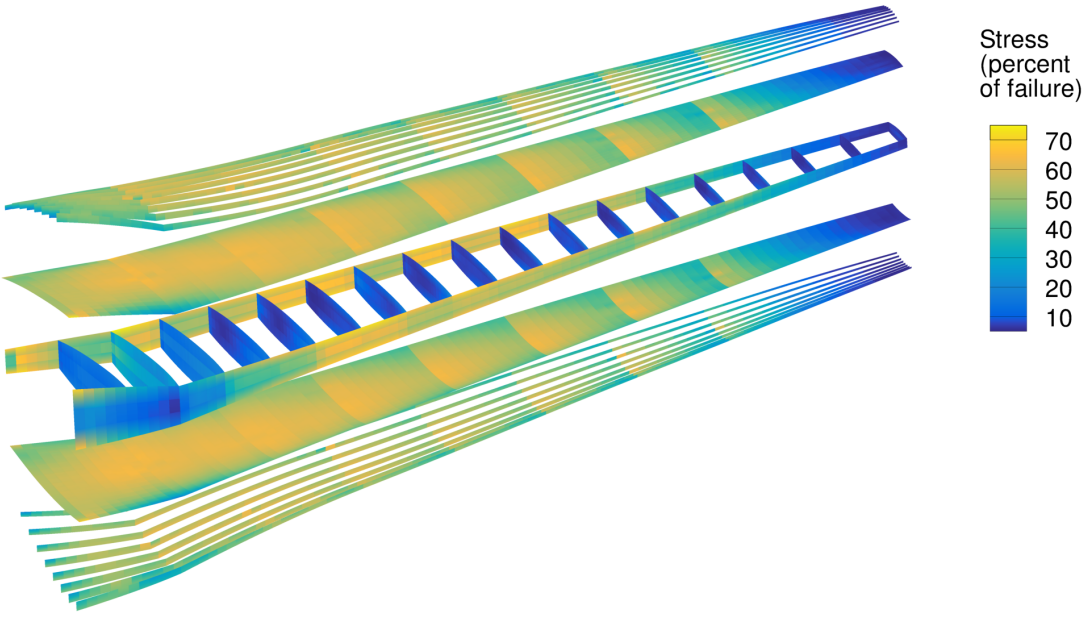


Figure 10.11: Structural failure criterion at the optimum (with tank weight, 2.4 m^3 fuel volume)

10.5.5 Aerostructural Optimization for Maximum Range

In the previous example, I relied on a composite objective function in the absence of an airplane-level performance model and assumed a given fuel volume. We can make some assumptions and gain some intuition on whether placing compressed hydrogen in the wing of this test case airplane is viable. For the new objective function, let us maximize the range subject to a design payload.

Since I don't have an airplane-level empty weight model, I assume that the aircraft's operating empty weight (OEW) is 53% of MTOW and that the design payload is 23% of MTOW. I subtract the wing structure's weight from the previous optimum point (2,586 kg) and obtain OEW-less-wing of 26,511 kg. I then attribute the remaining 24% weight to fuel. A small fraction of the fuel is wing fuel from the previous optimum point (100.8 kg hydrogen at 42 kg/m^3 plus 1070.7 kg tank weight per wing). The remainder is computed at 9% hydrogen weight fraction and installed in the fuselage (975 kg hydrogen plus 9858 kg tank weight).

The actual TOW can now be computed as:

$$m_{\text{TO}} = m_{\text{payload}} + \text{OEW}_{\text{less wing}} + m_{\text{fuselage fuel}} + m_{\text{fuselage tank}} + 2(m_{\text{wing struct}} + m_{\text{wing tank}} + m_{\text{wing fuel}}) \tag{10.1}$$

and the zero fuel weight as:

$$m_{\text{zero fuel}} = m_{\text{payload}} + \text{OEW}_{\text{less wing}} + m_{\text{fuselage tank}} + 2(m_{\text{wing struct}} + m_{\text{wing tank}}) \tag{10.2}$$

An obvious limitation of this simplified model is that it neglects OEW growth outside the wing due to MTOW growth.

With pre- and post-mission weights, the range can be computed using the Breguet range equation as follows:

$$R = \frac{U_\infty}{g}\frac{L}{D}\frac{1}{1000\,\mathrm{SFC}} ln \frac{m_{\mathrm{TO}}}{m_{\text{zero fuel}}} \tag{10.3}$$

where R is the range in km, g is the gravitational constant, U_∞ is the true airspeed in m/s, SFC is the specific fuel consumption in kg/Ns (assumed to be 1.5×10^{-5}, slightly better than published figures for previous-generation single aisle turbofans). I can correct this SFC to account for the higher heating value of hydrogen compared to kerosene (nearly three times greater), but it won't affect the optimization result. With the heat value correction, the previous optimum airplane is computed to have a range of about 1900 km at design payload—much less than contemporary single aisles, but much more than all-electric proposals. I can now optimize the airplane with respect to this objective function as described in Table 10.5.

Figures 10.12 and 10.13 illustrate that, for this demonstration airplane, it is favorable to add more fuel tank volume in the wing root, even after considering the aerodynamic, structural, and weight penalties using high-fidelity physics. The optimizer increased each wing's fuel volume from 2.4 to 3.1 m^3.

This is a technology validation study for the optimization approach, not an airplane study per se. The CFD and structural meshes were reasonably coarse, and the airplane-level weight model was of very low fidelity. I was not explicitly modeling the fuel cell or electric propulsion system. Nonetheless, the result suggests that storing compressed hydrogen in the wing root of a transport-class airplane can, in principle, be favorable even after considering aerodynamic and structural penalties. While this result cannot and should not be generalized to other configurations, it illustrates that compressed hydrogen may be an intriguing fuel for regional-length missions.

		Quantity	Lower	Upper	Scaling
maximize	range (km)	1			0.001
with respect to	Angle of attack	2	0 °	10 °	0.1
	Stringer thickness	54	0.0016 m	0.02 m	100.0
	Spar thickness	18	0.0016 m	0.02 m	100.0
	Skin thickness	18	0.0016 m	0.02 m	100.0
	Rib thickness	18	0.0016 m	0.02 m	100.0
	Sectional shape	96	-0.5	0.5	1.0
	Tank radius	7	0.14 m	3.0 m	0.1
	Tank length	7	1.0 m	2.5 m	0.1
	Tank x-y offset	14	-0.2 m	0.2 m	0.1
	Tank tilt	7	-1.0 °	4.0 °	1.0
	Total number of design variables	**241**			
subject to	$L - W$ (cruise)	1	0 N	0 N	$1/W$
	$L - 2.5W$ (maneuver)	1	0 N	0 N	$1/W$
	Structural failure at 2.5 g (aggregated)	3		1.0	1.0
	Wingbox thickness (vs. baseline)	100	1.0		1.0
	Packing (aggregated distance)	7	0.0 m		100
	Intersection perimeter	7		0.0 m	1.0
	Total number of constraints	**119**			

Table 10.5: Problem formulation: aerostructural design optimization of wing and fuel tanks for maximum range

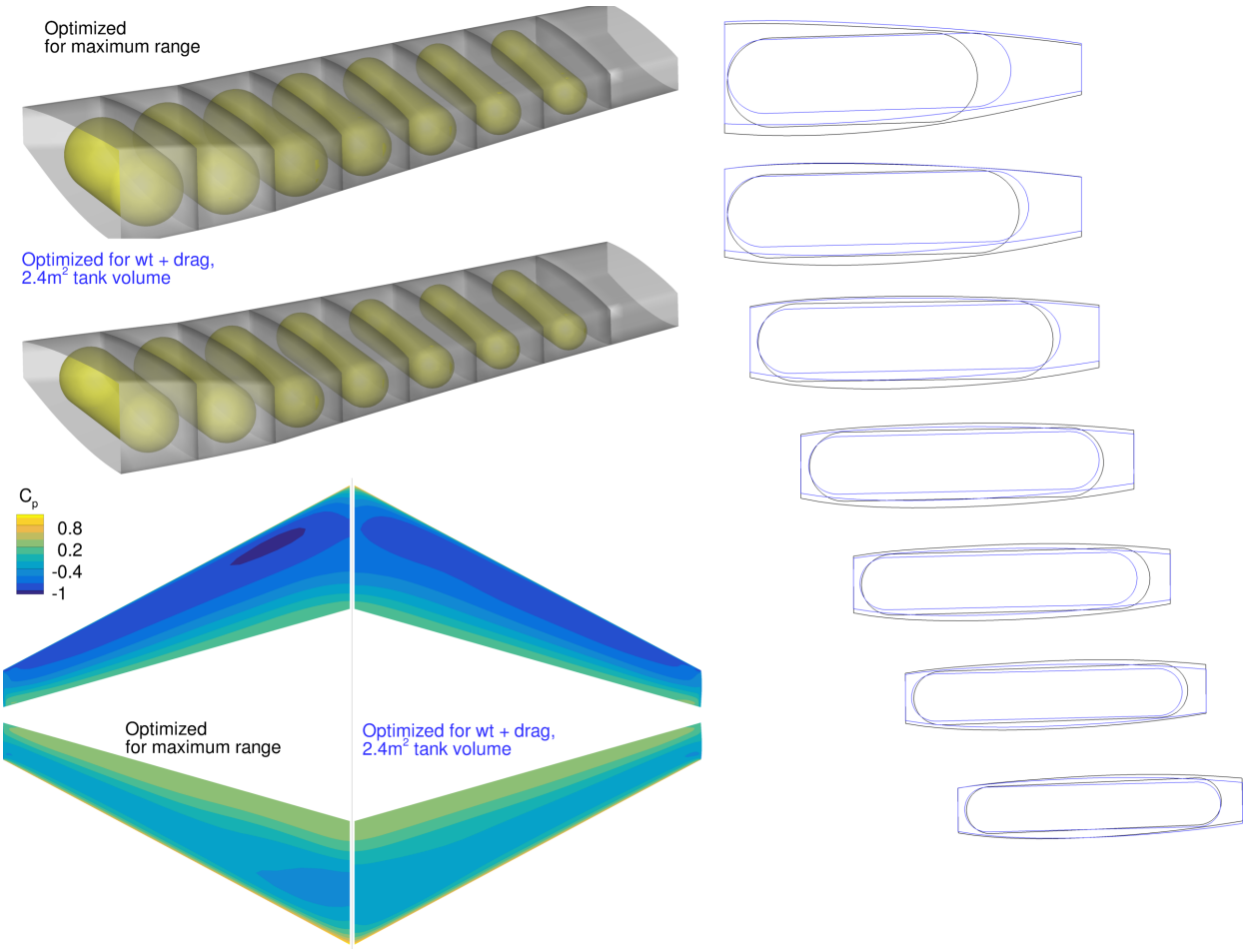


Figure 10.12: The maximum range optimization problem adds significant fuel volume and root thickness

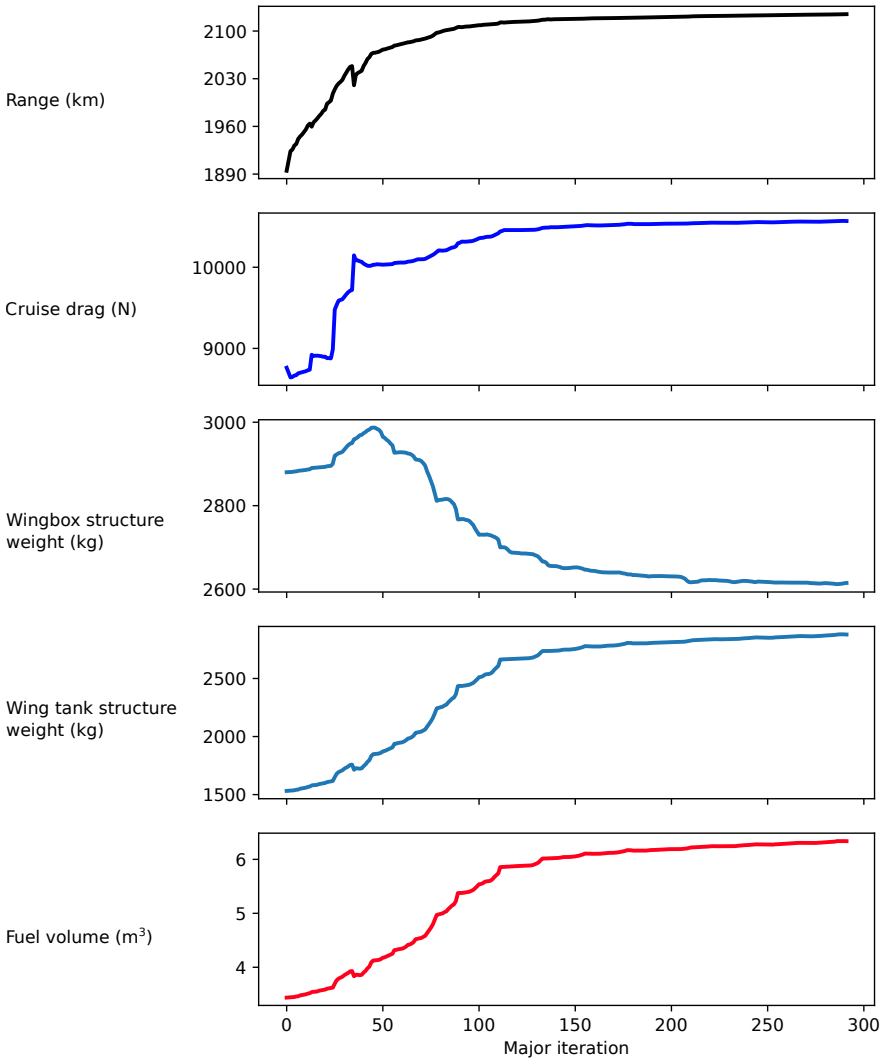


Figure 10.13: The optimizer increases range at the expense of tank weight and drag by adding fuel volume

10.6 Concluding Remarks

Hydrogen fuel is an intriguing option for zero-emissions aviation at typical commercial transport ranges. However, its extremely low density and difficult options for mitigating this fact (high pressure or deep cryogenics) create spatial integration challenges for airplane designers. It also provides a relevant test case for this technical approach to multidisciplinary packing problems. By solving a series of aerostructural packing optimization problems involving a wing and several

hydrogen fuel tanks, I have validated the approach on a problem of significantly greater complexity than previously demonstrated. I showed that MDO can be used to perform spatial integration trade studies on a relevant wing design using high-fidelity aerostructural physics and the new KS-distance geometry constraint. I found that the aerostructural packing optimization runs can be performed on relevant time scales (less than one eight-hour shift using modest HPC resources). The optimization runs converge with regularity and identify subtle changes in OML shape that enable major improvements in system capability.

Despite the limitations acknowledged above, the results suggest that storing compressed hydrogen in the wing root of a single-aisle transport *may* be a useful option at the airplane level. Compressed hydrogen may be feasible for regional-length missions outside the reach of battery-electric airplanes. However, the immense weight of compressed hydrogen storage, even under the relatively optimistic weight assumptions made here, is a significant drawback of the compressed hydrogen approach and probably forecloses the possibility of using it for transcontinental routes. Other potential zero-carbon fuels (such as ammonia) should be explored in parallel.

CHAPTER 11

Concluding Remarks

11.1 Key Results

The first half of the dissertation was focused on system modeling and optimization for electric aircraft. Chapter 2 provided a comprehensive review of the electric propulsion landscape, including concepts, prototypes, studies, and design considerations. In Chapter 3, I briefly reviewed aircraft conceptual MDO in general, then detailed prior attempts at MDO for electric aircraft applications.

Chapter 4 introduced methods and models for modeling, simulating, and optimizing aircraft with electric propulsion. These methods were publicly released as OpenConcept, a new open-source software framework built on OpenMDAO. OpenConcept is the first open-source tool for MDO of aircraft with electric propulsion architectures that is easily user-extensible and incorporates efficient, accurate gradients. It is fast, capable of performing broad tradespace studies with many MDO runs. I demonstrated OpenConcept's capabilities with a case study involving a series hybrid conversion of the Beechcraft King Air C90. For this set of mission requirements, there were tipping points where electric, hybrid, and conventional architectures become favored, depending on the range required and battery technology. OpenConcept also computed a breakeven line where the hybrid / electric aircraft costs the same to operate as a conventional version on the same mission. The breakeven line shifts to a lower battery technology level (or longer range) if the airframe and aerodynamics improve — this confirms that the viability of an electric aircraft depends on

much more than just the battery technology.

While the King Air trade study initially did not consider thermal effects, Chapter 5 added thermal models and methods to OpenConcept. I introduced physics-based models for tracking component temperatures over time and removing heat from electrical machines. Models for ducted heat exchangers were developed, including a low-cost incompressible model as well as a more costly compressible model that can analyze the Meredith effect (drag offset due to waste heat). I repeated the Chapter 4 trade study, but added thermal constraints and TMS design variables in order to isolate the performance impact of thermal management. The results of the revised study indicate that the cost of thermal management varies greatly throughout the design space, with tipping points similar to the original study. While developing intuition about the impact of thermal management may be difficult, MDAO is a rigorous way to consider these tradeoffs.

Chapter 6 extended the set of thermal models to include purpose-built semi-empirical motor and battery heat sink models, fluid pumps and hoses, and chillers. To validate that the thermal models are relevant and accurate, I simulated several test flights of the Pipistrel Velis Electro and compared the result to the flight test data. OpenConcept thermal models were useful for parameter identification and were predictive for thermal states, indicating good agreement between model and experiment.

Finally, in Chapter 7, I walked through an MDO trade space exploration of a parallel hybrid transport aircraft which was developed in support of a NASA demonstrator program. I outlined the process of redesigning the engine cycle to benefit from hybrid propulsion and described the surrogate modeling methodology. Next, I constructed a model of a relatively complex TMS architecture including a chiller, multiple coolant loops, and multiple ducted heat exchangers. I used dozens of MDO runs to compute the sensitivity of aircraft performance with respect to key parameters, finding that some were very important (such as range), while others had modest impact following optimization (such as battery temperature limits). This study also demonstrated that a chiller can be useful even for aircraft flying at high altitudes where the outside air is very cold. Unfortunately,

the study also found that the fuel burn benefit attributable to hybridization is very modest under this set of mission rules: less than 4%.

The second half of the dissertation developed a workflow for optimizing aircraft with bulky propulsion components in high fidelity. Chapter 8 reviewed the literature on packaging optimization and identified major shortcomings in the methods available to MDO practitioners working on industrially-relevant problems. I introduced a mathematical formulation for posing general packaging constraints based on triangulated computer geometry. Demonstration problems included three aerodynamic shape optimization problems of increasing complexity, culminating in the design of an aeroshell around a model of a seated human. These initial results confirmed that the constraint formulation worked as intended and may be promising for more complex aircraft problems.

Chapters 9 and 10 applied the new spatial constraint to two different aircraft design challenges. First, the wing and battery of an electric single-aisle airplane were simultaneously designed using MDO. The wing was shaped subject to aerodynamic drag, while the battery influenced aircraft range by making more energy available. The results validated the functionality of the constraint formulation for coupling the design of the OML and systems components via spatial integration. While the study did not consider structural weight, it indicated the value of using a battery geometry with more degrees of freedom to improve the packaging efficiency in the wing — a conformal battery had significantly higher range than the same aircraft with a bank of three batteries restricted to a prismatic shape.

The optimization study of Chapter 10 significantly increased the complexity of the MDO problem to the point of industrial relevance. An aerostructural MDO problem including aerodynamic drag, structural weight, and static aeroelastic effects was solved for an aircraft wing with integral compressed hydrogen tanks. This study included the design of multiple interior objects, high-fidelity multidisciplinary physics, and the use of a geometry engine that uses genuine CAD surfaces. The results show that storing compressed hydrogen tanks in the wing box *by itself* does not result in breakthrough emissions-free range, but can be a useful additional store of fuel to com-

pliment other locations in the aircraft. Significant wing root thickening increased available fuel volume and decreased structural weight, while also incurring a significant drag penalty.

11.2 Novel Contributions

1. *I developed an open-source framework for conceptual MDO of electric aircraft, OpenConcept.*

   In my review of electric aircraft modeling and simulation tools in Chapter 3, I identified duplication of effort as a major issue in academia. Ironically, to address this problem, I developed another new software tool for modeling, simulation, and optimization of aircraft with electric propulsion architectures — the distinction being that the software has been open-source from the beginning. Because it is built using OpenMDAO, OpenConcept is easily extensible to new and specific problems (e.g., incorporating an empirical map for a motor based on proprietary test data). It provides efficient and accurate gradients, which enables low-cost gradient-based optimization methods and rapid tradespace exploration. Finally, it is the only open-source tool in existence today which provides off-the-shelf heat sink models for electric aircraft components and enables users to optimize aircraft subject to unsteady thermal constraints. NASA selected OpenConcept as the tool of choice for tradespace exploration tasks in advance of a parallel hybrid demonstrator program.

2. *I conducted the first MDO studies of hybrid-electric aircraft subject to unsteady thermal constraints.*

   While an industry publication has examined the design of a TMS for a parallel hybrid [142], the study did not include unsteady thermal states and relied on point designs and engineering judgement to identify the critical design cases. A government study [131] addressed thermal trajectory optimization of an all-electric light aircraft. Prior to this dissertation, no study had

examined hybrid-electric aircraft subject to unsteady thermal constraints. In Chapters 5 and 7 I optimized the propulsion and thermal management of two different hybrid architectures and found that thermal mass effects are very important to the TMS sizing problem. The results showed for the first time that the relative cost of thermal management depends heavily on the mission parameters and the type of hybrid architecture.

3. *I developed novel semi-empirical motor and battery heat sink models reflective of current design practice.*

   I developed two new heat sink models based on open-source intelligence on the design practices of leading industry firms such as Siemens and Tesla Motors. These models scale properly as the components increase or decrease in size, reducing the size of the MDO problem (as certain heat sink design variables are eliminated).

4. *I developed an improved mathematical model of a chiller suitable for design optimization.*

   The equations for conceptual-level modeling of refrigeration systems are well-established and simple. They are suitable for hand calculations, but for gradient-based optimization or applications using numerical solvers, the mathematical properties produce significant headaches. In Chapter 6 I developed adjustments to the theoretical Carnot cycle which eliminate the numerical challenges. Inspired by the architecture of the Tesla Model 3, I also developed a mathematical model for a bypassable chiller. These improvements enabled me to optimize the design of a TMS including a chiller for the single-aisle hybrid in Chapter 7.

5. *I validated thermal management models using electric aircraft flight test data.*

   Because the flight test history of electric aircraft is very limited, and the majority of the extant data is in the hands of private companies, it has historically been difficult to validate electric aircraft models. Pipistrel Aircraft generously provided three flight test datasets and allowed me to validate the thermal models introduced in Chapters 5 and 6. Blind validation

tests indicated good agreement between model and experiment for the motor heat sink, duct, and heat exchanger models. To my knowledge, this is the first time that electric aircraft TMS models have been empirically validated.

6. *I developed a novel spatial integration constraint formulation.*

   In Chapters 8–10 I introduced a new mathematical formulation of spatial feasibility constraints using constraint aggregation. Prior to the work in this dissertation, extant geometry constraints for high-fidelity MDO were not general enough to solve certain problems of industrial relevance (those with packaging requirements on complex geometries, or where the object to be packaged can move relative to the OML). I validated the usefulness of this constraint by solving test problems of increasing complexity, up to and including high-fidelity aerostructural wing optimization with design and packaging of many objects.

7. *I developed a high-performance, open-source implementation of the spatial integration constraint.*

   In Appendix B I detail the implementation of the above-mentioned packaging constraint. With careful attention to the parallelization strategy, very high performance on present-day HPC resources is demonstrated. This performance was easily sufficient to solve the complex hydrogen wing test case in Chapter 10. After the defense of this dissertation, the code will be open-source under the name `geograd`. At least one aerospace prime contractor has already expressed interest in using the software and constraint in their own optimization problems.

8. *I developed a geometry engine for high-fidelity MDO based on CAD geometry.*

   While the MACH framework is highly capable for aerostructural MDO problems, users have traditionally relied on CAD-free geometry parameterization. As outlined in Appendix C, I developed an open-source wrapper for the ESP software which links CAD geometry with the rest of the framework and provides gradients using parallel finite differencing. I used this

capability to parameterize the tank geometry for the hydrogen aircraft wing design problem in Chapter 10. The CAD-based engine is already being used for shape optimization studies by other students in the MDOlab.

9. *I demonstrated MDO packaging optimization with high-fidelity physics on a problem of industrial relevance.*

   The Chapter 10 MDO study involved the design of a wing subject to RANS aerodynamics, finite element structural design, and coupled static aeroelastic effects. The packaging problem involved many independently-designed hydrogen fuel tanks which needed to be integrated not only with the OML, but also the various internal structural elements such as spars and ribs. I demonstrated that this complex MDO problem is not only feasible using today's computational resources, but feasible within one shift using relatively modest HPC resources. This raises the prospect of using MDO to handle spatial integration trade studies in an industrial setting. These types of trade studies are particularly frequent and laborious today, and using optimization could free up engineers to focus on higher value tasks.

10. *I contributed a highly-cited review article on electric aircraft propulsion.*

    Portions of Chapters 2, 3, and 5 were previously published as my first-author contributions to [17], a comprehensive review article covering electric aircraft design from an aerospace engineering perspective. The paper has already been cited 143 times as of the writing of this dissertation, indicating the significant appetite for such an article in the field.

11.3 Recommendations for Future Research

1. *Increase fidelity of electric machine and battery models*

   Because the scope of most of the work in this dissertation was very forward looking (2030+), I did not have a particular design of electric machine in mind in order to use higher-fidelity

physics-based models, nor did I have empirical data on the performance of these components at various operating conditions. The result is that I used very simple models for the electric machines and batteries. However, these data are relatively easy to generate or measure for components that exist today. An interesting extension to this research would be to incorporate physics-based or empirical models for motor, generator, and battery components. This would enable system voltage trade studies (which I did not approach in this dissertation).

2. *Develop new methods to make Newton solvers more robust*

   A significant portion of the time involved in this research was troubleshooting OpenMDAO cases where the Newton solver failed to converge (generally because of numerical singularities in the underlying models). While Newton solvers are very powerful, the need to provide good initial guesses and the associated burden imposed by this requirement has been previously identified [158]. Developing methods to improve the convergence characteristics of Newton solvers could be a fruitful area of research in the applied math domain. If the solver was more robust, I might have considered running cases with pyCycle in the main optimization loop instead of handled offline as a surrogate model.

3. *Improve OpenConcept integration with true trajectory optimization capability*

   While a relatively crude form of trajectory optimization was used for the thermal control variables in the Chapter 7 trade study, using a "true" trajectory optimization transcription would be a major improvement. I resisted using such a transcription for the broad tradespace studies because, like Newton solvers, trajectory optimization transcriptions require good initial state guesses. However, the Dymos [281] trajectory optimization tool, which is also built in OpenMDAO, has made significant improvements in robustness recently. At one point I prototyped an initial integration between Dymos and OpenConcept but have not had the bandwidth to fully develop it into a publicly releasable form. This would involve replacing the ODE states currently integrated using Simpson's rule with Dymos states. The challenge

is retaining ease of use, since trajectory optimization tools have a steep learning curve.

4. *Incorporate CFD flow path optimization for heat exchanger ducts*

   This work used empirical loss factors for pressure losses in the heat exchanger duct nozzle and diffuser. An interesting avenue of research would be to use CFD to optimize the geometry of the flow path. The flow path, heat exchanger, and overall TMS could be optimized together in a higher level of fidelity. Additional realism could be added by considering the structural weight changes introduced by changing the duct geometry. The DAFoam solver has already been demonstrated on interior flow problems and could be a good candidate for this type of study.

APPENDIX A

Surrogate Modeling of Engine Data

A.1 Introduction

Because iterative optimization algorithms (such as SLSQP or SNOPT) make many function calls, it is important to ensure that the computational cost of the models is reasonable for the task. It is also important for these algorithms that the objective function and constraints are evaluated without raising errors over a broad range of inputs. Unfortunately, computer codes used to evaluate aerodynamics and propulsion require high computational cost and sometimes fail to converge for a given set of inputs.

A strategy to mitigate cost and robustness problems with a simulation code is *surrogate modeling*, where simulations are run to generate data offline, and then a model fit to the data is used within the optimization loop. However, many engineering simulations are highly nonlinear and are poorly approximated by lower fidelity methods such as linear or polynomial regression. If surrogate modeling approximates the simulation too poorly, then the optimization process will not produce useful or realistic results. Therefore, improved surrogate modeling techniques are desirable for advanced computer simulations.

The parallel hybrid electric trade study of Chapter 7 includes the NASA N+3 concept turbofan engine [219]. A computationally-expensive model of the turbofan was developed using the simulation code pyCycle [218]. However, in order to make the computational cost of the optimization

problem feasible, an accurate surrogate model must be used instead of the full model when in the optimization loop.

The purpose of this Appendix is to document the process of selecting a surrogate modeling approach for the multidimensional turbofan engine data, and to document the accuracy of the surrogate model. This material was originally produced as a group term project in Professor Clayton Scott's EECS 545 (machine learning) class — the other group members were Rishi Nath Senthil and A.A. A.A. ran the radial basis function test case and helped write the text of the report. Rishi Nath Senthil helped with general editing of the report. I provided the data set, ran the test cases except for the radial basis function test case, and was the primary author of the report text.

A.2 Problem Statement

A.2.1 Dataset Description and Feature Analysis

The turbofan engine data consists of analyses run at various operating conditions using the pyCycle simulation tool. Each data point consists of the operating conditions $\mathbf{X}$, a four-dimensional feature vector, along with the output $\mathbf{Y}$, a four-dimensional vector. Each operating condition $\mathbf{X}$ consists of a throttle setting, altitude, Mach number, and hybrid power input. Feature analysis revealed that the problem is very linear in the hybrid power axis. Therefore, only throttle setting, altitude, and Mach number need to be considered in each surrogate; subsequently, a linear interpolation between different hybrid power levels is sufficient. The full pyCycle model output $\mathbf{Y} = f(\mathbf{X})$ consists of fuel flow ($\dot{m}$), thrust force (F), turbine inlet temperature (T_4), and stall margin. The dataset consists of approximately 1800 simulated engine runs at a full-factorial grid of Mach number, altitude, throttle, and hybrid power.

This data set is challenging to model because of its nonlinearity. Generally, thrust and fuel flow decrease with Mach number and altitude, but these relationships are highly nonlinear. Thrust

and fuel flow are more linear with throttle and hybrid power. The engine temperature output is highly nonlinear and has discontinuities where certain control limits are reached (for example, at full throttle at sea level). The stall margins are somewhere in between.

A.2.2 Surrogate Modeling

The surrogate modeling problem is as follows: we have multivariate input $\mathbf{X}$ and multivariate outputs $\mathbf{Y}$ and need to compute a surrogate model $g(\mathbf{X})$ which closely approximates $\mathbf{Y} = f(\mathbf{X})$ with low computational cost to evaluate.

We can measure the closeness of the approximation by measuring the mean error of the jth output of $\mathbf{Y}$:

$$\text{Mean absolute error} = \frac{\sum_{i=1}^{n} |g_j(\mathbf{X}_i) - \mathbf{Y}_{ij}|}{n} \tag{A.1}$$

$$\text{Mean relative error} = \frac{\sum_{i=1}^{n} |g_j(\mathbf{X}_i) - \mathbf{Y}_{ij}|}{\sum_{i=1}^{n} \mathbf{Y}_{ij}} \tag{A.2}$$

where n is the total number of training points and $g_j(\mathbf{X}_i)$ is the jth output of the surrogate model evaluated at training point $\mathbf{X}_i$.

A.3 Related Work

Surrogate modeling has its roots in theory of physical experiment design. Efficient space-filling experiments were combined with second-order polynomial regression to produce the *response surface methodology* [282]. This approach was later extended to computer simulations [283]. Bouhlel et al. developed the Python package Surrogate Modeling Toolkit (SMT) [284]. The specific surrogate modeling methods in SMT are reviewed separately in the methods section (for brevity).

As this engine data set is brand new, no previous published work has applied machine learning techniques to the data set. However, the SMT package has been used for several design optimization tasks on different problems. The KPLS method was successfully used as a surrogate model for wind turbine blades and car design [285], while RMTS was demonstrated with a helicopter rotor [284]. For these test problems, it does not appear that a systematic effort to evaluate alternative machine learning (ML) approaches was made. A different engine model (for a military engine) with fewer inputs was fit using a kriging model and used for a trajectory optimization problem [286]. Since the accuracy of the model appeared qualitatively suitable for the problem, the authors did not attempt to rigorously quantify model fit error in this case either, or seriously examine alternative, possibly superior ML approaches. This work is the first time different ML surrogates are rigorously compared for analyzing *any* engine data, not just this specific data set.

A.4 Methods

A.4.1 Preprocessing

Since the training and test data is not probabilistic, it required less cleaning and preprocessing than expected. We scaled all of the inputs to $\mathcal{O}(1)$ which involved scaling altitude by 0.001. This ensured that each feature direction was equally important when applying kernel-based methods.

A.4.2 Polynomial Regression

Polynomial regression fits a polynomial surface to the training data using a least-squares minimization procedure to vary the coefficients. Multivariate linear least squares fits a hyperplane through the dataset to predict a target variable y from multiple independent feature variables. It has the form:

$$y_i = \alpha + \beta_1 x_{1i} + \beta_2 x_{2i} + ... + \beta_n x_{ni} \tag{A.3}$$

where y_i is the variable to be predicted and $x_{1i} - x_{ni}$ are the i^{th} data point of the n-independent feature variables.

Quadratic regression fits a parabolic surface to the dataset using a least-squares minimization procedure. In one dimension it has the form:

$$y_i = \alpha x_i^2 + \beta x_i + \gamma, \alpha \neq 0 \tag{A.4}$$

where y_i is the variable to be predicted, x_i is the independent feature variable, α, β, γ are coefficients determined through the fitting procedure. If the input features are multi-dimensional, multivariate polynomial regression methods are employed and the expression turns into a matrix equation.

A.4.3 Ordinary Kriging

Kriging is a term for a particular class of gaussian process regression models. In ordinary kriging, the surrogate consists of a trend function(s) $f_i(\mathbf{X})$ (generally just a constant) and a realization of a random field $Z(\mathbf{X})$ as follows [282, 285]:

$$\hat{y}_i = \sum_{i=1}^{k} \beta_i f_i(\mathbf{X}) + Z(\mathbf{X}) \tag{A.5}$$

where the random field $Z(\mathbf{X})$ has the following covariance function:

$$\left[Z\left(\mathbf{X}^{(i)}\right), Z\left(\mathbf{X}^{(j)}\right)\right] = \sigma^2 R\left(\mathbf{X}^{(i)}, \mathbf{X}^{(j)}\right) \tag{A.6}$$

and R is a covariance kernel, such as the Gaussian kernel (covariance varies with squared distance) or the exponential kernel (covariance varies with absolute distance). $Z(\mathbf{X})$ is obtained by solving a constrained optimization problem involving the hyperparameters θ_i of the covariance kernel during the training procedure. Variance is minimized subject to a zero bias constraint.

Kriging thus produces the best unbiased estimate given a particular covariance kernel. Subsequent inference points can be computed inexpensively.

A.4.3.1 Kriging Partial Least Squares (KPLS)

Because ordinary kriging requires solving a constrained optimization problem that scales with the number of input variables, it can be computationally expensive to train on data sets with high-dimensional $\mathbf{X}$. To combat this, KPLS [285] first performs dimensionality reduction in the input space ($\mathbf{X} \rightarrow \hat{\mathbf{X}}$, $\mathbb{R}^n \rightarrow \mathbb{R}^m$, $n > m$) using principal components analysis (PCA). An ordinary kriging model is then trained on the reduced input space. This reduces the dimensionality of the kriging training problem while minimizing the loss of information from the higher-dimensional input space.

A.4.3.2 Sequential KPLS (KPLSK)

KPLSK is an incremental improvement over ordinary kriging. Sometimes, the ordinary kriging optimization problem will fail to converge due to a poor initial guess. In KPLSK, hyperparameters θ_i^* are first estimated using the KPLS training procedure in a reduced-dimensional space. Subsequently, an initial guess of the ordinary kriging hyperparameters θ_i are obtained by projecting θ^* back into the full dimensional space using the PCA transform. Finally, the ordinary kriging training optimization problem is solved using the initial guess obtained in the previous step [285]. This marginally enhances the quality of ordinary kriging model fit and significantly improves robustness.

A.4.4 Radial Basis Function Interpolation

Radial basis function interpolation interpolates the data as a linear combination of basis functions, as follows:

$$y = \mathbf{p}(\mathbf{x})w_p + \Sigma_i^{nt}\phi(\mathbf{x}, \mathbf{xt_i})\mathbf{w_r}$$

where $\mathbf{x}$ is the prediction input, $\mathbf{p}(\mathbf{x})$ is an optional mapping into a polynomial fit, $\mathbf{xt_i}$ is the *i*th training data, y is one element of the prediction output, ϕ is the Gaussian kernel, and w_p and $\mathbf{w_r}$ are parameters obtained by solving a linear system during the training step. Specifically, the coefficients w_p and $\mathbf{w_r}$ are computed by solving the following linear system:

$$\begin{bmatrix} \phi(\mathbf{xt}_1, \mathbf{xt}_1) & \dots & \phi(\mathbf{xt}_1, \mathbf{xt}_{nt}) & p(\mathbf{xt}_1)^T \\ \vdots & \ddots & \vdots & \vdots \\ \phi(\mathbf{xt}_{nt}, \mathbf{xt}_1) & \dots & \phi(\mathbf{xt}_{nt}, \mathbf{xt}_{nt}) & p(\mathbf{xt}_{nt})^T \\ p(\mathbf{xt}_1) & \dots & p(\mathbf{xt}_{nt}) & 0 \end{bmatrix} \begin{bmatrix} \mathbf{w_r}_1 \\ \vdots \\ \mathbf{w_r}_{nt} \\ w_p \end{bmatrix} = \begin{bmatrix} yt_1 \\ \vdots \\ yt_{nt} \\ 0 \end{bmatrix}$$

It is worth noting that the offsetting polynomial is not required in this model. As such, the RBF method can be defined as a weighted sum of radial basis functions. When using RBF there are many different types of basis functions that can be used, such as the Gaussian:

$$\phi(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(\frac{||\mathbf{x}_i - \mathbf{x}_j||_2^2}{d0^2}\right)$$

A.4.5 Regularized Minimum-Energy Tensor-Product Splines (RMTS)

Another method of analysis used on the data is called Regularized Minimum-Energy Tensor-Product Splines (RMTS). A spline is a function defined by piece-wise polynomials. The prediction function for RMTS is

$$y = \mathbf{F}(\mathbf{x})\mathbf{w}$$

where $\mathbf{w}$ is the vector of spline coefficients and $\mathbf{F}(\mathbf{x})$ is the vector mapping the spline coefficients to the prediction output. RMTS is a novel technique that is unique to SMT [287]. RMTS computes $\mathbf{w}$, the coefficients of the splines by solving an energy minimization problem. This is an

unconstrained optimization problem of the form:

$$\min_{\mathbf{w}} \quad \tfrac{1}{2}\mathbf{w}^T\mathbf{H}\mathbf{w} + \tfrac{1}{2}\beta\mathbf{w}^T\mathbf{w} + \tfrac{1}{2}\tfrac{1}{\alpha}\sum_i^{nt}\left[\mathbf{F}(\mathbf{xt}_i)\mathbf{w} - yt_i\right]^2$$

Here, $\mathbf{xt}_i$ is the input vector for the ith training point, $\mathbf{H}$ is the matrix containing the second derivatives of the splines, and α and β are regularization coefficients. While computing second derivatives and subsequently their integrals is expensive, the authors argue that with modern-day computational resources the computational cost is minimal compared to the performance this method is capable of.

A.4.6 Mixture of Experts (MoE)

Mixture of Experts (MoE) is a method to replace a single model with a weighted sum of local models, or "experts" [288]. Conceptually, it relies on the idea that the input space is heterogeneous, and that models trained on a subset of the input space may perform better than a surrogate trained on the whole input space. MoE occurs in three steps:

1. Clustering: the inputs are clustered together with their output values by means of parameter estimation of the joint distribution.
2. Training: Train local experts, one per cluster.
3. Recombination: Create final model combining all the local experts using the Gaussian Mixture Model (GMM).

In MoE, the recombination step is traditionally done with the Expectation-Maximization method. When using MoE, it is important to consider the number of clusters as well as the types of experts (eg. linear, quadratic polynomials etc.). While MoE is an effective tool in certain applications, it is important to note that it is not a one-size-fits-all-approach. In fact, MoE can actually under-perform global models in some cases [289], such as a dataset drawn from a single distribution.

Mean abs error	**RTMS**	**KPLS**	**KRG**	**KPLSK**	**RBF**	**QP**	**LS**	**MOE2**
Thrust (lb)	27.477	1.291	1.298	0.00591	3.042	403.213	1163.657	64.94
Fuel Burn (lb/s)	7.03e-3	4.25e-4	4.23e-4	9.12e-4	8.76e-4	3.081e-2	8.798e-2	6.421e-3
T_4 **(Rankine)**	6.423	1.280	1.275	2.054	2.288	34.773	48.500	4.698
Mean rel error								
Thrust	0.44%	0.02%	0.02%	≈0.0%	0.05%	6.44%	18.60%	1.04%
Fuel Burn	1.09%	0.07%	0.07%	0.14%	0.13%	4.77%	13.63%	0.99%
T_4 **(Rankine)**	0.26%	0.05%	0.05%	0.08%	0.09%	1.41%	1.96%	0.19%

Table A.1: Crossvalidation error for the surrogate modeling methods

A.5 Evaluation

A.5.1 Evaluation of Surrogate Modeling Methods

We performed a side-by-side comparison of 8 surrogate modeling machine learning methods on the turbine engine data set. Each method was implemented in the SMT Python package. Using the SMT package allowed us to compare more methods without spending time implementing each one individually, and the common API made swapping out various surrogate modeling ML methods straightforward. In order to evaluate the accuracy of the model, we used a k-fold cross validation procedure with $k = 10$. This balances the computational cost of fitting the models with the validity of the cross validation procedure. We randomly jumbled the data, then divided the data into ten folds. For each fold, we trained the model on the other 90% of the data, and computed model fit error against the fold data. We used the mean absolute error and mean relative error metrics in Equations A.1 and A.2. We averaged the error metrics across the ten folds and tabulated the result in Table A.1.

We found that the kriging based models (ordinary, KPLS, and KPLSK) and radial basis functions, produced consistently excellent performance across the entire design space and for all output variables, with average error less than 0.1% of the mean value for ordinary kriging and KPLS. However, we had to provide default guesses for the hyperparameters which were different than the defaults. The optimization algorithm used to fit the kriging models would sometimes fail to

converge when the default initial hyperparameter guesses were used. We found that the for the RBF, using the default parameters, without a polynomial fit improved the results. In fact, using a regularization parameter of $\lambda = \frac{1}{n}$, led to a tenfold increase in mean absolute error.

The RMTS method showed acceptable performance, with average errors on the order of 1% (at least ten times higher than kriging). The quadratic polynomial and least squares linear models performed poorly on thrust data, with average errors of 6.44% and 18.6% respectively. This is consistent with the highly nonlinear dataset, and represents a level of performance is not acceptable for use in multidisciplinary design optimization.

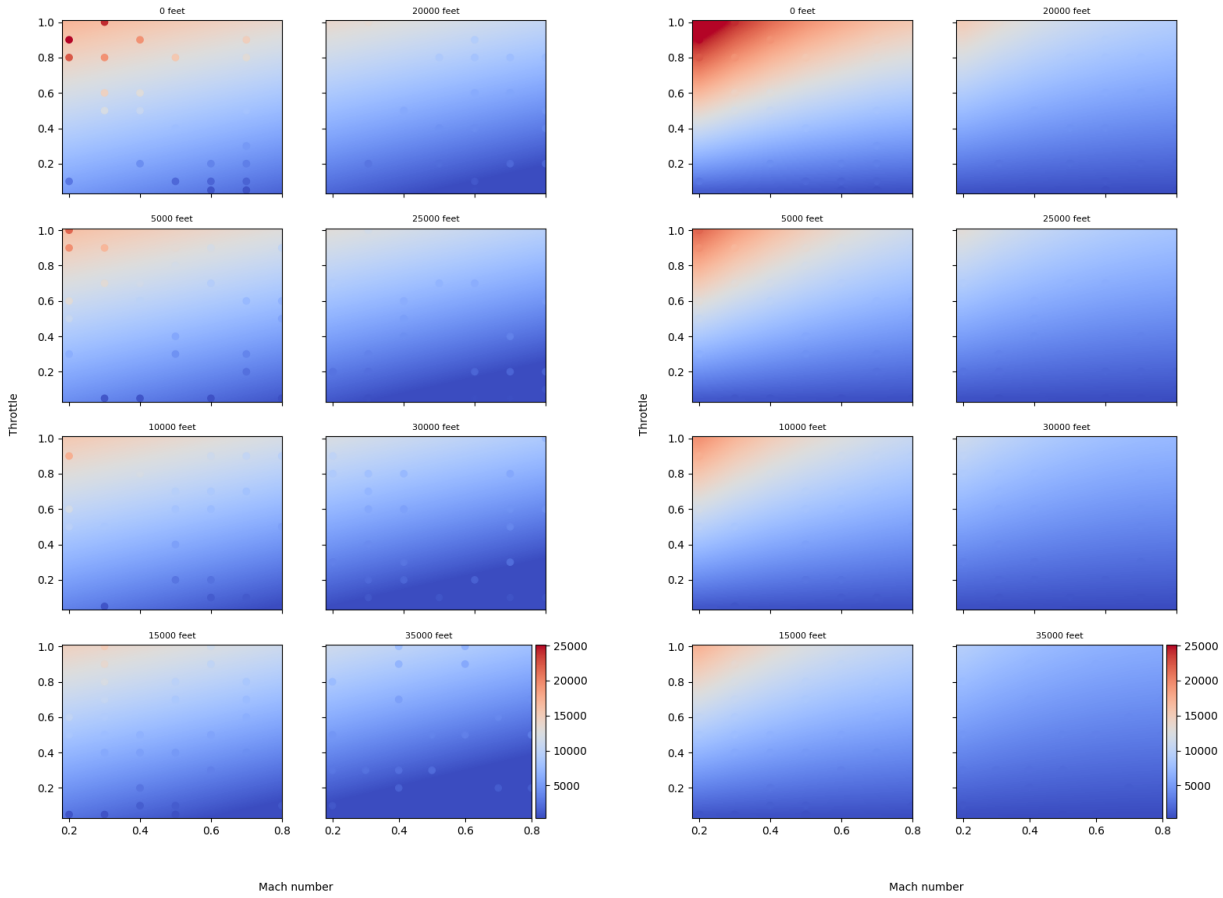
The mixture of experts model with two clusters had worse performance than kriging, *even though kriging-based models were included in the candidate experts*. This can be explained because the training data is drawn from the same distribution, so GMM will not produce a useful clustering. In this situation, all MoE does is reduce the amount of training data available to each model and degrades performance. When MoE is trained with one cluster, the result is identical to the best kriging or RBF model for each output.

We also qualitatively evaluated the performance of each model in different parts of the flight envelope. Since the kriging models are interpolating, they may fit more poorly near the data set edges. We created two-dimensional slices through all outputs generated by all the surrogates. We plotted the surrogate estimates as a surface plot, and a 30% holdout test dataset as colored dots (indicating the true value). A representative example for thrust data is Figure A.1 and Figure A.2 shows the same for fuel burn data. We see that the held-out thrust data is not accurately modeled by the linear surrogate (left) at high and low thrust levels. However, the KPLS surrogate is nearly indistinguishable from the holdout data, even near the edges of the model. The fit is so accurate that there is essentially no difference between the dot color and the background color, rendering the test data points virtually invisible for the KPLS plot.

A.6 Concluding Remarks

As expected, the results show that linear and quadratic models were unable to capture the nonlinear trends in the engine deck. Kriging and radial basis functions were very accurate and qualitatively fit well throughout the flight envelope. There was not much difference between ordinary kriging and the KPLS variants. While arguably the most complex model, MoE performed worse than kriging and RBF, because the engine data is drawn from the same distribution or cluster. By dividing data from the same distribution into different clusters, MoE's expert models lose information that they would otherwise have with a cluster of size one.

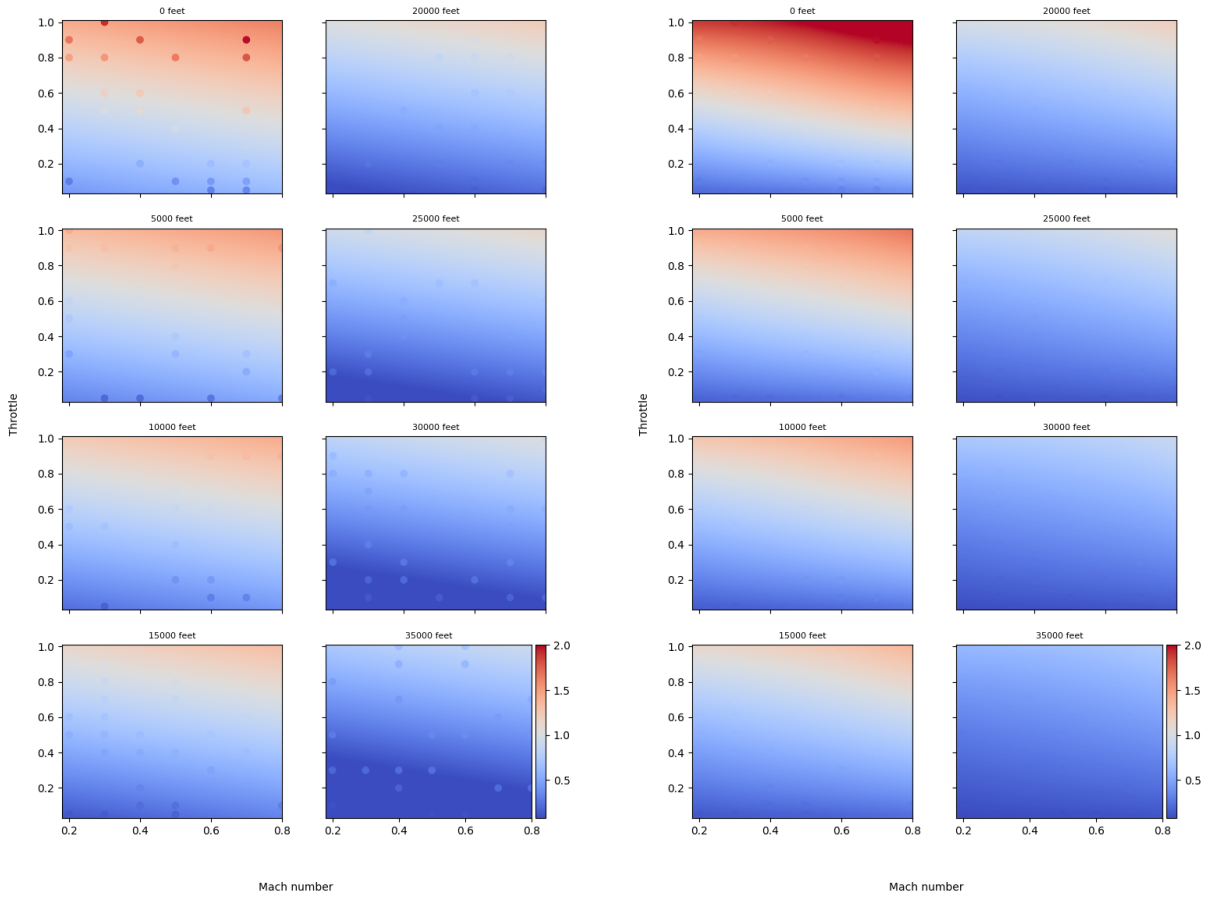
I elected to use ordinary kriging for the Chapter 7 because of OpenMDAO's pre-existing kriging surrogate implementation. Using the KPLS variants afforded almost no improvement in performance, yet would have required significant software development effort to integrate into OpenMDAO/OpenConcept.



(a) Least Squares

(b) KRG

Figure A.1: N+3 hybrid thrust (lb) as a function of altitude, Mach number, and throttle. Kriging far outperforms least squares.



(a) Least Squares

(b) KRG

Figure A.2: N+3 hybrid fuel burn (lb/s) as a function of altitude, Mach number, and throttle. Kriging far outperforms least squares.

APPENDIX B

Geograd Implementation and Performance Benchmarking

B.1 Introduction

In Chapters 8 and 9, the geometric constraint package known as `geograd` was implemented using the TensorFlow framework. TensorFlow is a Python library designed for machine learning. Using TensorFlow, I was able to compute the packaging constraints in a pure single-instruction multiple-data (SIMD) fashion. It is very fast when running on the GPU on my local machine, but I do not have access to GPU resources on the compute clusters I typically use. The purpose of Appendix is to detail the implementation and performance of a significantly improved Fortran 90 implementation of `geograd` which takes advantage of algorithm opportunities that cannot be expressed TensorFlow's SIMD language.

The first section will describe the computation in more detail and the prior state of the software. Second, I describe a test case I constructed for timings and benchmarking. Following that, I describe the development and verification of the baseline Fortran code. Next, I walk through each improvement to the algorithm and describe the dynamic load balancing problem that emerged during this step. Finally, I describe the dynamic load balancing approach and present the final timings.

I was ultimately able to achieve over a **1350x** speedup on my benchmark case compared to my original TensorFlow-based pure SIMD implementation. This Appendix was originally a solo term project and report for Professor Quentin Stout's EECS 587 parallel computing class and has been lightly edited for inclusion in the dissertation.

B.2 Description of the Computation and Previous Work

As originally described in Chapter 8, I identified a mathematical formulation for a geometric constraint that is deterministic, differentiable, and C^0 continuous. I restate the basics here for clarity. We begin with triangulated representations of aircraft outer surface r and an inner object s to pack inside it, as pictured in Figure B.1. r and s are each represented as lists of vertices **A**, **B**, **C** of dimension 3 by m or n (the size of each mesh).

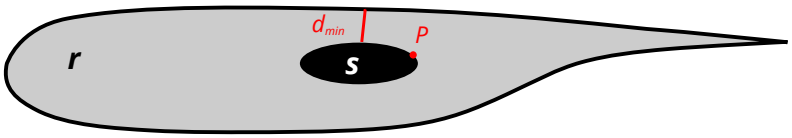


Figure B.1: A section view of an object s inside wing r and the minimum distance d_{min} between them

When one object encloses another, the minimum distance d_{min} between them is greater than zero by some margin. Therefore, a first approach to a geometric constraint might involve computing $d_{\text{min,rs}} \geq 0 + \text{tol}$. We can compute d_{min} between two triangulated surfaces by computing the minimum distance between each individual pair of triangles. The minimum distance between a pair of triangles can be found by a total of fifteen primitive tests between the vertices: six point-triangle tests, and nine line-line tests [247].

$$d_{\text{min,rs}} = \min(d_{\text{min},ij}) \text{ for each triangle } i,\ j \text{ in surfaces } r,\ s \tag{B.1}$$

However, as explained in Chapter 8, the optimizer behaves poorly when the constraint is posed

in this way because information about the *nearly* closest points is effectively discarded across the `min` operation. Instead, I use a constraint aggregation approach using the KS function. The computation is as follows:

$$KS_{\text{geom}}(\mathbf{x}) = \frac{1}{\rho}\ln\left[\sum_{i,j=1}^{m,n} e^{\rho\left(d_{\min}(\mathbf{x}) - d_{ij}(\mathbf{x})\right)}\right] - d_{\min}(\mathbf{x}) \leq 0, \tag{B.2}$$

where $\mathbf{x}$ is a vector of design variables, $d_{ij}(\mathbf{x})$ is the pairwise distance between the ith facet of r and jth facet of s, $d_{min}(\mathbf{x})$ is the minimum distance between r and s at the current design point, and ρ is a user-controllable constant. Strictly speaking, d_{ij} is a function of the surface mesh points of r and s, not the design variables $\mathbf{x}$ directly.

Initially, I implemented the computation of Equation B.2 using the TensorFlow framework [250]. TensorFlow uses a graph representation of a series of array operations and computes a result in a SIMD fashion. Since many machine learning workflows require gradients for training purposes, TensorFlow natively supports analytic reverse-mode derivatives, making it ideal for use in this gradient-based optimization application. TensorFlow also supports heterogeneous computer architectures, including GPU acceleration. I found that I could run the calculation almost instantaneously on the GPU of my desktop machine.

I used my TensorFlow implementation to generate the results in Chapter 8 [261] and Chapter 9 [268]. For the Chapter 9 results, I used the Texas Advanced Computing Center (TACC) Stampede2 cluster. Unfortunately, while TACC does have GPU nodes, the geometry calculation is a very small portion of the overall compute cost (which includes expensive, CPU-based fluid dynamics calculations). Therefore, it did not make economic sense to reserve GPU nodes just for the geometry constraints since it would be idle a large percentage of the time. Instead, I used TensorFlow's CPU support and created an optimized, CPU-parallel implementation. I used the `mpi4py` package to scatter the computation over all MPI processes, run one TensorFlow call on each process, and gather the results. I built TensorFlow from source with Intel MKL support and

architecture-specific optimizations, including the Intel AVX-512 SIMD extensions. This configuration provided enough performance for my immediate research needs, but increased the wall time significantly compared to the GPU card on my desktop. I need to reduce the computation time in order to scale up to more complex optimization problems with more objects to pack inside the airframe, such as the problem in Chapter 10.

TensorFlow's SIMD approach makes it infeasible to use control flow and early exits to reduce the cost of the geometry calculations, which results in a large number of redundant calculations. I brainstormed several potential approaches which can be used to reduce compute cost, including:

- Arranging control flow so that the most likely branches are tested first
- Using the minimum instead of the sum of the 15 pairwise primitives for each triangle (reduces the gradient compute cost by 15x)
- Using bounding box tests to quickly exclude pairs of facets which are far away from each other and cannot possibly be the minimum distance

In order to use these approaches, I need more granular control flow than TensorFlow provides. Therefore, I needed to drop the TensorFlow SIMD formalism and write my own CPU-optimized implementation in Fortran with Python bindings. The latest version of `geograd` consists of this new Fortran 90 implementation.

B.3 Benchmarking Case Description

I wanted to isolate performance metrics to the geometry code, yet test the code in a setting representative of an optimization run, so I generated an artificial test problem. Consider a blended wing body (BWB) aircraft, as pictured in Figure B.2. The test geometry was obtained as a triangulated surface file using the NASA OpenVSP Hangar [290]. I created a notional football-shaped

geometry to represent a generic system component to be packed optimally into the wing and placed it into the right wing root, as pictured in Figure B.3. We can then compute Equation B.2 between the aircraft outer shape r and the football-shaped inner component s to see whether the current layout is feasible.

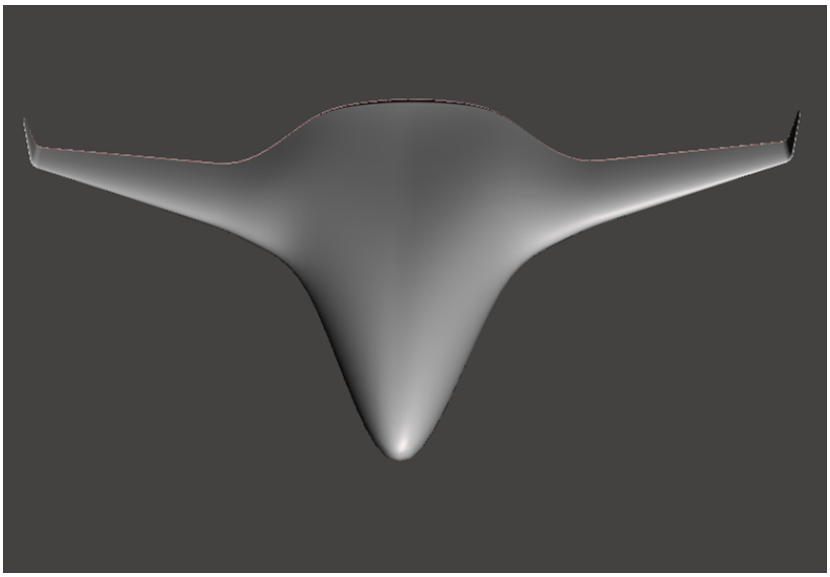


Figure B.2: Blended wing body geometry used for the test problem

During optimization, the relative positions of the two objects will change. Generally, the changes in position will be fairly subtle. However, to make the test problem as challenging as realistically possible, the component translates from one wing root to the other in 50 equally-spaced increments. The arrow in Figure B.3 shows the path of the component. The test comprises computing Equation B.2 50 times, one at each component position. I implemented the test as a Python unittest file.Two timings are obtained; one for analysis only (no gradient), and another with gradients.

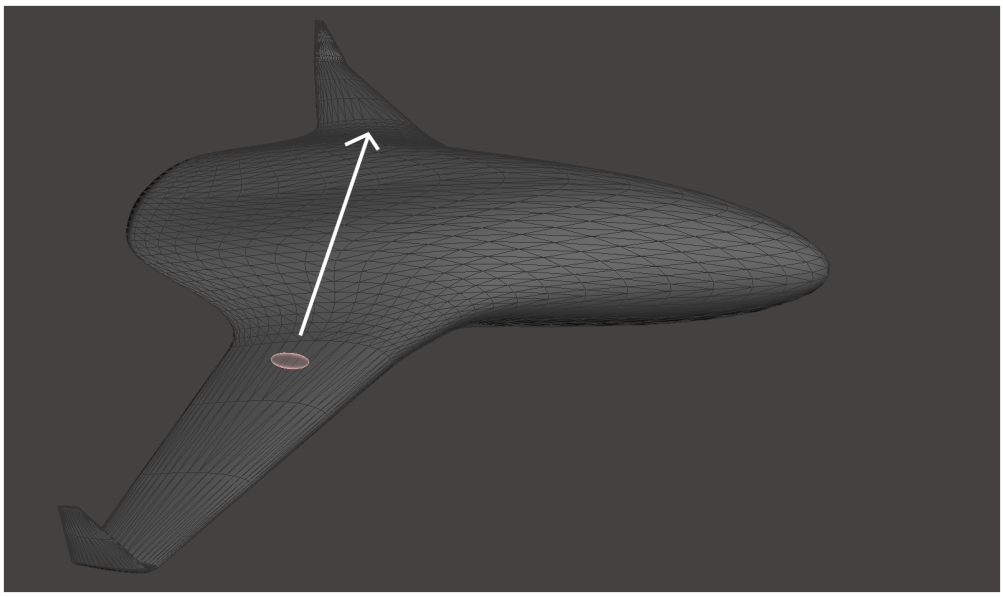


Figure B.3: The component geometry translates from one side of the aircraft to the other in the test problem

B.4 Baseline Code Development and Verification

I need to ensure the code is not only fast, but also accurate, so I developed the code in the following general steps:

- Writing line-line and point-triangle test primitive subroutines
- Algorithmic differentiation of the geometric primitives
- Writing the serial KS function subroutine (implementing Equation B.2)
- Creating the Python-Fortran interface
- Unit testing of the geometric primitives and KS function
- Parallelizing the KS function subroutine

B.4.1 Writing Geometric Primitives

I implemented the line-line and point-triangle tests as described in Ericson [247]. The subroutines, contained in `triangles.F90` take four three-dimensional points as input and return

the minimum distance. The sequence of control flow is designed to exit early in some common cases, deferring the least likely paths until as late as possible. I also developed unit tests using triangles in prescribed configurations to exercise each of the branches of the mathematics.

B.4.2 Algorithmic Differentiation of the Primitives

We not only need to know the minimum distance result of the primitives; we also need the derivatives with respect to the function inputs. It is possible to derive this by hand; however, it is much easier to use source code transformation via algorithmic differentiation (AD). I differentiated the point-triangle and line-line primitive subroutines using the INRIA Tapenade software. There are two modes in AD: forward and reverse (also known as "backprop" in the machine learning community). I used reverse mode, which requires only one function evaluation to get the derivatives of a single output (d_{min}) with respect to all 12 of the inputs (four points, three dimensions each).

B.4.3 Assembling the KS Function and Gradients

I implemented the loops to compute Equation B.2 with respect to all of the m facets of an outer surface and n facets of the inner surface. The subroutine `compute` takes 8 inputs:

- A_r, B_r, C_r: lists of vertices comprising the m triangles of the outer surface
- A_s, B_s, C_s: lists of vertices comprising the n triangles of the inner object surface
- $d_{\text{min,current}}$: the minimum distance between the two surfaces at the current design point (this is used to normalize the calculation which improves numerical conditioning)
- ρ: a user-defined constant which defines how tight the objects can fit next to each other

The subroutine returns:

- KS: the constraint described in Equation B.2
- d_{min}: minimum distance between the two surfaces

The major advantage that this Fortran implementation has over TensorFlow is data reuse. TensorFlow computes each array operation in a pairwise fashion sequentially. Every entry in each array is read one time per array operation, and one write is performed per array operation. This should result in a bad compute-to-memory access ratio. In the Fortran implementation, each pair of triangles is read from the global arrays only one time. Subsequently, all the primitive tests and summation is performed on local copies which should easily fit in the L1 cache. This should result in a major speedup over the TensorFlow implementation by itself.

I was careful to ensure sequential memory access by arranging the array order correctly. Fortran uses column-major order, so I used the spatial dimensions x, y, z as the first array dimension, and triangle count i, j as the second dimension. This results in coalesced memory access when individual vertices are read from the global arrays. However, for the purposes of visualization in this paper, I illustrate the transpose of the matrix, with each row comprising a triangle.

I also wrote a function `compute_derivs` which additionally computes the following derivatives with respect to the surface mesh points (all m by 3 arrays):

$$\frac{dKS}{d\mathbf{A}_r} \quad \frac{dKS}{d\mathbf{B}_r} \quad \frac{dKS}{d\mathbf{C}_r} \tag{B.3}$$

and the following derivatives with respect to the object mesh points (all n by 3 arrays):

$$\frac{dKS}{d\mathbf{A}_s} \quad \frac{dKS}{d\mathbf{B}_s} \quad \frac{dKS}{d\mathbf{C}_s} \tag{B.4}$$

These derivatives are required in order to compute the total derivative of the KS function with respect to the design variables. Recall that Equation B.2 depends on the surface mesh vertices A, B, and C of objects r and s. Applying the chain rule, we have:

$$\frac{dKS}{d\mathbf{x}} = \frac{dKS}{d\mathbf{A}_r(\mathbf{x})}\frac{d\mathbf{A}_r(\mathbf{x})}{d\mathbf{x}} + \frac{dKS}{d\mathbf{B}_r}\frac{d\mathbf{B}_r}{d\mathbf{x}} + \frac{dKS}{d\mathbf{C}_r}\frac{d\mathbf{C}_r}{d\mathbf{x}} + \frac{dKS}{d\mathbf{A}_s}\frac{d\mathbf{A}_s}{d\mathbf{x}} + \frac{dKS}{d\mathbf{B}_s}\frac{d\mathbf{B}_s}{d\mathbf{x}} + \frac{dKS}{d\mathbf{C}_s}\frac{d\mathbf{C}_s}{d\mathbf{x}} \quad \text{(B.5)}$$

A separate Python tool, pyGeo, provides the design variable sensitivities $d\mathbf{A,B,C}/d\mathbf{x}$.

Differentiating top-level code, particularly when it involves dynamically-allocated arrays or MPI calls, can be very inefficient. Therefore, I differentiated Equation B.2 by hand, obtaining:

$$\frac{dKS}{d\mathbf{A}_r} = -\frac{\sum_{i,j=1}^{m,n} \frac{d\, d_{ij}}{d\mathbf{A}_r} e^{\rho\left(d_{\min} - d_{ij}\right)}}{\sum_{i,j=1}^{m,n} e^{\rho\left(d_{\min} - d_{ij}\right)}} \quad \text{(B.6)}$$

and so on for the remaining 5 lists of input vertices. $d\, d_{ij}/d\mathbf{A}_r$ is computed using the AD version of the geometric primitives.

Note that, while $dKS/d\mathbf{A}_r$ is a dense m by 3 matrix, each pairwise update in the summation ($d\, d_{ij}/d\mathbf{A}_r$) is sparse and affects at most 3 entries at a time. Note also that the summation in the denominator is a global quantity, while the numerator is a local quantity specific to each facet pair i,j. Since the entries of the derivatives depend on both local information as well as the global summation across all the facet pairs, the overall computation is not embarrassingly parallel. However, the amount of communication required is fairly low (a single partial sum is Allgathered from all processes) in between two separate double ij loops.

B.4.4 Fortran-Python interface

The lab's optimization framework is Python-based. Therefore, all geometry information fed into the Fortran code needs to be passed in from a Python process executed under MPI. Normally, writing Python native interfaces is laborious, and in the past I have used the low-level Python-C API to do this.

A better way is to use the `f2py` interface generator. `f2py` was developed as part of the `numpy` project and it is distributed with `numpy`. It consists of a command-line interface which reads and compiles Fortran source and generates a `.so` shared object file which can be imported as a Python package. The user can control the structure of the input and output arguments by writing an `f2py` "signature" file, which I did in this case. `f2py` passes in numpy arrays by reference as native Fortran array pointers in the correct column-major memory layout.

B.4.5 Testing and Verification

A major benefit of the Python interface is that it is easy to create unit and integration tests. As I developed the codebase in a bottom-up approach, I ensured that unit tests were written and passing before moving on. First, I generated unit tests which cover every branch of the geometry primitives using simple configurations of triangles (`test_primitives.py`). Next, it is absolutely crucial to test the accuracy of AD derivatives, since Tapenade sometimes can be buggy and these types of errors can be difficult to detect in integration testing.

In order to verify derivatives, we use the *complex step* method (CS). The complex step method (Equation B.7) is conceptually similar to finite differencing, except that it is exactly accurate even when a very small step (on the order of machine ϵ) is used. This makes it suitable for testing the exact derivatives obtained via AD to high precision.

$$\frac{df(x)}{dx} = \frac{\text{Im}(f(x+ih))}{h} \tag{B.7}$$

The drawback of the complex step method is that it requires that the code being tested support complex inputs. Fortunately, the MDOlab has developed a method for compiling complex versions of our Fortran codes. We use a Python script to transform the base code into a complexified version. The script changes data types and overloads certain functions so that they produce the correct control flow even with complex inputs. I added this complex build to the Makefile

and wrote unit tests verifying the derivatives of both the geometric primitives as well as the top-level KS function derivatives. All the derivative outputs matched CS to a relative tolerance of 10^{-7}, indicating that both the AD derivatives and the by-hand derivative assembly were correctly implemented.

Finally, I ported my previous integration tests from `geograd`–TensorFlow over to this project and verified that the output values match the output values of my new code for known geometries. They did match to a high degree of accuracy.

B.4.6 Parallelizing the Baseline Code

As a first cut, I parallelized the new Fortran code in the same way that I had parallelized `geograd`. Here are the assumptions and requirements for the parallelization:

- Assume each process has a full copy of the input meshes r and s when it is called (no scatter from root required)
- Each process must end up with a full copy of both the KS value and all the derivative arrays
- The p processors will be provisioned such that $p << m, n$

We divide up the problem along the r mesh only. Each processor is given a range of rows in the r mesh arrays to compute against all of the s triangles. Figure B.4 shows the parallelized arrangement. First, an owner-computes approach is used to compute the summation of the KS function. Each process receives the summation value via an allreduce. If analysis is being performed without the need for derivatives, the computation is done at this point. If not, the derivatives are computed in an owner-computes fashion. Since the processors "share jurisdiction" over the s mesh, an MPI Allreduce summation is used to combine local copies of the s derivatives into a global array. The r mesh is simply concatenated and sent to all using the MPI Allgather subroutine.

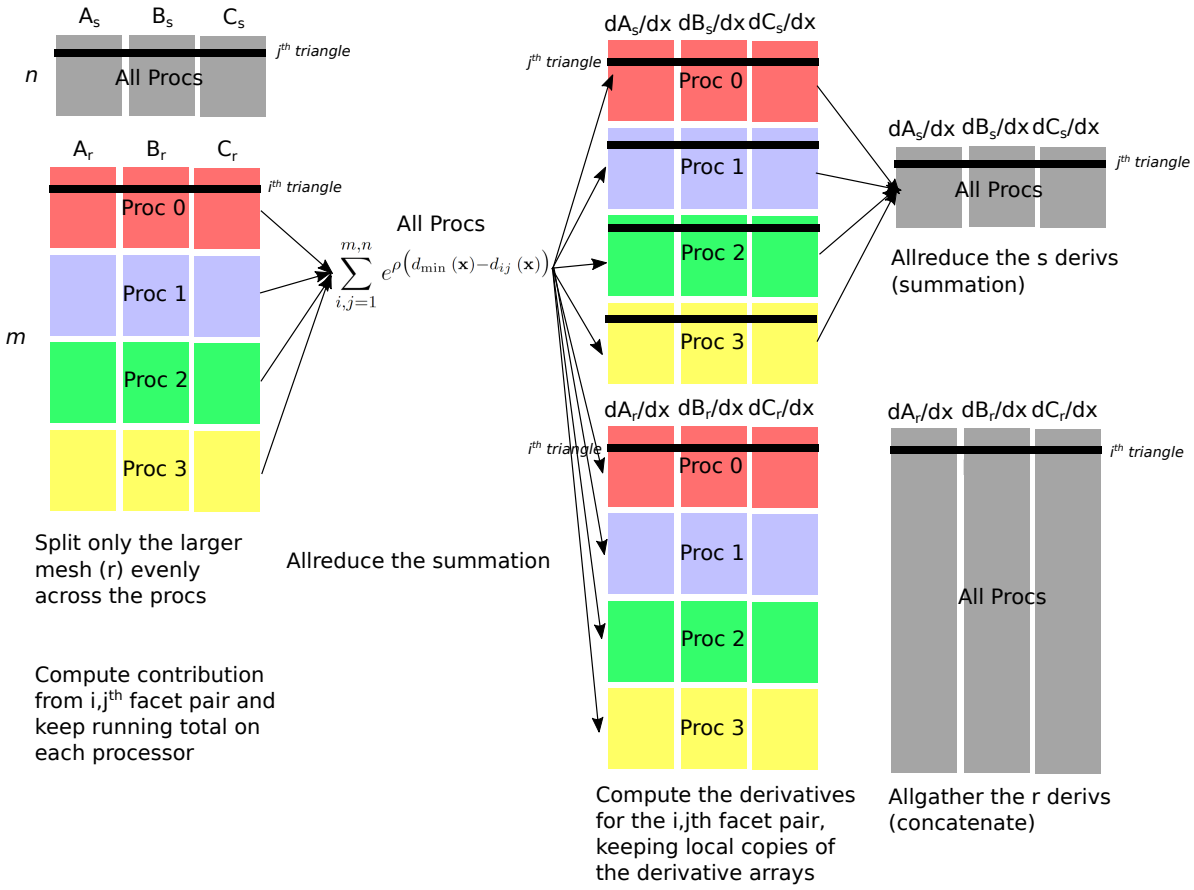


Figure B.4: The problem is parallelized by dividing up the larger array and distributing its computation evenly across the processors

B.4.7 Timing the Baseline

I ran the `geograd` (TensorFlow-based) implementation and my new Fortran implementation on the benchmark problem described in Section B.3 using 4 to 48 cores of a compute node on the TACC Stampede2 cluster. All code was compiled using the Intel Fortran compiler with maximum optimization, and Intel MPI. Unless otherwise specified in the report, I give timings and speedups for the 48-core case. The TensorFlow-based benchmark took 20s to run analysis only, and 38s for derivatives. My new Fortran-based benchmark took 0.371s for analysis only (**53x** speedup), and 1.93s for derivatives (**19x** speedup). These were extremely promising results, and I had not yet implemented any of the algorithm improvements.

B.5 Algorithm Improvements

Because TensorFlow does not support control flow (at least, not in a form which reduces computation for unused branches), there is great opportunity to gain speed by improving the algorithm. In this section I describe the improvements and report speedup relative to the previous iteration (not cumulative versus the original TensorFlow implementation).

B.5.1 Minimum Instead of Sum of the 15 Triangle Tests

Recall that each pair i, j of triangles requires 6 point-triangle and 9 line-line test primitives. In TensorFlow, there is no computational advantage to taking the minimum value of the 15 pairwise primitive tests, so I sum the KS results instead. However, in Fortran, I can eliminate 14 of the 15 derivative calls by taking the minimum instead of the sum over each triangle. I added a case control flow structure to only run the derivative subroutine for the most critical primitive. This had only a modest effect on the analysis-only benchmark (**1.1x** speedup), but a large effect on the derivatives benchmark (**3.93x** speedup). This optimization remains an exactly correct expression of Equation B.2.

B.5.2 Bounding Box Testing

If a pair of triangles is far away from each other, it is possible to prove cheaply that their interaction is not important. We can accomplish this through *bounding box* testing. Let us draw an axis-aligned bounding box around the inner object s as pictured in Figure B.5. Let us also define some tolerance value (say, the maximum dimension of s in any dimension). If the bounding box around triangle j is more than tol away from the bounding box of s, then we can guarantee that the minimum distance between them is at least tol. We can then replace a potentially large number of d_{ij} computations with $d_{ij} = tol$.

Computing bounding box tests is much cheaper than computing d_{ij} using the subroutines.

The bounding box test is simply several min/max operations on each triangle j, plus a few floating point comparisons. Since aircraft geometry predominantly varies in the wingspan direction, we can exclude a huge number of triangles with just one or two comparisons (by placing the x, or spanwise, axis tests first in the sequence).

I implemented the bounding box test using the heuristic tol setting of $max(x_{max}-x_{min},\ y_{max}-y_{min},\ z_{max}-z_{min})$ and re-ran the benchmark cases. The derivative benchmark gained an additional **3.4x** speedup compared to the previous optimization.

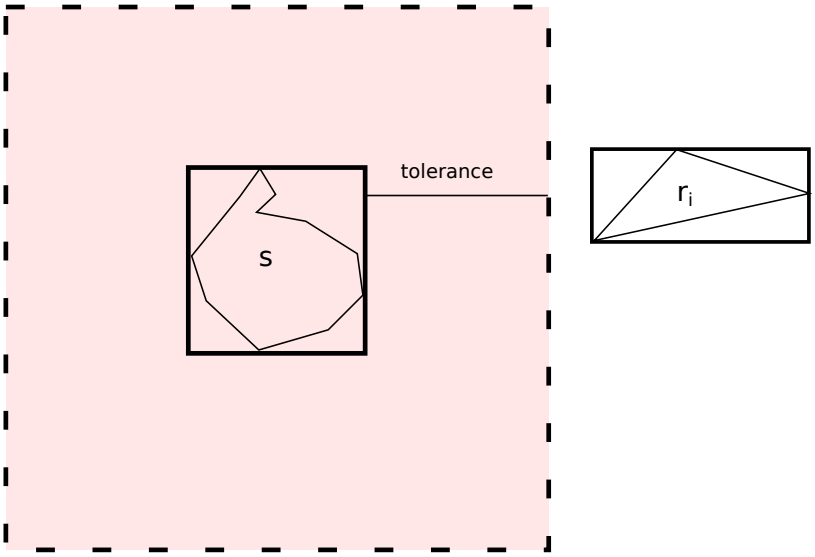


Figure B.5: Bounding box tests can quickly exclude obviously unimportant facets from needing expensive computation

B.6 Dynamic Load Balancing

While the baseline Fortran implementation and the "min of fifteen" optimization had good load balancing (utilization above 95% with 4 processors; above 65% with 48 processors), the bounding box test wrecked the load balancing. The reason for this is spatial correlation in the lists of mesh vertices. Consider the aircraft geometry pictured above. A mesh representation might include all of the "left wing" triangles in the first quarter of the list, or it may be completely

random (there are no guarantees). A bounding box test excludes triangles that are far from the present position of the object s. Therefore, if s is in the right wing, the processor which is assigned all of the left wing facets may have zero triangles to compute. Furthermore, as s and r deform and move during optimization, the load imbalance will change from processor to processor.

I measured this using the benchmarking case. After implementing the bounding box test, my processor utilization across 48 MPI processes decreased from the high 60% range to less than 15%. Examining the problem further, I found that one or two processors would have basically zero runtime, while the others were fully loaded. I verified that these low-runtime processes had most or all of their triangles excluded from expensive tests by the bounding box. Below, I will describe two approaches I used to solve the problem.

B.6.1 Global Load Balancing

Due to the dynamic nature of the load balancing issue, the load balancing step needs to be done on every function call. As a first cut, I decided to treat the bounding box tests as negligible in cost and simply smear all of the "active" triangles (the ones passing the bounding-box test and requiring more expensive tests) evenly across the processors sequentially. Briefly, the method is as follows:

- First, divide up all the triangles evenly across the processors
- Compute bounding box tests and place a boolean into a local vector indicating whether it is active or inactive
- Allgather the bounding box test results into a global vector on each processor
- Compute the sum of all active triangles in the global vector
- Compute a scan on the global vector to find the global indices i of the triangles in r across which the problem will be split for load balancing
- Divide the triangles up across the new split indices

The structure of the problem remains as pictured in Figure B.4, except that each processor will now have an unequal number of triangles to compute. I chose to ignore the potential issue of some processors having more inactive (cheap) triangles to compute than others. This keeps the array split indices limited in size and the split chunks contiguous in memory, which simplifies the collective operations required to get the derivatives assembled at the end.

This global load balancing method was very effective. Processor utilization improved from about 11% to 65% (similar to the levels seen before implementing the bounding box tests). The remaining idleness is attributable to some point-triangle or line-line tests being more expensive to compute than others (early exits, etc.), or to operating system interrupts. If the number of triangles per processor is high enough, load balancing improves to about 95% (as in weak-scaling benchmark results). With the load balancing in place, derivatives gained another **2.6x** speedup versus the unbalanced bounding box benchmark. However, I noted that weak scaling seemed to suffer somewhat for this iteration.

After conducting some detailed profiling, I noted that the load balancing step was increasing in time as the number of triangles increased. The reason was the MPI Allgather operation I used to construct the global load balancing vector. My reasoning for this was to save bounding box test results so they wouldn't need to be recomputed later in the main loop. However, it was clear that this would potentially cause poor scaling for larger problems.

B.6.2 Distributed Load Balancing

In order to eliminate the Allgather operation from the load balancing step, two changes were required. First, I had to develop a distributed scan methodology to determine where the load balancing break indices would be located, without gathering a global vector of bounding box test results. I was able to do this using an Allreduce operation with only as many values as processors, eliminating the weak scaling issue. The second change was to insert the bounding box tests back into the main KS computation loop. This resulted in doing some redundant bounding box tests,

but they are so cheap that it barely matters.

I ran the benchmark for this distributed load balancing variant and found that the processor utilization remained equally high, and that the poor weak scaling of the load balancing step had disappeared. Distributing the load balancing accounts for only about a **1.1x** speedup compared to the global load balancing, but it eliminates a sub-optimal scaling property that could come back to haunt me in the future when working with larger problems.

B.7 Performance

The performance of my parallel Fortran implementation of the algorithm was excellent compared to the TensorFlow implementation. Table B.1 shows that the derivative computation could be accelerated more than 500x, while analysis-only improved three orders of magnitude. This is a breakthrough in performance which will enable me to tackle more complex problems in my research, such as the multiple-tank aerostructural packing problem in Chapter 10.

Table B.1 shows the relative contribution of each optimization step, both for the full derivative computation as well as for analysis only (without derivatives). The improved memory access pattern versus TensorFlow accounts for most of the speedup in the baseline case. The "1 of 15" optimization only really improved the derivative code, but it was a sizable speedup. The bounding box test, properly load balanced, accounts for one more order of magnitude speedup.

The scaling studies were mixed. Because the derivative code has to do a large allgather and allreduce to assemble the derivative output, there is no way that perfect strong or weak scaling can be achieved. We have more hope for the analysis-only code which has a scalar allreduce in the middle. Unfortunately there is a steep drop off in strong and weak scaling for the derivative code. Table B.2 shows that efficiency drops to only 17% for the largest case. For this number of processors, the collective operations take almost as much time as the actual loop work. On the other hand, the analysis-only routine scales reasonably well. At 48 processors, the amount of work

	Speedup versus TensorFlow	
Derivatives	Cumulative	Relative
Baseline	19.7	19.7
1 of 15	63.9	3.2
Bounding Box	217.5	3.4
Balanced Bounding Box	563.8	2.6
Analysis Only		
Baseline	53.9	53.9
1 of 15	59.3	1.1
Balanced Bounding Box	1254.7	21.1
Distributed LB BB	1388.9	1.1

Table B.1: Speedup on 48 processors versus the original TensorFlow implementation

per processor starts to get pretty small and the chunks of work are not large enough to hide jitter and differences in work due to control flow and general "luck". We saw this as a drop in processor utilization from 95% to about 65%. This accounts for most of the loss of efficiency - it is not due to communication overhead.

We would expect better weak scaling for the analysis-only code due to little collective communication, and that turns out to be correct. Table B.3 shows that the weak scaling efficiency for both codes is pretty good. We see a modest improvement in weak scaling efficiency due to the distributed load balancing, particularly for the analysis-only case. The load-balancing time in these studies is basically zero and I confirmed high processor utilization across all the cases, so the only reason I can think of for imperfect and decreasing weak scaling is OS interrupts or potentially the instrumentation I inserted to collect run time data (which uses some collectives by necessity).

B.7.1 Future Improvement Ideas

I am more than satisfied with the performance of the code, which is scalable enough for our purposes and certainly fast enough. However, I have two ideas to improve scaling for the derivative code. First, I need to alleviate the requirement for allgathering and allreducing the huge

Derivatives	all results are wall time in s				
Number of processors	48	24	16	9	4
TensorFlow	38				
Baseline	1.925	3.121	4.430	7.823	17.111
1 of 15	0.595	0.783	1.089	1.905	3.953
Bounding Box	0.175	0.169	0.175	0.224	0.282
Balanced Bounding Box	0.067	0.050	0.052	0.076	0.140
Best strong scaling efficiency	17%	46%	67%	82%	100%
Analysis only					
Number of processors	48	24	16	9	4
TensorFlow	20				
Baseline	0.371	0.634	0.991	1.567	3.399
1 of 15	0.337	0.577	0.885	1.426	3.026
Balanced Bounding Box	0.016	0.022	0.028	0.047	0.098
Distributed LB BB	0.014	0.021	0.029	0.046	0.097
Best strong scaling efficiency	56%	76%	85%	94%	100%

Table B.2: Strong scaling results for various optimization levels

Jacobian arrays onto each process which will inevitably harm both strong and weak scaling. If I can pass in the smaller array $d\mathbf{A}/d\mathbf{x}$ from outside, I can compute the product inside the Fortran routine and allgather / allreduce a much smaller number of entries of $dKS/d\mathbf{x}$. This should improve both strong and weak scaling for the derivatives code and is very straightforward to implement.

Second, I can increase the amount of serial work done per processor depending on the design problem. Future work could involve optimizing aircraft geometries which pack dozens or hundreds of inner objects inside. In that case, I can create a manager-worker system to allocate one (or maybe a few) processors per object. This will reduce the number of allreduce operations that are required to assemble the derivatives and increase processor utilization by increasing the size of the average work chunk.

Derivatives	48	24	16	9	4
Global load balancing (time, s)	0.347	0.200	0.169	0.152	0.137
Weak scaling efficiency	39.4%	68.3%	80.8%	89.7%	100.0%
Distributed load balancing (time, s)	0.332	0.199	0.169	0.152	0.141
Weak scaling efficiency	42.4%	70.6%	83.4%	92.6%	100.0%
Analysis only	48	24	16	9	4
Global load balancing (time, s)	0.138	0.111	0.102	0.101	0.098
Weak scaling efficiency	71.3%	88.7%	96.3%	97.5%	100.0%
Distributed load balancing (time, s)	0.127	0.110	0.103	0.100	0.098
Weak scaling efficiency	77.7%	89.8%	95.7%	98.0%	100.0%

Table B.3: Weak scaling results for various optimization levels

B.8 Concluding Remarks

I demonstrated the design, development, verification, and testing of a Fortran-based geometry constraint code for gradient-based optimization. The code has a necessarily-complicated build process involving algorithmic differentiation, complex step derivative verification, and a Python-Fortran interface. Through careful memory access design, a 50x speedup was achieved compared to the already well-designed TensorFlow implementation. Subsequent algorithm improvements in the Fortran code increased the speedup to as much as 1350x. Changes to the algorithm introduced a dynamic load balancing problem, which was solved using a scalable load balancing mechanism. Strong and weak scaling were good for the analysis-only code, but suffered for the derivative code due to required collective communication. Future work could alleviate software interface constraints and reduce the scaling penalty on the derivative code by reducing collective communication substantially.

APPENDIX C

CAD-based Optimization Geometry with Engineering Sketch Pad

C.1 Introduction

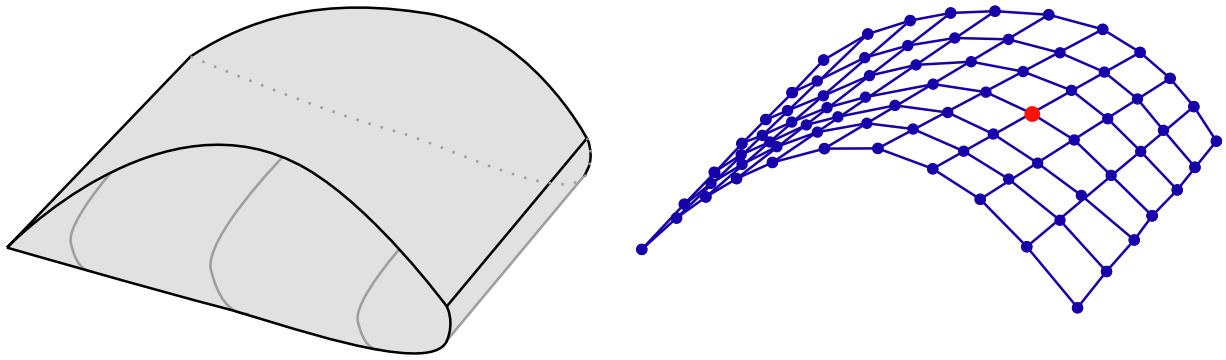
Shape optimization with high-fidelity physics requires a *geometry engine* to translate changes in geometric design variables into whatever geometric representation is used by the physical simulations. For example, in aerodynamic shape optimization of a wing, the geometry engine needs to convert changes in wingspan or aspect ratio into deformations of the CFD surface mesh. In a structural optimization, the geometry engine computes the locations of the finite element nodes. Historically, the MACH framework has relied on a CAD-free parameterization [228] based on the free-form deformation method. However, it is difficult to translate shapes optimized with FFD into buildable geometries for industrial use, because industrial fabrication relies on CAD-based models.

For this dissertation, I developed a CAD-based geometry engine compatible with the MACH aerostructural optimization workflow, known as `DVGeoESP`. This appendix describes the general approach, but is not an exhaustive list of low-level computational tricks, nor an API reference.

C.2 CAD Surface Description

The first step in a CAD-based shape optimization is to create a baseline geometry in ESP. The baseline geometry should be consistent with the requirements of the analysis method(s) to be used. For example, geometry to be used with CFD should generally be a "watertight" enclosed surface. Finite element geometry should include surfaces coincident with the intended location of the idealized elements (e.g. ribs, spars, skins). All points in the baseline computational model must be coincident with a surface in the CAD model.

A notional CAD surface is pictured in Figure C.1a. A CFD mesh can be generated using the CAD surface with an external program such as Pointwise or ICEM CFD. Alternatively, if the geometry of an existing CFD mesh is known, a new CAD surface can be created that matches the existing mesh. Figure C.1b illustrates a CFD mesh corresponding to the notional CAD object.



(a) Notional CAD surface

(b) CFD mesh discretization of CAD surface

Figure C.1: CAD model and CFD mesh in undeformed state

The CAD object is segmented into one or more B-spline surfaces. Each spline surface has a two-dimensional parameterization (denoted by the variables u and v). Engineering Sketch Pad typically defines u and v on a range of 0 to 1, but circular or spherical shapes may vary from 0 to 2π or multiples thereof. Figure C.2 shows a schematic illustration of the u, v parameterization on a B-spline surface. Each point on the surface can be uniquely defined in u, v coordinates.

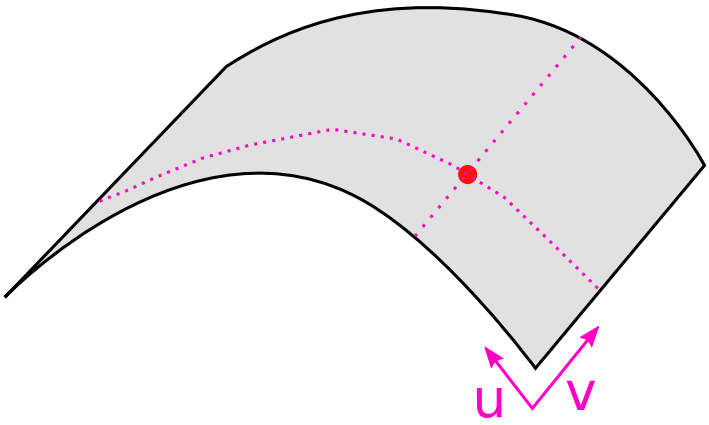


Figure C.2: Parametric coordinates on a B spline surface

C.3 Adding a Point Set

A point set is simply a list of coordinate points which lie on the CAD object's surface. A point set may represent the nodal coordinates of a finite element model or mesh points of a CFD model. Upon initialization, point sets must be associated to the CAD surface. This is done by storing the unique parametric coordinates associated with each point in the point set.

Let $\boldsymbol{X}$ be a list of coordinates of length ℓ and column dimension 3. The i-th point $\boldsymbol{X}_i$ can be obtained as:

$$\boldsymbol{X}_i = f(\boldsymbol{x}, u_i, v_i, S_i) \tag{C.1}$$

where $f()$ is the CAD model, $\boldsymbol{x}$ is the vector of geometric design variables, and u_i and v_i are the parametric coordinates of the point on S_i, the i-th B-spline surface. On initialization, the u and v coordinates are obtained using an inverse lookup procedure available in Engineering Sketch Pad which can be denoted as:

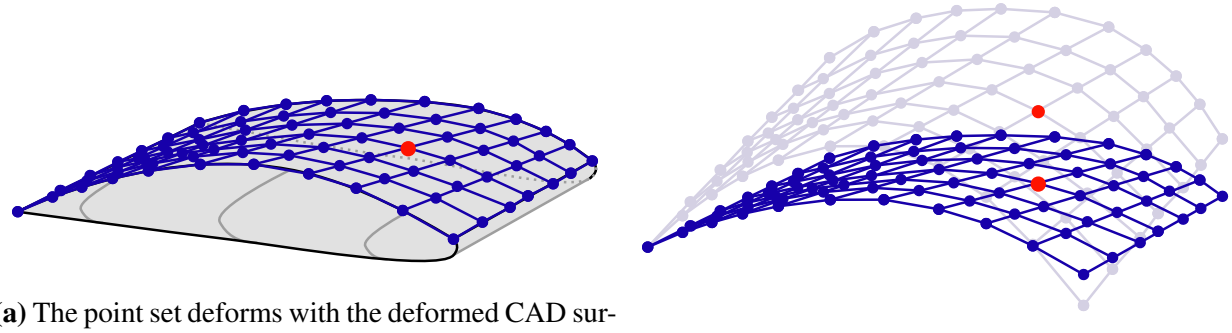
$$(u_i, v_i) = h(\boldsymbol{x_0}, \boldsymbol{X}_i, S_i) \tag{C.2}$$

where $\boldsymbol{x_0}$ are the design variables for the base case. The inverse lookup function h is expensive, so this is done only once on initialization. The point set can then be saved as a list of parametric

coordinates $(\boldsymbol{u}, \boldsymbol{v})$ along with the index of the CAD B-spline surface which had the closest match, $\boldsymbol{S}$. DVGeometryESP will return an error if the minimum distance from a point to *any* B-spline surface is more than some tolerance.

C.4 Updating a Point Set

During optimization, the geometric design variables $\boldsymbol{x}$ will change, thus causing the B-spline geometry to change. This requires updating the point set coordinates. The (u, v) parametric coordinates remain constant during optimization, thus preserving relative locations of mesh points and the overall topology. Figure C.3a shows the deformed CAD object with the associated mesh points. Figure C.3b compares the CFD mesh before and after the geometry changes.



(a) The point set deforms with the deformed CAD surface

(b) Mesh topology is preserved across CAD geometry changes

Figure C.3: Post-deformation CAD surface and CFD mesh

C.5 Computing Derivatives

For optimization, the framework needs derivatives of the CFD mesh points with respect to the design variables, or:

$$\frac{d\boldsymbol{X}_{ij}}{d\boldsymbol{x}_k} \tag{C.3}$$

This can be expressed as a tensor, but in pyGeo we flatten the spatial dimension so it becomes a Jacobian. The tensor entries are computed using the finite difference method in parallel. Each processor is evenly allocated a number of design variables, and for each variable the entire point set is deformed across a small finite difference step before being reset. The full Jacobian is subsequently allgathered to every processor, unless the point set is distributed. In that case, only the Jacobian rows stored on a processor are gathered there.

APPENDIX D

Fluid Systems Weight and Performance Data

This appendix contains tabulated and computed data for low-pressure aerospace hoses as well as pumps. Public data on aerospace components can be difficult to find, and I was unable to locate coolant-specific pump data for aircraft. However, Eaton Corporation publishes substantial data on its fuel pumps, which are similar to the flow rates and pressures required for coolant applications, so I used these for benchmarking. Tabulating the hose data was necessary in order to back-calculate the material density and design material stress for a realistic aerospace-rated hose.

Model No	Type	Voltage	Current (A)	Flow (pph)	Pressure (PSIG)	Weight (lb)	Elec Power (W)	Flow (kg/s)	Fluid Power (W)	Pump Efficiency	Elec Spec Power (W/kg)
568-1-30780	Fuel	24	7	1500	30	6.44	168	0.19	47.7	28.38%	57.5
568-1-29407	Fuel	23	13	3150	27	5.2	299	0.40	90.1	30.14%	126.8
39-0001-1100	VF Boost	200	14	30000	22	17.25	2800	3.78	699.2	24.97%	357.9
568-1-24730	Boost	200	7.1	39000	12	7.5	1420	4.91	495.8	34.92%	417.4
568-1-30685	Boost (747)	200	7	28000	18	9.3	1400	3.53	533.9	38.14%	331.9
568-1-27202	Boost (A320)	200	7	10000	33	11	1400	1.26	349.6	24.97%	280.6
568-1-24190	Transfer	200	4.9	23200	10	3.6	980	2.92	245.8	25.08%	600.2

Table D.1: Eaton Corporation Fuel Pump Data and Computed Quantities [291]

Model No	d_{inner} **(in)**	d_{outer} **(in)**	t_{wall}	**Density (kg/**m^3**)**	**Operating Press. (psi)**	**Burst Press. (psi)**	**Weight (lb/in)**	**Operating Stress (psi)**	**Burst Stress (psi)**
306-2	0.125	0.344	0.110	0.0496	300	2000	0.004	171.2	1141.6
306-3	0.188	0.406	0.109	0.0492	250	1700	0.005	215.6	1466.1
306-4	0.25	0.469	0.110	0.0485	200	1250	0.006	228.3	1426.9
306-6	0.375	0.594	0.110	0.0480	150	1000	0.008	256.8	1712.3
306-8	0.5	0.75	0.125	0.0489	150	750	0.012	300.0	1500.0
306-10	0.625	0.875	0.125	0.0475	150	700	0.014	375.0	1750.0

Table D.2: Eaton Corporation Low-Pressure Hose Data and Computed Quantities [212]

Bibliography

[1] B. J. Brelje, J. R. R. A. Martins, Development of a conceptual design model for aircraft electric propulsion with efficient gradients, in: Proceedings of the AIAA/IEEE Electric Aircraft Technologies Symposium, Cincinnati, OH, 2018. doi:10.2514/6.2018-4979.

[2] International Civil Aviation Organization, Technology Standards (2011).
URL http://www.icao.int/environmental-protection/Pages/technology-standards.aspx

[3] International Civil Aviation Organization (ICAO), Consolidated statement of continuing ICAO policies and practices related to environmental protection – Global Market-based Measure (MBM) scheme (2016).

[4] M. D. Guynn, J. J. Berton, M. J. Tong, W. J. Haller, Advanced Single-Aisle Transport Propulsion Design Options Revisited, in: 13th AIAA Aviation Technology, Integration, and Operations Conference, Los Angeles, CA, 2013. doi:10.2514/6.2013-4330.

[5] G. J. Follen, R. Del Rosario, R. A. Wahls, N. K. Madavan, NASA's Fundamental Aeronautics Subsonic Fixed Wing Project: Generation N+3 Technology Portfolio, in: SAE 2011 AeroTech Congress and Exhibition, 2011. doi:10.4271/2011-01-2521.

[6] U.S. Energy Information Administration, U.S. Gulf Coast Kerosene-Type Jet Fuel Spot Price FOB (2021).
URL https://www.eia.gov/petroleum/

[7] U.S. Energy Information Administration, Wholesale Electricity and Natural Gas Market Data (2021).
URL https://www.eia.gov/electricity/wholesale/

[8] R. Thomson, N. Sachdeva, M. Nazukin, N. Martinez, Aircraft Electrical Propulsion – The Next Chapter of Aviation? (2017).
URL https://www.rolandberger.com/en/Publications/pub_new_trends_in_electric_aircraft.html

[9] M. Hepperle, Electric Flight – Potential and Limitations, Tech. rep., NATO, Braunschweig (2012).

[10] C. Pornet, Electric Drives for Propulsion System of Transport Aircraft, in: M. Chromat (Ed.), New Applications of Electric Drives, InTech, 2015, Ch. 5.

[11] National Academy of Engineering Committee on Propulsion and Energy Systems to Reduce Commercial Aviation Carbon Emissions, Commercial Aircraft Propulsion and Energy Systems Research: Reducing Global Carbon Emissions, National Academies Press, 2016. doi:10.17226/23490.

[12] A. S. Gohardani, G. Doulgeris, R. Singh, Challenges of future aircraft propulsion: A review of distributed propulsion technology and its potential application for the all electric commercial aircraft, Progress in Aerospace Sciences 47 (5) (2011) 369–391. doi:10.1016/j.paerosci.2010.09.001.

[13] A. S. Gohardani, A synergistic glance at the prospects of distributed propulsion technology and the electric aircraft concept for future unmanned air vehicles and commercial/military aviation, Progress in Aerospace Sciences 57 (2013) 25 – 70. doi:10.1016/j.paerosci.2012.08.001.

[14] B. Sarlioglu, C. T. Morris, More Electric Aircraft: Review, Challenges, and Opportunities for Commercial Transport Aircraft, IEEE Transactions on Transportation Electrification 1 (1) (2015) 54–64. doi:10.1109/TTE.2015.2426499.

[15] C. A. Perullo, D. N. Mavris, A review of hybrid-electric energy management and its inclusion in vehicle sizing, Aircraft Engineering and Aerospace Technology 86 (6) (2014) 550–557. doi:10.1108/AEAT-04-2014-0041.

[16] T. J. Wall, R. Meyer, A Survey of Hybrid Electric Propulsion for Aircraft, in: 53rd AIAA/SAE/ASEE Joint Propulsion Conference, Atlanta, GA, 2017, pp. 1–15. doi:10.2514/6.2017-4700.

[17] B. Brelje, J. R. R. A. Martins, Electric, hybrid, and turboelectric fixed-wing aircraft: A review of concepts, models, and design approaches, Progress in Aerospace Sciences 104 (2019) 1–19. doi:10.1016/j.paerosci.2018.06.004.

[18] A. T. Isikveren, S. Kaiser, C. Pornet, P. C. Vratny, Pre-design strategies and sizing techniques for dual-energy aircraft, Aircraft Engineering and Aerospace Technology 86 (6) (2014) 525–542. doi:10.1108/AEAT-08-2014-0122.

[19] M. K. Bradley, C. K. Droney, Subsonic Ultra Green Aircraft Research: Phase II – Volume II – Hybrid Electric Design Exploration, NASA/CR–2015-218704 (2015).
URL https://ntrs.nasa.gov

[20] J. Thauvin, G. Barraud, M. Budinger, D. Leray, X. Roboam, B. Sareni, Hybrid Regional Aircraft: A Comparative Review of New Potentials Enabled by Electric Power, in: 52nd AIAA/SAE/ASEE Joint Propulsion Conference, Salt Lake City, UT, 2016. doi:10.2514/6.2016-4612.

[21] J. K. Ausserer, F. G. Harmon, Integration, Validation, and Testing of a Hybrid-Electric Propulsion System for a Small Remotely Piloted Aircraft, in: 10th International Energy Conversion Engineering Conference, Atlanta, GA, 2012. doi:10.2514/6.2012-4239.

[22] C. Pornet, C. Gologan, P. C. Vratny, A. Seitz, O. Schmitz, A. T. Isikveren, M. Hornung, Methodology for Sizing and Performance Assessment of Hybrid Energy Aircraft, Journal of Aircraft 52 (1) (2015) 341–352. doi:10.2514/1.C032716.

[23] J. Rheaume, C. Lents, Energy Storage for Commercial Hybrid Electric Aircraft, SAE Technical Paper 2016-01-2014 (2016). doi:10.4271/2016-01-2014.

[24] I. Geiss, R. Voit-Nitschmann, Sizing of the energy storage system of hybrid-electric aircraft in general aviation, CEAS Aeronautical Journal 8 (1) (2017) 53–65. doi:10.1007/s13272-016-0220-5.

[25] K. R. Antcliff, M. D. Guynn, T. Marien, D. P. Wells, S. J. Schneider, M. T. Tong, Mission Analysis and Aircraft Sizing of a Hybrid-Electric Regional Aircraft, in: 54th AIAA Aerospace Sciences Meeting, San Diego, CA, 2016. doi:10.2514/6.2016-1028.

[26] C. Pornet, A. T. Isikveren, Conceptual design of hybrid-electric transport aircraft, Progress in Aerospace Sciences 79 (2015) 114–135. doi:10.1016/j.paerosci.2015.09.002.

[27] A. T. Isikveren, C. Pornet, P. C. Vratny, M. Schmidt, Optimization of Commercial Aircraft Using Battery-Based Voltaic-Joule/Brayton Propulsion, Journal of Aircraft 54 (1) (2017) 246–261. doi:10.2514/1.C033885.

[28] M. Marwa, B. Martos, S. M. Martin, R. Anderson, Analytic and Numeric Forms for the Performance of Propeller-Powered Electric and Hybrid Aircraft, in: 55th AIAA Aerospace Sciences Meeting, Grapevine, TX, 2017, pp. 1–37. doi:10.2514/6.2017-0211.

[29] R. H. Jansen, C. Bowman, A. Jankovsky, R. Dyson, J. L. Felder, Overview of NASA Electrified Aircraft Propulsion (EAP) Research for Large Subsonic Transports, in: 53rd AIAA/SAE/ASEE Joint Propulsion Conference, Atlanta, GA, 2017, pp. 1–20. doi:10.2514/6.2017-4701.

[30] N. Xue, W. Du, J. R. R. A. Martins, W. Shyy, Handbook of Clean Energy Systems, Vol. 5 : Energy Storage, John Wiley & Sons, Ltd, 2015, Ch. 26: Lithium-Ion Batteries: Thermo-Mechanics, Performance, and Design Optimization, pp. 2849–2864. doi:10.1002/9781118991978.

[31] L. A. Costello, State of the art of piloted electric airplanes, NASA's centennial challenge data and fundamental design implications, Master's Thesis; Embry-Riddle Aeronautical University Scholarly Commons (2011).
URL http://commons.erau.edu/cgi/viewcontent.cgi?article=1036&context=edt

[32] Electric Aircraft Corporation, Electric Aircraft Corporation: Home of the ElectraFlyer (2018).
URL http://www.electraflyer.com/

[33] N. Lapena-Rey, J. Mosquera, E. Bataller, F. Ortí, First Fuel-Cell Manned Aircraft, Journal of Aircraft 47 (6) (2010) 1825–1835. doi:10.2514/1.42234.

[34] C. Riboldi, F. Gualdoni, An integrated approach to the preliminary weight sizing of small electric aircraft, Aerospace Science and Technology 58 (2016) 134–149. doi:10.1016/j.ast.2016.07.014.

[35] Diamond Aircraft, About Diamond - Innovation (2017).
URL http://www.diamondaircraft.com/about-diamond/innovation/

[36] Pipistrel, Taurus Electro G2 Technical Data (2017).
URL http://www.pipistrel.si/plane/taurus-electro/technical-data

[37] J. W. Langelaan, A. Chakrabarty, A. Deng, K. Miles, V. Plevnik, J. Tomazic, T. Tumazic, G. Veble, Green Flight Challenge : Aircraft Design and Flight Planning for Extreme Fuel Efficiency, Journal of Aircraft 50 (3) (2013). doi:10.2514/1.C032022.

[38] A. Nanda, The Propulsive Design Aspects on the World's First Direct Drive Hybrid Airplane, Master's Thesis; Embry-Riddle Aeronautical University Scholarly Commons (2011).

[39] W. C. Yates, Personal Communication (2017).

[40] K. Petermaier, Electric propulsion components with high power densities for aviation, in: 2015 Transformational Vertical Flight Workshop, Siemens AG, 2015.
URL https://nari.arc.nasa.gov/sites/default/files/attachments/Korbinian-TVFW-Aug2015.pdf

[41] Airbus Group, E-Fan: The New Way to Fly, Brochure (2015).
URL http://company.airbus.com/service/mediacenter/download/?uuid=48b1bd2c-a428-4c65-82e5-ed3e923bd142

[42] Airbus Group, E-Fan: toward more electric planes (2017).
URL http://company.airbus.com/responsibility/airbus-e-fan-the-future-of-electric-aircraft/Programme.html

[43] C. Friedrich, P. Robertson, Hybrid-Electric Propulsion for Aircraft, Journal of Aircraft 52 (1) (2015) 176–189. doi:10.2514/1.C032660.

[44] University of Cambridge, Watts up - aeroplanes go hybrid-electric (2014).
URL http://www.cam.ac.uk/research/news/watts-up-aeroplanes-go-hybrid-electric

[45] T. A. Horne, Pipistrel Alpha Electro: The Trainer of the Future?, AOPA Pilot (October 2017).
URL https://www.aopa.org/news-and-media/all-news/2015/october/pilot/f_pipistrel/

[46] Pipistrel, Pipistel Alpha Electro Information Pack Rev 05 (2017).
URL www.flypipistrel.com/info-packs/Pipistrel-Alpha-ELECTRO-Information-Pack.pdf

[47] Siemens AG, Aerobatic Airplane "Extra 330LE" (2016).
URL https://www.siemens.com/press/pool/de/events/2016/corporate/2016-12-innovation/inno2016-aerobatic-airplane-e.pdf

[48] N. K. Borer, M. D. Patterson, J. K. Viken, M. D. Moore, J. Bevirt, A. M. Stoll, A. R. Gibson, Design and Performance of the NASA SCEPTOR Distributed Electric Propulsion Flight Demonstrator, in: 16th AIAA Aviation Technology, Integration, and Operations Conference, Washington, DC, 2016. doi:10.2514/6.2016-3920.

[49] S. Clarke, M. Redifer, K. V. Papathakis, A. Samuel, T. Foster, X-57 power and command system design, in: 2017 IEEE Transportation and Electrification Conference and Expo, ITEC 2017, 2017, pp. 393–400. doi:10.1109/ITEC.2017.7993303.

[50] K. V. Papathakis, P. A. Burkhardt, D. W. Ehmann, A. M. Sessions, Safety Considerations for Electric, Hybrid-Electric, and Turbo-Electric Distributed Propulsion Aircraft Testbeds, in: 53rd AIAA/SAE/ASEE Joint Propulsion Conference, Atlanta, GA, 2017. doi:10.2514/6.2017-5032.

[51] J. Welstead, J. L. Felder, Conceptual Design of a Single-Aisle Turboelectric Commercial Transport with Fuselage Boundary Layer Ingestion, in: 54th AIAA Aerospace Sciences Meeting, San Diego, CA, 2016. doi:10.2514/6.2016-1027.

[52] J. Welstead, J. L. Felder, M. D. Guynn, W. Haller, M. T. Tong, S. Jones, I. Ordaz, J. Quinlan, B. Mason, Overview of the NASA STARC-ABL (Rev . B) Advanced Concept (2017).
URL https://ntrs.nasa.gov/search.jsp?R=20170005612

[53] J. L. Felder, G. V. Brown, H. D. Kim, J. Chu, Turboelectric distributed propulsion in a hybrid wing body aircraft, in: International Symposium on Air Breathing Engines 2011, 2011.

[54] M. J. Armstrong, C. A. H. Ross, M. J. Blackwelder, Trade Studies for NASA N3-X Turboelectric Distributed Propulsion System Electrical Power System Architecture, SAE International Journal of Aerospace 5 (2) (2012) 325–335. doi:10.4271/2012-01-2163.

[55] H. D. Kim, J. L. Felder, M. T. Tong, M. J. Armstrong, Revolutionary Aeropropulsion Concept for Sustainable Aviation: Turboelectric Distributed Propulsion, in: 21st International Symposium on Air Breathing Engines (ISABE), Busan, Korea, 2013.

[56] H. D. Kim, J. L. Felder, M. T. Tong, J. J. Berton, W. Haller, Turboelectric distributed propulsion benefits on the N3-X vehicle, Aircraft Engineering and Aerospace Technology 86 (6) (2014) 558–561. doi:10.1108/AEAT-04-2014-0037.

[57] A. Jankovsky, C. Bowman, R. H. Jansen, Building Blocks for Transport-Class Hybrid and Turboelectric Vehicles, NASA Presentation (2016).
URL https://ntrs.nasa.gov/search.jsp?R=20170006238

[58] J. L. Felder, NASA Electric Propulsion System Studies, NASA Presentation (2017).
URL https://ntrs.nasa.gov/search.jsp?R=20160009274

[59] M. Hornung, A. T. Isikveren, M. Cole, A. Sizmann, Ce-Liner – Case Study for eMobility in Air Transportation, in: 13th AIAA Aviation Technology, Integration, and Operations Conference, Los Angeles, CA, 2013. doi:10.2514/6.2013-4302.

[60] A. T. Isikveren, A. Seitz, P. C. Vratny, C. Pornet, K. O. Plötner, M. Hornung, Conceptual Studies of Universally-Electric Systems Architectures Suitable for Transport Aircraft, in: Deutscher Luft- und Raumfahrtkongress, Berlin, Germany, 2012.

[61] S. Stückl, J. van Toor, H. Lobentanzer, VOLTAIR - The All Electric Propulsion Concept Platform – A Vision For Atmospheric Friendly Flight, in: 28th Congress of the International Council of the Aeronautical Sciences, Brisbane, Australia, 2012.

[62] EADS, VoltAir All-electric Transport Concept Platform, Brochure.
URL https://www.airbusgroup.com/service/mediacenter/download/?uuid=1af7b378-404c-4345-a243-a13de66a2dbd

[63] Airbus Group and Rolls-Royce, E-Thrust, Brochure (2012).
URL http://company.airbus.com/service/mediacenter/download/?uuid=64ea2c23-91b1-4787-9d1d-5b22b7d716b9

[64] B. T. Schiltgen, J. Freeman, Aeropropulsive Interaction and Thermal System Integration within the ECO-150: A Turboelectric Distributed Propulsion Airliner with Conventional Electric Machines, in: 16th AIAA Aviation Technology, Integration, and Operations Conference, Washington, DC, 2016. doi:10.2514/6.2016-4064.

[65] Eviation, Alice Commuter (2018).
URL eviation.co/alice/

[66] Eviation, In a Major Step Toward Fully Electric Aviation, Eviation and Kokam Announce Battery Supply Deal Worth Over 1 Million at Singapore Airshow, Press Release (feb 2018).
URL https://www.prnewswire.com/news-releases/in-a-major-step-toward-fully-electric-aviation-eviation-and-kokam-announce-battery-supply-deal-worth-over-1-million-at-singapore-airshow-300594703.html

[67] T. B. Haines, XTI Trifan 600 Draws the Curious at NBAA, AOPA (2017).
URL https://www.aopa.org/news-and-media/all-news/2017/october/11/xti-trifan-600-draws-the-curious-at-nbaa

[68] C. Combs, Personal Communication (2017).

[69] Zunum Aero, Zunum Aero - Aircraft (2017).
URL http://zunum.aero/aircraft/

[70] B. Fehrm, Airbus , Rolls-Royce and Siemens develops Hybrid-Electric demonstrator, Leeham News and Comment (2017).
URL https://leehamnews.com/2017/11/29/airbus-rolls-royce-siemens-develops-hybrid-electric-demonstrator/

[71] K. J. Karimi, Future Aircraft Power Systems- Integration Challenges Future Aircraft Power Systems (2007).
URL https://eng.umd.edu/$\sim$austin/ense622.d/lecture-resources/Boeing787-MoreElectricAircraft.pdf

[72] S. F. Clark, 787 Propulsion System, Boeing Aero Quarterly (2012) 13.
URL www.boeing.com/boeingedge/aeromagazine

[73] Airbus Group, Airbus, Rolls-Royce, and Siemens team up for electric future (2017).
URL http://www.airbus.com/newsroom/press-releases/en/2017/11/airbus--rolls-royce--and-siemens-team-up-for-electric-future-par.html

[74] M. K. Bradley, C. K. Droney, Subsonic ultra green aircraft research: Phase I final report, Tech. Rep. CR–2011-216847, NASA (April 2011).

[75] B. T. Schiltgen, A. R. Gibson, J. D. Keith, Mission Performance Comparisons of Subsonic Airliners with Current and Future Propulsion Technologies, in: 48th AIAA Aerospace Sciences Meeting, Orlando, FL, 2010. doi:10.2514/6.2010-279.

[76] B. Schiltgen, M. W. Green, A. R. Gibson, D. W. Hall, D. B. Cummings, Benefits and Concerns of Hybrid Electric Distributed Propulsion with Conventional Electric Machines, in: 48th AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Atlanta, GA, 2012. doi:10.2514/6.2012-3769.

[77] B. T. Schiltgen, A. R. Gibson, M. W. Green, J. Freeman, More Electric Aircraft: "Tube and Wing" Hybrid Electric Distributed Propulsion with Superconducting and Conventional Electric Machines, SAE Technical Paper 2013-01-2306 (2013). doi:10.4271/2013-01-2306.

[78] M. W. Green, B. T. Schiltgen, A. R. Gibson, Analysis of a Distributed Hybrid Propulsion System with Conventional Electric Machines, in: 48th AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Atlanta, GA, 2012. doi:10.2514/6.2012-3768.

[79] V. Mukhopadhyay, T. A. Ozoroski, M. L. Mcmillin, Structural Configuration Analysis of Advanced Flight Vehicle Concepts with Distributed Hybrid-Electric Propulsion, in: 2018 AIAA Aerospace Sciences Meeting, Orlando, FL, 2018. doi:10.2514/6.2018-1747.

[80] M. D. Moore, B. Fredericks, Misconceptions of electric aircraft and their emerging aviation markets, in: 52nd Aerospace Sciences Meeting, no. AIAA Paper 2014-0535, AIAA, 2014. doi:10.2514/6.2014-0535.

[81] J. Murray, J. Lechniak, The LEAPTech Experiment: Approach, Results, Recommendations, NASA Presentation (2016).
URL https://ntrs.nasa.gov/search.jsp?R=20160012394

[82] K. A. Deere, S. Viken, M. Carter, J. K. Viken, J. M. Derlaga, A. M. Stoll, Comparison of High-Fidelity Computational Tools for Wing Design of a Distributed Electric Propulsion Aircraft, in: 35th AIAA Applied Aerodynamics Conference, Denver, CO, 2017. doi:10.2514/6.2017-3925.

[83] J. Bennett, NASA's Next Great X-Plane Will Try to Revolutionize Electric Flight, Popular Mechanics (July 2017).
URL http://www.popularmechanics.com/flight/a26609/nasa-plane-electric-x-57-maxwell/

[84] T. Risen, For NASA's X-57 Electric Plane, More Battery Tests Ahead, Aerospace America (July 2017).
URL https://aerospaceamerica.aiaa.org/x57-battery-tests/

[85] H. D. Kim, G. V. Brown, J. L. Felder, Distributed Turboelectric Propulsion for Hybrid Wing Body Aircraft, in: 2008 International Powered Lift Conference, London, UK, 2008.
URL http://mdao.grc.nasa.gov/publications/IPLF08-Kim.pdf

[86] J. Felder, H. Kim, G. Brown, Turboelectric Distributed Propulsion Engine Cycle Analysis for Hybrid-Wing-Body Aircraft, in: 47th AIAA Aerospace Sciences Meeting, Orlando, FL, 2009, pp. 1–25. doi:10.2514/6.2009-1132.

[87] J. L. Felder, M. T. Tong, J. Chu, Sensitivity of mission energy consumption to turboelectric distributed propulsion design assumptions on the N3-X hybrid wing body aircraft, in: 48th AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Atlanta, GA, 2012. doi:10.2514/6.2012-3701.

[88] G. V. Brown, Weights and Efficiencies of Electric Components of a Turboelectric Aircraft Propulsion System, in: 49th AIAA Aerospace Sciences Meeting, Orlando, FL, 2011. doi:10.2514/6.2011-225.

[89] J. J. Berton, W. J. Haller, A Noise and Emissions Assessment of the N3-X Transport, in: 52nd Aerospace Sciences Meeting, National Harbor, MD, 2014. doi:10.2514/6.2014-0594.

[90] M. J. Armstrong, C. A. H. Ross, M. J. Blackwelder, Propulsion System Component Considerations for NASA N3-X Turboelectric Distributed Propulsion System, SAE International Journal of Aerospace 5 (2) (2012) 344–353. doi:10.4271/2012-01-2165.

[91] C. Ross, M. Armstrong, M. Blackwelder, C. Jones, P. Norman, S. Fletcher, Turboelectric Distributed Propulsion Protection System Design Trades, SAE Technical Paper 2014-01-2141 (2014). doi:10.4271/2014-01-2141.

[92] J. C. Shaw, P. Norman, S. Galloway, G. Burt, A Method for the Evaluation of the Effectiveness of Turboelectric Distributed Propulsion Power System Architectures, SAE International Journal of Aerospace 7 (1) (2014) 35–43. doi:10.4271/2014-01-2120.

[93] K. Davies, P. Norman, C. Jones, S. Galloway, G. Burt, Modelling the Fault Behaviour of a Superconducting Turboelectric Distributed Propulsion Network Overview of Superconducting DC Networks, SAE Technical Paper 2014-01-2142 (2014). doi:10.4271/2014-01-2142.

[94] J. C. Shaw, S. Fletcher, P. Norman, S. Galloway, G. Burt, Failure Analysis of a Turboelectric Distributed Propulsion Aircraft Electrical Network : A Case Study, SAE Technical Paper 2015-01-2403 (2015). doi:10.4271/2015-01-2403.

[95] F. Berg, J. Palmer, P. Miller, M. Husband, G. Dodds, HTS electrical system for a distributed propulsion aircraft, IEEE Transactions on Applied Superconductivity 25 (3) (2015). doi:10.1109/TASC.2014.2384731.

[96] F. Berg, J. Palmer, P. Miller, G. Dodds, HTS System and Component Targets for a Distributed Aircraft Propulsion System, IEEE Transactions on Applied Superconductivity 27 (4) (2017). doi:10.1109/TASC.2017.2652319.

[97] M. Kreimeier, E. Stumpf, Benefit evaluation of hybrid electric propulsion concepts for CS-23 aircraft, CEAS Aeronautical Journal 8 (4) (2017) 691–704. doi:10.1007/s13272-017-0269-9.

[98] M. D. Olson, A Conceptual Approach to Flight - Training Mission and Cost Analysis of an All-Electric Aircraft Equipped with Regenerative Energy Devices, in: 15th AIAA Aviation Technology, Integration, and Operations Conference, Dallas, TX, 2015. doi:10.2514/6.2015-3189.

[99] Y. Fefermann, C. Maury, C. Level, K. Zarati, J.-P. Salanne, C. Pornet, B. Thoraval, A. Isikveren, Hybrid-Electric Motive Power Systems for Commuter Transport Applications, in: 30th Congress of the International Council of the Aeronautical Sciences, Daejeon, Korea, 2016.

[100] T. S. Dean, G. E. Wroblewski, P. J. Ansell, Mission Analysis and Component-Level Sensitivity Study of Hybrid-Electric General Aviation Propulsion Systems, in: 2018 AIAA Aerospace Sciences Meeting, Kissimmee, FL, 2018. doi:10.2514/6.2018-1749.

[101] F. G. Harmon, A. A. Frank, J.-J. Chattot, Conceptual Design and Simulation of a Small Hybrid-Electric Unmanned Aerial Vehicle, Journal of Aircraft 43 (5) (2006) 1490–1498. doi:10.2514/1.15816.

[102] R. Hiserote, F. G. Harmon, Analysis of Hybrid-Electric Propulsion System Designs for Small Unmanned Aircraft Systems, in: 8th Annual International Energy Conversion Engineering Conference, Nashville, TN, 2010. doi:10.2514/6.2010-6687.

[103] K. Merical, T. Beechner, P. Yelvington, Hybrid-Electric, Heavy-Fuel Propulsion System for Small Unmanned Aircraft, SAE International Journal of Aerospace 7 (1) (2014) 126–134. doi:10.4271/2014-01-2222.

[104] F. Lambert, Tesla releases impressive pictures of Model 3 drive unit after test, Musk says they drove 1M miles, Electrek (October 2018).
URL https://electrek.co/2018/10/15/tesla-drive-after-million-miles-test/

[105] C. Goldberg, D. Nalianda, P. Pilidis, R. Singh, Economic Viability Assessment of NASA's Blended Wing Body N3-X Aircraft, in: 53rd AIAA/SAE/ASEE Joint Propulsion Conference, Atlanta, Ga, 2017. doi:10.2514/6.2017-4604.

[106] R. H. Jansen, K. P. Duffy, G. V. Brown, Partially Turboelectric Aircraft Drive Key Performance Parameters, in: 53rd AIAA/SAE/ASEE Joint Propulsion Conference, Atlanta, GA, 2017. doi:10.2514/6.2017-4702.

[107] J. Gray, C. A. Mader, G. K. W. Kenway, , J. R. R. A. Martins, Approach to modeling boundary layer ingestion using a fully coupled propulsion-RANS model, in: 55th AIAA Aerospace Sciences Meeting (SciTech), Grapevine, TX, 2017. doi:10.2514/6.2017-1753.

[108] B. M. Yutko, N. Titchener, C. Courtin, M. Lieu, L. Wirsing, J. Tylko, C. T. Jeffrey, T. W. Roberts, C. S. Church, Conceptual Design of a D8 Commercial Aircraft, in: 17th AIAA Aviation Technology, Integration, and Operations Conference, Denver, CO, 2017. doi:10.2514/6.2017-3590.

[109] A. T. Wick, J. R. Hooker, C. J. Hardin, C. H. Zeune, Integrated Aerodynamic Benefits of Distributed Propulsion, in: 53rd AIAA Aerospace Sciences Meeting, Kissimmee, FL, 2015. doi:10.2514/6.2015-1500.

[110] A. M. Stoll, J. Bevirt, M. D. Moore, W. J. Fredericks, N. K. Borer, Drag reduction through distributed electric propulsion, in: 14th AIAA Aviation Technology, Integration, and Operations Conference, no. AIAA Paper 2014-2851, AIAA, 2014. doi:10.2514/6.2014-2851.

[111] K. A. Deere, J. K. Viken, S. A. Viken, M. B. Carter, M. R. Wiese, N. Farr, Computational Analysis of a Wing Designed for the X-57 Distributed Electric Propulsion Aircraft, in: 17th

AIAA Aviation Technology, Integration, and Operations Conference, Denver, CO, 2017.
URL https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20170005883.pdf

[112] K. A. Deere, S. Viken, M. Carter, J. K. Viken, M. Wiese, N. Farr, Computational Analysis of Powered Lift Augmentation for the LEAPTech Distributed Electric Propulsion Wing, in: 35th AIAA Applied Aerodynamics Conference, Denver, CO, 2017. doi:10.2514/6.2017-3921.

[113] N. K. Borer, J. M. Derlaga, K. A. Deere, M. B. Carter, S. Viken, M. D. Patterson, B. Litherland, A. Stoll, Comparison of Aero-Propulsive Performance Predictions for Distributed Propulsion Configurations, in: 55th AIAA Aerospace Sciences Meeting, Grapevine, TX, 2017. doi:10.2514/6.2017-0209.

[114] L. R. Miranda, J. E. Brennan, Aerodynamic effects of wingtip-mounted propellers and turbines, in: 4th Applied Aerodynamics Conference, AIAA, 1986. doi:10.2514/6.1986-1802.

[115] J. Freeman, P. Osterkamp, M. W. Green, A. R. Gibson, B. T. Schiltgen, Challenges and opportunities for electric aircraft thermal management, Aircraft Engineering and Aerospace Technology 86 (6) (2014) 519–524. doi:10.1108/AEAT-04-2014-0042.

[116] J. T. McKenna, XTI Aims to Fly 60 Percent Scale TriFan 600 Within Year, Rotor and Wing International (October 2017).
URL http://www.rotorandwing.com/2017/10/13/xti-aims-fly-60-scale-trifan-600-within-year

[117] Federal Aviation Administration, AC 20-184: Guidance on Testing and Installation of Rechargeable Lithium Battery and Battery Systems on Aircraft (2015).

[118] S. Arora, W. Shen, A. Kapoor, Review of mechanical design and strategic placement technique of a robust battery pack for electric vehicles, Renewable and Sustainable Energy Reviews 60 (2016) 1319–1331. doi:10.1016/j.rser.2016.03.013.

[119] M.-C. Flynn, C. Jones, P. Rakhra, P. Norman, S. Galloway, Impact of Key Design Constraints on Fault Management Strategies for Distributed Electrical Propulsion Aircraft, in: 53rd AIAA/SAE/ASEE Joint Propulsion Conference, Atlanta, GA, 2017. doi:10.2514/6.2017-5034.

[120] M. K. Bradley, C. K. Droney, Subsonic Ultra Green Aircraft Research: Phase II. N+4 Advanced Concept Development (2012). doi:2060/20150017039.
URL ntrs.nasa.gov

[121] A. M. Stoll, G. Veble Mikic, Design Studies of Thin-Haul Commuter Aircraft with Distributed Electric Propulsion, in: 16th AIAA Aviation Technology, Integration, and Operations Conference, Washington, DC, 2016. doi:10.2514/6.2016-3765.

[122] D. L. Huff, B. S. Henderson, E. Envia, Motor Noise for Electric Powered Aircraft, 22nd AIAA/CEAS Aeroacoustics Conference (2016) 1–12doi:10.2514/6.2016-2882.

[123] D. E. Bryson, C. R. Marks, R. M. Miller, M. P. Rumpfkeil, Multidisciplinary Design Optimization of Quiet, Hybrid-Electric Small Unmanned Aerial Systems, Journal of Aircraft 53 (6) (2016) 1959–1963. doi:10.2514/1.C033455.

[124] T. Donateo, A. Ficarella, Designing a Hybrid Electric Powertrain for an Unmanned Aircraft with a Commercial Optimization Software, SAE International Journal of Aerospace 10 (1) (2017) 2017–01–9000. doi:10.4271/2017-01-9000.

[125] P. Krause, O. Wasynczuk, S. Pekarek, Windings and Rotating Magnetomotive Force, 2nd Edition, IEEE, 2012, pp. 145–184.

[126] N. Xue, W. Du, A. Gupta, W. Shyy, A. M. Sastry, J. R. R. A. Martins, Optimization of a single lithium-ion battery cell with a gradient-based algorithm, Journal of the Electrochemical Society 160 (8) (2013) A1071–A1078. doi:10.1149/2.036308jes.

[127] P. C. Vratny, C. Gologan, C. Pornet, A. T. Isikveren, M. Hornung, Battery Pack Modeling Methods for Universally-Electric Aircraft, in: 4th CEAS Air and Space Conference, Linkoping, Sweden, 2013.

[128] N. K. Borer, C. L. Nickol, F. P. Jones, R. J. Yasky, K. Woodham, J. S. Fell, B. L. Litherland, P. L. Loyselle, A. J. Provenza, L. W. Kohlman, A. G. Samuel, Overcoming the Adoption Barrier to Electric Flight, in: 54th AIAA Aerospace Sciences Meeting, San Diego, CA, 2016. doi:10.2514/6.2016-1022.

[129] J. Hoelzen, Y. Liu, B. Bensmann, C. Winnefeld, A. Elham, J. Friedrichs, R. Hanke-Rauschenbach, Conceptual Design of Operation Strategies for Hybrid Electric Aircraft, Energies 11 (1) (2018) 217. doi:10.3390/en11010217.

[130] J. Tomazic, A. Zemva, Efficient and lightweight battery management system contributes to victory in the Green Flight Challenge 2011, Electric Power Systems Research 98 (2013) 70–76. doi:10.1016/j.epsr.2013.01.008.

[131] R. D. Falck, J. Chin, S. L. Schnulo, J. M. Burt, J. S. Gray, Trajectory Optimization of Electric Aircraft Subject to Subsystem Thermal Constraints, in: 18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Denver, CO, 2017. doi:10.2514/6.2017-4002.

[132] D. Trawick, C. A. Perullo, M. J. Armstrong, D. Snyder, J. C. Tai, D. N. Mavris, Development and Application of GT-HEAT for the Electrically Variable Engine Design, in: 55th AIAA Aerospace Sciences Meeting, Grapevine, TX, 2017. doi:10.2514/6.2017-1922.

[133] N. Xue, W. Du, T. A. Greszler, W. Shyy, J. R. R. A. Martins, Design of a lithium-ion battery pack for PHEV using a hybrid optimization method, Applied Energy 115 (2014) 591–602. doi:10.1016/j.apenergy.2013.10.044.

[134] G. Avanzini, E. L. D. Angelis, F. Giulietti, Optimal performance and sizing of a battery-powered aircraft, Aerospace Science and Technology 59 (2016) 132–144. doi:10.1016/j.ast.2016.10.015.

[135] P. C. Vratny, H. Kuhn, M. Hornung, Influences of voltage variations on electric power architectures for hybrid electric aircraft, CEAS Aeronautical Journal 8 (1) (2017) 31–43. doi:10.1007/s13272-016-0218-z.

[136] A. Lowe, D. Mavris, Technology Selection for Optimal Power Distribution Efficiency in a Turboelectric Propulsion System, SAE International Journal of Aerospace 5 (2) (2012) 425–437. doi:10.4271/2012-01-2180.

[137] C. E. Jones, P. J. Norman, S. J. Galloway, M. J. Armstrong, A. M. Bollman, Comparison of candidate architectures for future distributed propulsion aircraft, IEEE Transactions on Applied Superconductivity 26 (6) (2016) 1–9. doi:10.1109/TASC.2016.2530696.

[138] G. Warwick, NASA Moves Electric-Propulsion Components Closer To Reality, Aviation Week (2017).
URL http://aviationweek.com/commercial-aviation/nasa-moves-electric-propulsion-components-closer-reality

[139] D. J. Sadey, L. Taylor, R. Beach, Proposal and Development of a High Voltage Variable Frequency Alternating Current Power System for Hybrid Electric Aircraft, in: 14th International Energy Conversion Engineering Conference, Salt Lake City, UT, 2016. doi:10.2514/6.2016-4928.

[140] S. Trimble, GE reveals major achievements in hybrid electric propulsion, FlightGlobal (August 2017).

[141] S. Trimble, ANALYSIS : Key challenges in the race to develop hybrid-electric aircraft, FlightGlobal (November 2017).

[142] C. E. Lents, L. W. Hardin, J. Rheaume, L. Kohlman, Parallel Hybrid Gas-Electric Geared Turbofan Engine Conceptual Design and Benefits Analysis, in: 52nd AIAA/SAE/ASEE Joint Propulsion Conference, Salt Lake City, UT, 2016. doi:10.2514/6.2016-4610.

[143] J. R. R. A. Martins, A. B. Lambe, Multidisciplinary design optimization: A survey of architectures, AIAA Journal 51 (9) (2013) 2049–2075. doi:10.2514/1.J051895.

[144] R. T. Haftka, Optimization of flexible wing structures subject to strength and induced drag constraints, AIAA Journal 15 (8) (1977) 1101–1106. doi:10.2514/3.7400.

[145] L. A. Schmit, Jr., Structural synthesis—precursor and catalyst. recent experiences in multidisciplinary analysis and optimization, Tech. Rep. CP-2337, NASA (1984).

[146] J. R. R. A. Martins, J. J. Alonso, J. J. Reuther, High-fidelity aerostructural design optimization of a supersonic business jet, Journal of Aircraft 41 (3) (2004) 523–530. doi:10.2514/1.11478.

[147] G. K. W. Kenway, J. R. R. A. Martins, Multipoint high-fidelity aerostructural optimization of a transport aircraft configuration, Journal of Aircraft 51 (1) (2014) 144–160. doi:10.2514/1.C032150.

[148] D. A. Burdette, J. R. R. A. Martins, Design of a transonic wing with an adaptive morphing trailing edge via aerostructural optimization, Aerospace Science and Technology 81 (2018) 192–203. doi:10.1016/j.ast.2018.08.004.

[149] E. Jonsson, C. Riso, C. A. Lupp, C. E. S. Cesnik, J. R. R. A. Martins, B. I. Epureanu, Flutter and post-flutter constraints in aircraft design optimization, Progress in Aerospace Sciences 109 (2019) 100537. doi:10.1016/j.paerosci.2019.04.001.

[150] J. Agte, O. de Weck, J. Sobieszczanski-Sobieski, P. Arendsen, A. Morris, M. Spieck, MDO: Assessment and direction for advancement—an opinion of one international group, Structural and Multidisciplinary Optimization 40 (2010) 17–33. doi:10.1007/s00158-009-0381-5.

[151] G. Bower, Vahana Configuration Trade Study - Part II (2017).
URL https://vahana.aero/vahana-configuration-trade-study-part-ii-1edcdac8ad93

[152] Z. Lyu, Z. Xu, J. R. R. A. Martins, Benchmarking optimization algorithms for wing aerodynamic design optimization, in: Proceedings of the 8th International Conference on Computational Fluid Dynamics, Chengdu, Sichuan, China, 2014, iCCFD8-2014-0203.

[153] J. R. R. A. Martins, J. T. Hwang, Review and unification of methods for computing derivatives of multidisciplinary computational models, AIAA Journal 51 (11) (2013) 2582–2599. doi:10.2514/1.J052184.

[154] G. K. W. Kenway, G. J. Kennedy, J. R. R. A. Martins, Scalable parallel approach for high-fidelity steady-state aeroelastic analysis and adjoint derivative computations, AIAA Journal 52 (5) (2014) 935–951. doi:10.2514/1.J052255.

[155] J. S. Gray, J. T. Hwang, J. R. R. A. Martins, K. T. Moore, B. A. Naylor, OpenMDAO: An open-source framework for multidisciplinary design, analysis, and optimization, Structural and Multidisciplinary Optimization 59 (4) (2019) 1075–1104. doi:10.1007/s00158-019-02211-z.

[156] J. T. Hwang, J. R. R. A. Martins, A computational architecture for coupling heterogeneous numerical models and computing coupled derivatives, ACM Transactions on Mathematical Software 44 (4) (2018) Article 37. doi:10.1145/3182393.

[157] J. S. Gray, C. A. Mader, G. K. W. Kenway, J. R. R. A. Martins, Modeling boundary layer ingestion using a coupled aeropropulsive analysis, Journal of Aircraft 55 (3) (2018) 1191–1199. doi:10.2514/1.C034601.

[158] J. P. Jasa, Multidisciplinary design optimization of an aircraft considering path-dependent performance, Ph.D. thesis, University of Michigan (2020).

[159] J. C. Gladin, D. Trawick, C. A. Perullo, J. C. Tai, D. N. Mavris, Modeling and Design of a Partially Electric Distributed Aircraft Propulsion System with GT-HEAT, in: 55th AIAA Aerospace Sciences Meeting, Grapevine, TX, 2017, pp. 1–18. doi:10.2514/6.2017-1924.

[160] J. C. Gladin, C. A. Perullo, J. C. Tai, D. N. Mavris, A Parametric Study of Hybrid Electric Gas Turbine Propulsion as a Function of Aircraft Size Class and Technology Level, in: 55th AIAA Aerospace Sciences Meeting, Grapevine, TX, 2017. doi:10.2514/6.2017-0338.

[161] F. M. Capristan, J. R. Welstead, An Energy-Based Low-Order Approach for Mission Analysis of Air Vehicles in LEAPS, in: 2018 AIAA Aerospace Sciences Meeting, Kissimmee, FL, 2018. doi:10.2514/6.2018-1755.

[162] J. R. Welstead, D. Caldwell, R. Condotta, N. Monroe, An Overview of the Layered and Extensible Aircraft Performance System (LEAPS) Development, in: 2018 AIAA Aerospace Sciences Meeting, Kissimmee, FL, 2018. doi:10.2514/6.2018-1754.

[163] J. T. Hwang, A. Ning, Large-scale multidisciplinary optimization of an electric aircraft for on-demand mobility, in: 2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Kissimmee, FL, 2018, pp. 1–18. doi:10.2514/6.2018-1384.

[164] M. J. Duffy, S. R. Wakayama, R. Hupp, R. Lacy, M. Stauffer, A study in reducing the cost of vertical flight with electric propulsion, in: 17th AIAA Aviation Technology, Integration, and Operations Conference, no. AIAA Paper 2014-3442, AIAA, 2017. doi:10.2514/6.2017-3442.

[165] A. Brown, W. L. Harris, A Vehicle Design and Optimization Model for On-Demand Aviation, in: 2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Kissimmee, FL, 2018. doi:10.2514/6.2018-0105.

[166] C. A. Perullo, D. N. Mavris, Assessment of Vehicle Performance Using Integrated NPSS Hybrid Electric Propulsion Models, in: 50th AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Cleveland, OH, 2014. doi:10.2514/6.2014-3489.

[167] S. L. Schnulo, J. Chin, A. D. Smith, A. Dubois, Steady State Thermal Analyses of SCEPTOR X-57 Wingtip Propulsion, in: 17th AIAA Aviation Technology, Integration, and Operations Conference, Denver, CO, 2017, pp. 1–14. doi:10.2514/6.2017-3783.

[168] G. E. Wroblewski, P. J. Ansell, Mission Analysis and Emissions for Conventional and Hybrid-Electric Commercial Transport Aircraft, in: 2018 AIAA Aerospace Sciences Meeting, Kissimmee, FL, 2018. doi:10.2514/6.2018-2028.

[169] V. Cipolla, F. Oliviero, HyPSim : A Simulation Tool for Hybrid Aircraft Performance Analysis, in: Variational Analysis and Aerospace Engineering, 2016, pp. 95–116. doi:10.1007/978-3-319-45680-5.

[170] J. M. Vegh, J. J. Alonso, T. H. Orra, C. R. Ilario da Silva, Flight Path and Wing Optimization of Lithium-Air Battery Powered Passenger Aircraft, in: 53rd AIAA Aerospace Sciences Meeting, Kissimmee, FL, 2015. doi:10.2514/6.2015-1674.

[171] F. Capristan, D. Caldwell, R. Condotta, B. Petty, Aircraft Analysis Using the Layered and Extensible Aircraft Performance System (LEAPS) (2020).
URL ntrs.nasa.gov

[172] J. S. Gray, T. Hearn, K. Moore, J. T. Hwang, J. R. R. A. Martins, A. Ning, Automatic evaluation of multidisciplinary derivatives using a graph-based problem formulation in OpenMDAO, in: Proceedings of the 15th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Atlanta, GA, 2014. doi:10.2514/6.2014-2042.

[173] J. D. Mattingly, W. H. Heiser, D. T. Pratt, Aircraft Engine Design, 2nd Edition, American Institute of Aeronautics and Astronautics, Reston, VA, USA, 2000.

[174] J. Roskam, Airplane Design, Volumes 1-8, Roskam Aviation and Engineering Corporation, 1989.

[175] D. P. Raymer, Aircraft Design: A Conceptual Approach, 5th Edition, AIAA, 2012.

[176] J. T. Hwang, J. Jasa, J. R. R. A. Martins, High-fidelity design-allocation optimization of a commercial aircraft maximizing airline profit, Journal of Aircraft 56 (3) (2019) 1165–1178. doi:10.2514/1.C035082.

[177] E. W. Weisstein, Simpson's Rule, from MathWorld–A Wolfram Web Resource (2003).
URL http://mathworld.wolfram.com/SimpsonsRule.html

[178] T. A. Horne, TBM 850 By the Numbers, AOPA Pilot (jan 2012).

[179] SOCATA-Daher, Pilot's Information Manual - TBM 850 from S/N 434 to 999 (2015).

[180] T. A. Horne, King Air C90GTi: More Power, More Panel, AOPA Pilot (jun 2008).

[181] S. L. Schnulo, R. D. F. Jeff Chin, J. S. Gray, K. V. Papathakis, S. C. Clarke, N. Reid, N. K. Borer, Development of a multi-segment mission planning tool for SCEPTOR X-57, in: 2018 Multidisciplinary Analysis and Optimization Conference, AIAA, Atlanta, GA, 2018. doi:10.2514/6.2018-3738.

[182] J. Chin, S. L. Schnulo, T. Miller, K. Prokopius, J. S. Gray, Battery Performance Modeling on SCEPTOR X-57 Subject to Thermal and Transient Considerations , in: AIAA Scitech 2019 Forum, AIAA, San Diego, CA, 2019. doi:10.2514/6.2019-0784.

[183] M. Nöst, C. Doppler, M. Klell, A. Trattner, Thermal Management of PEM Fuel Cells in Electric Vehicles, SpringerBriefs in Applied Sciences and Technology, Springer International Publishing, 2017, p. 93–112. doi:10.1007/978-3-319-57445-5_7.

[184] R. Edwards, A. Smith, Design and Test of a Structurally-Integrated Heat Sink for the Maxwell X-57 High Lift Motor Controller (2019).
URL https://ntrs.nasa.gov/citations/20190033145

[185] R. H. Jansen, C. Bowman, A. Jankovsky, Sizing Power Components of an Electrically Driven Tail Cone Thruster and a Range Extender, in: 16th AIAA Aviation Technology, Integration, and Operations Conference, Washington, DC, 2016. doi:10.2514/6.2016-3766.

[186] B. J. Brelje, J. P. Jasa, J. R. R. A. Martins, J. S. Gray, Development of a conceptual-level thermal management system design capability in OpenConcept, in: NATO Research Symposium on Hybrid/Electric Aero-Propulsion Systems for Military Applications (AVT-RSY-323), Trondheim, NO, 2019. doi:10.14339/STO-MP-AVT-323.

[187] J. W. Chapman, S. L. Schnulo, M. P. Nitzsche, Development of a Thermal Management System for Electrified Aircraft. doi:10.2514/6.2020-0545.

[188] M. E. Bell, J. S. Litt, An Electrical Modeling and Thermal Analysis Toolbox for Electrified Aircraft Propulsion Simulation. doi:10.2514/6.2020-3676.

[189] S. Byahut, A. Uranga, Power distribution and thermal management modeling for electrified aircraft, in: 2020 AIAA/IEEE Electric Aircraft Technologies Symposium (EATS), 2020, pp. 1–15. doi:10.2514/6.2020-3578.

[190] F. P. Incropera, Fundamentals of Heat and Mass Transfer, John Wiley and Sons, Inc., USA, 2006.

[191] W. Kays, A. London, Compact Heat Exchangers, Third Edition, McGraw-Hill Book Company, 1984.

[192] J. P. Jasa, B. J. Brelje, C. A. Mader, J. R. R. A. Martins, Coupled design of a supersonic engine and thermal system, in: World Congress of Structural and Multidisciplinary Optimization, Beijing, China, 2019.

[193] R. M. Manglik, A. E. Bergles, Heat transfer and pressure drop correlations for the rectangular offset strip fin compact heat exchanger, Experimental Thermal and Fluid Science (1995) 171–180doi:10.1016/0894-1777(94)00096-Q.

[194] F. W. Meredith, Cooling of aircraft engines with special reference to ethylene glycol radiators enclosed in ducts, Tech. Rep. 1683, Government of the United Kingdom Air Ministry Reports and Memoranda (1935).

[195] T. Theodorsen, The fundamental principles of the n.a.c.a. cowling, Journal of the Aeronautical Sciences 5 (5) (1938) 169–174. doi:10.2514/8.566.

[196] The Dow Chemical Company, Dowfrost technical data sheet (2019).

[197] G. Bower, Tesla Model 3 Battery Cooling Much-Improved ... Track Mode? (August 2018).
URL https://insideevs.com/news/338711/tesla-model-3-battery-cooling-much-improved-track-mode/

[198] G. Bower, Tesla Model 3 Battery Can Transfer Twice The Heat Of Model S P100D (August 2018).
URL https://insideevs.com/news/339220/tesla-model-3-battery-can-transfer-twice-the-heat-of-model-s-p100d/

[199] W. A. Hermann, Liquid cooling manifold with multi-function thermal interface (U.S. Patent US8263250B2, 2012).

[200] H. Bhundiya, M. Hunt, Measurement of the effective radial thermal conductivities of 18650 and 26650 lithium ion battery, in: NASA Thermal and Fluids Analysis Workshop, 2018.

[201] F. Anton, eAircraft: Hybrid-elektrische Antriebe für Luftfahrzeuge, in: 14. Tag der Deutschen Luft- und Raumfahrtregionen, Potsdam, 2019.

[202] MagniX, magni250 and magni500 Specification (2021).
URL https://www.magnix.aero/products/

[203] The Dow Chemical Company, SYLTHERM XLT Heat Transfer Fluid Product Technical Data (1998).

[204] D. Tracy, The Tesla Model 3 'Superbottle' Easter Egg Is a Fascinating Packaging Solution (December 2018).
URL https://jalopnik.com/the-tesla-model-3s-superbottle-easter-egg-is-a-fascin-1830992728

[205] M. J. Moran, H. N. Shapiro, D. D. Boettner, M. B. Bailey, Fundamentals of engineering thermodynamics, John Wiley & Sons, 2010.

[206] De Kleijn Energy Consultants and Engineers, Coefficient of Performance (2021).
URL https://industrialheatpumps.nl/en/how_it_works/cop_heat_pump/

[207] G. J. Kennedy, J. E. Hicken, Improved constraint-aggregation methods, Computer Methods in Applied Mechanics and Engineering 289 (2015) 332–354. doi:10.1016/j.cma.2015.02.017.

[208] A. B. Lambe, J. R. R. A. Martins, G. J. Kennedy, An evaluation of constraint aggregation strategies for wing box mass minimization, Structural and Multidisciplinary Optimization 55 (1) (2017) 257–277. doi:10.1007/s00158-016-1495-1.

[209] Honeywell, Micro Vapor Cycle System (2020).
URL https://aerospace.honeywell.com/en/learn/products/air-and-thermal-systems/micro-vapor-cycle-system

[210] M. Thurber, Honeywell develops a/c system for future flyers, in: Aviation International News, 2020.
URL https://www.ainonline.com/aviation-news/aerospace/2020-06-22/honeywell-develops-c-system-future-flyers

[211] F. M. White, Viscous Fluid Flow, McGraw-Hill, New York, NY, 1991.

[212] Eaton Aeroquip, Rubber Hose, Fittings and Assemblies for Aerospace Applications (2013).
URL https://shop.boeing.com/medias/TF100-16E-Rubber-Hose-Fittings-Assys.pdf

[213] Pipistrel Vertical Solutions d.o.o., EASA Type-Certificate Data Sheet No. EASA.A.573 for type Virus SW 121 (2021).

[214] Pipistrel Vertical Solutions d.o.o., E-811 Technical Characteristics (2021).
URL https://www.pipistrel-aircraft.com/aircraft/electric-flight/e-811/

[215] Pipistrel Vertical Solutions d.o.o., EASA Type-Certificate Data Sheet No. EASA.E.234 for type E-811 Engine (2020).

[216] G. K. Kenway, R. Henderson, J. E. Hicken, N. B. Kuntawala, D. W. Zingg, J. R. R. A. Martins, R. G. McKeand, Reducing aviation's environmental impact through large aircraft for short ranges, in: Proceedings of the 48th AIAA Aerospace Sciences Meeting and Exhibit, Orlando, FL, 2010. doi:10.2514/6.2010-1015.

[217] R. W. Dyson, L. Rodriguez, M. Ellen Roth, P. Raitano, Solid-State Exergy Optimized Electric Aircraft Thermal and Fault Management, in: 2020 AIAA/IEEE Electric Aircraft Technologies Symposium (EATS), AIAA, 2020.

[218] E. S. Hendricks, J. S. Gray, pycycle: A tool for efficient optimization of gas turbine engine cycles, Aerospace 6 (87) (August 2019). doi:10.3390/aerospace6080087.

[219] S. M. Jones, W. J. Haller, M. T. Tong, An N+3 technology level reference propulsion system, Tech. rep., NASA Glenn Research Center, nASA/TM—2017-219501 (2017).
URL https://ntrs.nasa.gov/citations/20170005426

[220] M. Nita, D. Scholz, Estimating the Oswald Factor from Basic Aircraft Geometrical Parameters, in: Deutscher Luft- und Raumfahrtkongress, Berlin, Germany, 2012.

[221] 737 Technical Site, Technical Specifications (2021).
URL http://www.b737.org.uk/techspecsdetailed.htm

[222] P. E. Gill, W. Murray, M. A. Saunders, SNOPT: An SQP algorithm for large-scale constrained optimization, SIAM Review 47 (1) (2005) 99–131. doi:10.1137/S0036144504446096.

[223] N. Wu, G. Kenway, C. A. Mader, J. Jasa, J. R. R. A. Martins, pyOptSparse: a Python framework for large-scale constrained nonlinear optimization of sparse systems, Journal of Open Source Software 5 (54) (2020) 2564. doi:10.21105/joss.02564.

[224] A. Wächter, L. T. Biegler, On the implementation of an interior point filter line-search algorithm for large-scale nonlinear programming, Mathematical Programming 106 (2006) 25–57.

[225] Z. Lyu, G. K. W. Kenway, J. R. R. A. Martins, Aerodynamic shape optimization investigations of the Common Research Model wing benchmark, AIAA Journal 53 (4) (2015) 968–985. doi:10.2514/1.J053318.

[226] Z. Lyu, J. R. R. A. Martins, Aerodynamic design optimization studies of a blended-wing-body aircraft, Journal of Aircraft 51 (5) (2014) 1604–1617. doi:10.2514/1.C032491.

[227] Uber Technologies, eVTOL common reference models (2018).
URL https://www.uber.com/info/elevate/ecrm/

[228] G. K. Kenway, G. J. Kennedy, J. R. R. A. Martins, A CAD-free approach to high-fidelity aerostructural optimization, in: Proceedings of the 13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference, no. AIAA 2010-9231, Fort Worth, TX, 2010. doi:10.2514/6.2010-9231.

[229] T. R. Brooks, G. K. W. Kenway, J. R. R. A. Martins, Benchmark aerostructural models for the study of transonic aircraft wings, AIAA Journal 56 (7) (2018) 2840–2855. doi:10.2514/1.J056603.

[230] S. Chen, Z. Lyu, G. K. W. Kenway, J. R. R. A. Martins, Aerodynamic shape optimization of the Common Research Model wing-body-tail configuration, Journal of Aircraft 53 (1) (2016) 276–293. doi:10.2514/1.C033328.

[231] G. Carrier, D. Destarac, A. Dumont, M. Méheut, I. S. E. Din, J. Peter, S. B. Khelil, J. Brezillon, M. Pestana, Gradient-based aerodynamic optimization with the $elsA$ software, in: 52nd Aerospace Sciences Meeting, 2014. doi:10.2514/6.2014-0568.

[232] G. R. Anderson, M. Nemec, M. J. Aftosmis, Aerodynamic shape optimization benchmarks with error control and automatic parameterization, in: 53rd AIAA Aerospace Sciences Meeting, 2015, p. 1719. doi:10.2514/6.2015-1719.

[233] E. Fabiano, D. Mavriplis, Adjoint-based aerodynamic design on unstructured meshes, in: 54nd Aerospace Sciences Meeting, 2016. doi:10.2514/6.2016-1295.

[234] X. He, J. Li, C. A. Mader, A. Yildirim, J. R. R. A. Martins, Robust aerodynamic shape optimization—from a circle to an airfoil, Aerospace Science and Technology 87 (2019) 48–61. doi:10.1016/j.ast.2019.01.051.

[235] F. Bisson, Nadarajah, Adjoint-based aerodynamic optimization of benchmark problems, in: 53nd Aerospace Sciences Meeting, 2015. doi:10.2514/6.2015-1948.

[236] S. T. LeDoux, J. C. Vassberg, D. P. Young, S. Fugal, D. Kamenetskiy, W. P. Huffman, R. G. Melvin, M. F. Smith, Study based on the AIAA aerodynamic design optimization discussion group test cases, AIAA Journal 53 (2015) 1910–1935. doi:10.2514/1.j053535.

[237] Y. Zhang, Z.-H. Han, L. Shi, W.-P. Song, Multi-round surrogate-based optimization for benchmark aerodynamic design problems, in: 54th AIAA Aerospace Sciences Meeting, 2016, p. 1545. doi:10.2514/6.2016-1545.

[238] D. A. Masters, D. J. Poole, N. J. Taylor, T. Rendall, C. B. Allen, Impact of shape parameterisation on aerodynamic optimisation of benchmark problems, in: 54rd AIAA Aerospace Sciences Meeting, 2016. doi:10.2514/6.2016-1544.

[239] G. R. Anderson, M. J. Aftosmis, Adaptive shape control for aerodynamic design, in: 56th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, 2015, p. 0398. doi:10.2514/6.2015-0398.

[240] G. Fasano, Solving Non-standard Packing Problems by Global Optimization and Heuristics, 2014. doi:10.1007/978-3-319-05005-8.

[241] G. M. Fadel, M. M. Wiecek, Packing Optimization of Free-Form Objects in Engineering Design, Springer International Publishing, Cham, 2015, pp. 37–66. doi:10.1007/978-3-319-18899-7_3.

[242] F. Eisenbrand, S. Funke, J. Reichel, E. Schomer, Packing a trunk, in: Algorithms - ESA 2003, 2003. doi:10.1007/978-3-540-39658-1_56.

[243] S. Tiwari, G. Fadel, P. Fenyes, A Fast and Efficient Compact Packing Algorithm for SAE and ISO Luggage Packing Problems, Journal of Computing and Information Science in Engineering 10 (2) (2010) 021010. doi:10.1115/1.3330440.

[244] Y. Yu, Z. Lyu, Z. Xu, J. R. R. A. Martins, On the influence of optimization algorithm and starting design on wing aerodynamic shape optimization, Aerospace Science and Technology 75 (2018) 183–199. doi:10.1016/j.ast.2018.01.016.

[245] Y. Stoyan, T. Romanova, A. Pankratov, A. Chugay, Optimized Object Packings Using Quasi-Phi-Functions, Springer International Publishing, Cham, 2015, pp. 265–293. doi:10.1007/978-3-319-18899-7_13.

[246] D. Horvat, Ray-casting point-in-polyhedron test, in: Proceedings of CESCG 2012: The 16th Central European Seminar on Computer Graphics, 2012.

[247] C. Ericson, Real-Time Collision Detection, CRC Press, Inc., Boca Raton, FL, USA, 2004. doi:10.1201/b14581.

[248] G. Kreisselmeier, R. Steinhauser, Systematic control design by optimizing a vector performance index, in: International Federation of Active Controls Symposium on Computer-Aided Design of Control Systems, Zurich, Switzerland, 1979. doi:10.1016/S1474-6670(17)65584-8.

[249] T. Möller, A fast triangle-triangle intersection test, J. Graph. Tools 2 (2) (1997) 25–30. doi:10.1080/10867651.1997.10487472.

[250] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mane, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viegas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng, TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems (2016). arXiv:1603.04467.
URL http://arxiv.org/abs/1603.04467

[251] N. R. Secco, J. P. Jasa, G. K. W. Kenway, J. R. R. A. Martins, Component-based geometry manipulation for aerodynamic shape optimization with overset meshes, AIAA Journal 56 (9) (2018) 3667–3679. doi:10.2514/1.J056550.

[252] A. Yildirim, G. K. W. Kenway, C. A. Mader, J. R. R. A. Martins, A Jacobian-free approximate Newton–Krylov startup strategy for RANS simulations, Journal of Computational Physics 397 (2019) 108741. doi:10.1016/j.jcp.2019.06.018.

[253] G. K. W. Kenway, C. A. Mader, P. He, J. R. R. A. Martins, Effective adjoint approaches for computational fluid dynamics, Progress in Aerospace Sciences 110 (2019) 100542. doi:10.1016/j.paerosci.2019.05.002.

[254] G. K. W. Kenway, J. R. R. A. Martins, Multipoint aerodynamic shape optimization investigations of the Common Research Model wing, AIAA Journal 54 (1) (2016) 113–128. doi:10.2514/1.J054154.

[255] G. K. W. Kenway, J. R. R. A. Martins, Buffet-onset constraint formulation for aerodynamic shape optimization, AIAA Journal 55 (6) (2017) 1930–1947. doi:10.2514/1.J055172.

[256] N. R. Secco, J. R. R. A. Martins, RANS-based aerodynamic shape optimization of a strut-braced wing with overset meshes, in: 2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, American Institute of Aeronautics and Astronautics, Kissimmee, FL, 2018. doi:10.2514/6.2018-0413.

[257] E. Luke, E. Collins, E. Blades, A fast mesh deformation method using explicit interpolation, Journal of Computational Physics 231 (2) (2012) 586–601. doi:10.1016/j.jcp.2011.09.021.

[258] W. M. Chan, J. L. Steger, Enhancements of a three-dimensional hyperbolic grid generation scheme, Applied Mathematics and Computation 51 (2–3) (1992) 181–205. doi:10.1016/0096-3003(92)90073-A.

[259] P. E. Gill, W. Murray, M. A. Saunders, SNOPT: An SQP algorithm for large-scale constrained optimization, SIAM Journal of Optimization 12 (4) (2002) 979–1006. doi:10.1137/S1052623499350013.

[260] M. Reed, U. Raschke, R. Tirumali, B. Parkinson, Developing and implementing parametric human body shape models in ergonomics software, in: 3rd International Digital Human Modeling Conference, 2014.

[261] B. J. Brelje, J. Anibal, A. Yildirim, C. A. Mader, J. R. R. A. Martins, Flexible formulation of spatial integration constraints in aerodynamic shape optimization, AIAA Journal 58 (6) (2020) 2571–2580. doi:10.2514/1.J058366.

[262] K. Chen, S. Wang, M. Song, L. Chen, Configuration optimization of battery pack in parallel air-cooled battery thermal management system using an optimization strategy, Applied Thermal Engineering 123 (2017) 177–186. doi:10.1016/j.applthermaleng.2017.05.060.

[263] B. Severino, F. Gana, R. Palma-Behnke, P. A. Estevez, W. R. Calderon-Munoz, M. E. Orchard, J. Reyes, M. Cortes, Multi-objective optimal design of lithium-ion battery packs based on evolutionary algorithms, J. Power Sources 267 (2014) 288–299. doi:10.1016/j.jpowsour.2014.05.088.

[264] L. Shui, F. Chen, A. Garg, X. Peng, N. Bao, J. Zhang, Design optimization of battery pack enclosure for electric vehicle, Structural and Multidisciplinary Optimization 58 (2018) 331–347. doi:10.1007/s00158-018-1901-y.

[265] T. Placke, R. Kloepsch, S. Dühnen, M. Winter, Lithium ion, lithium metal, and alternative rechargeable batterytechnologies: the odyssey for high energy density, J. Solid State Electrochem. 21 (2017) 1939—1964. doi:10.1007/s10008-017-3610-7.

[266] Shell, Aviation Fuel – Jet A/A-1 (2018).
URL https://www.exxonmobil.com/en/aviation/products-and-services/products/jet-a-jet-a-1

[267] C. A. Mader, G. K. W. Kenway, A. Yildirim, J. R. R. A. Martins, ADflow: An open-source computational fluid dynamics solver for aerodynamic and multidisciplinary optimization, Journal of Aerospace Information Systems (2020). doi:10.2514/1.I010796.

[268] B. J. Brelje, J. R. R. A. Martins, Coupled component sizing and aerodynamic shape optimization via geometric constraints, in: AIAA AVIATION Forum, American Institute of Aeronautics and Astronautics, Dallas, TX, 2019. doi:10.2514/6.2019-3105.

[269] D. Kramer, Hydrogen-powered aircraft may be getting a lift, Physics Today 73 (12) (2020) 27–29. doi:10.1063/PT.3.4632.

[270] A. Silverstein, E. W. Hall, Liquid hydrogen as a jet fuel for high-altitude aircraft, Tech. Rep. RM E55C28a, NACA, Washington, DC (April 1955).

[271] J. L. Sloop, Liquid hydrogen as a propulsion fuel, 1945-1959., Tech. Rep. SP 4404, NASA, Washington, DC (1978).

[272] H2FLY, The future of aviation (2020).
URL http://h2fly.de

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

[274] T. W. Sederberg, S. R. Parry, Free-form deformation of solid geometric models, SIGGRAPH Comput. Graph. 20 (4) (1986) 151–160. doi:10.1145/15886.15903.

[275] R. Haimes, J. Dannenhoffer, The engineering sketch pad: A solid-modeling, feature-based, web-enabled system for building parametric geometry, in: 21st AIAA Computational Fluid Dynamics Conference, Fluid Dynamics and Co-located Conferences, American Institute of Aeronautics and Astronautics, 2013. doi:10.2514/6.2013-3073.

[276] L. Hascoet, V. Pascual, The Tapenade automatic differentiation tool: Principles, model, and specification, ACM Transactions on Mathematical Software 39 (3) (2013) 20:1–20:43. doi:10.1145/2450153.2450158.

[277] P. Peterson, F2PY: A tool for connecting Fortran and Python programs., International Journal of Computational Science and Engineering 4 (4) (2009) 296–305. doi:10.1504/ijcse.2009.029165.

[278] P. E. Gill, W. Murray, M. A. Saunders, User's Guide for SNOPT Version 7: Software for Large-Scale Nonlinear Programming, Systems Optimization Laboratory, Stanford University, California, 94305-4023, technical Report (2007).

[279] T. Q. Hua, H.-S. Roh, R. K. Ahluwalia, Performance assessment of 700-bar compressed hydrogen storage for light duty fuel cell vehicles, International Journal of Hydrogen Energy 42 (40) (2017) 25121 – 25129. doi:10.1016/j.ijhydene.2017.08.123.

[280] Toray 3960 prepreg system, Tech. rep., Toray Composite Materials America (2020).
URL https://www.toraycma.com/files/library/0b1f847f1abb9142.pdf

[281] R. Falck, J. S. Gray, K. Ponnapalli, T. Wright, dymos: A python package for optimal control of multidisciplinary systems, Journal of Open Source Software 6 (59) (2021) 2809. doi:10.21105/joss.02809.

[282] G. E. P. Box, K. B. Wilson, On the experimental attainment of optimum conditions, Journal of the Royal Statistical Society: Series B (Methodological) 13 (1) (1951) 1–38. doi:10.1111/j.2517-6161.1951.tb00067.x.

[283] J. Sacks, S. B. Schiller, W. J. Welch, Designs for computer experiments, Technometrics 31 (1) (1989) 41–47. doi:10.1080/00401706.1989.10488474.

[284] M. A. Bouhlel, J. T. Hwang, N. Bartoli, R. Lafage, J. Morlier, J. R. R. A. Martins, A Python surrogate modeling framework with derivatives, Advances in Engineering Software (2019) 102662doi:https://doi.org/10.1016/j.advengsoft.2019.03.005.

[285] M. A. Bouhlel, N. Bartoli, J. Morlier, A. Otsmane, An improved approach for estimating the hyperparameters of the kriging model for high-dimensional problems through the partial least squares method, Mathematical Problems in EngineeringArticle ID 6723410 (2016). doi:10.1155/2016/6723410.

[286] J. Jasa, B. Brelje, J. Gray, C. A. Mader, J. R. R. A. Martins, Large-scale path-dependent optimization of supersonic aircraft, Aerospace 7 (152) (October 2020). doi:10.3390/aerospace7100152.

[287] J. T. Hwang, J. R. R. A. Martins, A fast-prediction surrogate model for large datasets, Aerospace Science and Technology 75 (2018) 74–87. doi:10.1016/j.ast.2017.12.030.

[288] T. Hastie, R. Tibshirani, J. Friedman, The elements of statistical learning – data mining, inference, and prediction (2017).

[289] Y. Yang, Regression with multiple candidate models: Selecting or mixing?, STATISTICA SINICA 13 (1999) 783–809.

[290] Michael Kruger, OpenVSP Hangar - Blended Wing Body (2017).
URL http://hangar.openvsp.org/vspfiles/333

[291] Eaton Corporation, Airframe Fuel Pumps (2021).
URL https://www.eaton.com/Eaton/ProductsServices/Aerospace/Fuel/PCT_247872